

# Tibero

## 참조 안내서

Tibero 7

**TMAXTibero**

Copyright © 2025 TmaxTibero Co., Ltd. All Rights Reserved.

## Copyright Notice

Copyright © 2025 TmaxTibero Co., Ltd. All Rights Reserved.

대한민국 경기도 성남시 분당구 정자일로 45, 티맥스소프트타워 우)13613

## Website

<http://www.tmaxtibero.com>

## 기술서비스센터

Tel : +82-1544-8629

E-Mail : [info@tmax.co.kr](mailto:info@tmax.co.kr)

## Restricted Rights Legend

All TmaxTibero Software (Tibero®) and documents are protected by copyright laws and international convention. TmaxTibero software and documents are made available under the terms of the TmaxTibero License Agreement and this document may only be distributed or copied in accordance with the terms of this agreement. No part of this document may be transmitted, copied, deployed, or reproduced in any form or by any means, electronic, mechanical, or optical, without the prior written consent of TmaxTibero Co., Ltd. Nothing in this software document and agreement constitutes a transfer of intellectual property rights regardless of whether or not such rights are registered) or any rights to TmaxTibero trademarks, logos, or any other brand features.

This document is for information purposes only. The company assumes no direct or indirect responsibilities for the contents of this document, and does not guarantee that the information contained in this document satisfies certain legal or commercial conditions. The information contained in this document is subject to change without prior notice due to product upgrades or updates. The company assumes no liability for any errors in this document.

이 소프트웨어(Tibero®) 사용설명서의 내용과 프로그램은 저작권법과 국제 조약에 의해서 보호받고 있습니다. 사용설명서의 내용과 여기에 설명된 프로그램은 TmaxTibero Co., Ltd.와의 사용권 계약 하에서만 사용이 가능하며, 사용설명서는 사용권 계약의 범위 내에서만 배포 또는 복제할 수 있습니다. 이 사용설명서의 전부 또는 일부분을 TmaxTibero의 사전 서면 동의 없이 전자, 기계, 녹음 등의 수단을 사용하여 전송, 복제, 배포, 2차적 저작물작성 등의 행위를 하여서는 안 됩니다.

이 소프트웨어 사용설명서와 프로그램의 사용권 계약은 어떠한 경우에도 사용설명서 및 프로그램과 관련된 지적 재산권(등록 여부를 불문)을 양도하는 것으로 해석되지 아니하며, 브랜드나 로고, 상표 등을 사용할 권한을 부여하지 않습니다. 사용설명서는 오로지 정보의 제공만을 목적으로 하고, 이로 인한 계약상의 직접적 또는 간접적 책임을 지지 아니하며, 사용설명서 상의 내용은 법적 또는 상업적인 특정한 조건을 만족시키는 것을 보장하지는 않습니다. 사용설명서의 내용은 제품의 업그레이드나 수정에 따라 그 내용이 예고 없이 변경될 수 있으며, 내용상의 오류가 없음을 보장하지 아니합니다.

## Trademarks

Tibero® is a registered trademark of TmaxTibero Co., Ltd. Other products, titles or services may be registered trademarks of their respective companies.

---

Tibero®는 TmaxTibero Co., Ltd.의 등록 상표입니다. 기타 모든 제품들과 회사 이름은 각각 해당 소유주의 상표로서 참조용으로만 사용됩니다.

### Open Source Software Notice

Some modules or files of this product are subject to the terms of the following licenses. : OpenSSL, RSA Data Security, Inc., Apache Foundation, Jean-loup Gailly and Mark Adler, Paul Hsieh's hash

Detailed Information related to the license can be found in the following directory : \${INSTALL\_PATH}/license/oss\_licenses

본 제품의 일부 파일 또는 모듈은 다음의 라이선스를 준수합니다. : OpenSSL, RSA Data Security, Inc., Apache Foundation, Jean-loup Gailly and Mark Adler, Paul Hsieh's hash

관련 상세한 정보는 제품의 다음의 디렉터리에 기재된 사항을 참고해 주십시오. : \${INSTALL\_PATH}/license/oss\_licenses

### 안내서 정보

안내서 제목: Tibero 참조 안내서

발행일: 2025-09-30

소프트웨어 버전: Tibero 7.2.4

안내서 버전: v7.2.4

---



# 내용 목차

|                                                            |          |
|------------------------------------------------------------|----------|
| 안내서에 대하여 .....                                             | xxxiii   |
| <b>제1장 Introduction to Initialization Parameters .....</b> | <b>1</b> |
| 1.1. Overview .....                                        | 1        |
| 1.2. Class Attributes .....                                | 2        |
| 1.3. Initialization Parameter .....                        | 2        |
| 1.3.1. ACF_CLGC_CNT .....                                  | 2        |
| 1.3.2. ACF_CMPT_CNT .....                                  | 2        |
| 1.3.3. ACF_NMGR_CM_AUTO_REVIVAL .....                      | 3        |
| 1.3.4. ACF_NMGR_PREVENT_REVIVAL_ON_IR .....                | 3        |
| 1.3.5. ACTIVE_SESSION_HISTORY .....                        | 3        |
| 1.3.6. ACTIVE_SESSION_TIMEOUT .....                        | 4        |
| 1.3.7. ACTIVE_TX_SESSION_TIMEOUT .....                     | 4        |
| 1.3.8. ACT_DUMP_BACKUP_LIMIT_CNT .....                     | 5        |
| 1.3.9. ACT_DUMP_DEST .....                                 | 5        |
| 1.3.10. ACT_DUMP_TOTAL_SIZE_LIMIT .....                    | 5        |
| 1.3.11. ACT_LOG_DEST .....                                 | 6        |
| 1.3.12. ACT_LOG_FILE_SIZE .....                            | 6        |
| 1.3.13. ACT_LOG_TOTAL_SIZE_LIMIT .....                     | 6        |
| 1.3.14. AD_AUTH_KBR_SERVICE_NAME .....                     | 7        |
| 1.3.15. AD_AUTH_KEYTAB_FILE .....                          | 7        |
| 1.3.16. AD_AUTH_PREFIX .....                               | 7        |
| 1.3.17. ALL_LOGFILE_MEMBERS_SANITY_CHECK .....             | 8        |
| 1.3.18. AS_ADDR .....                                      | 8        |
| 1.3.19. AS_ALLOW_ONLY_RAW_DISKS .....                      | 8        |
| 1.3.20. AS_DDL_DROP_FILE .....                             | 9        |
| 1.3.21. AS_DISKSTRING .....                                | 9        |
| 1.3.22. AS_DISK_REPAIR_WAIT_TIME .....                     | 9        |
| 1.3.23. AS_PORT .....                                      | 10       |
| 1.3.24. AS_SCAN_SSVR_DISK .....                            | 10       |
| 1.3.25. AS_WTHR_CNT .....                                  | 10       |
| 1.3.26. AUDIT_FILE_DEST .....                              | 11       |
| 1.3.27. AUDIT_FILE_SIZE .....                              | 11       |
| 1.3.28. AUDIT_LOG_TOTAL_SIZE_LIMIT .....                   | 11       |
| 1.3.29. AUDIT_SYS_OPERATIONS .....                         | 12       |
| 1.3.30. AUDIT_TRAIL .....                                  | 12       |
| 1.3.31. AUTHENTICATION_TIMEOUT .....                       | 12       |
| 1.3.32. AUTO_COALESCE .....                                | 13       |
| 1.3.33. AUTO_COALESCE_INTERVAL .....                       | 13       |
| 1.3.34. AUTO_COALESCE_STOP .....                           | 13       |
| 1.3.35. BATCHUPDATE_MODE .....                             | 14       |

|         |                                              |    |
|---------|----------------------------------------------|----|
| 1.3.36. | BATCH_INSERT_ROW_CNT_THRESHOLD .....         | 14 |
| 1.3.37. | BIND_VARIABLE_CAPTURE .....                  | 14 |
| 1.3.38. | BIND_VARIABLE_CAPTURE_ALLOC_COUNT .....      | 15 |
| 1.3.39. | BIND_VARIABLE_CAPTURE_INTERVAL .....         | 15 |
| 1.3.40. | BIND_VARIABLE_CAPTURE_KEEP_FIRST .....       | 15 |
| 1.3.41. | BIND_VARIABLE_CAPTURE_MAX_COUNT .....        | 16 |
| 1.3.42. | BIND_VARIABLE_CAPTURE_MAX_VAR_COUNT .....    | 16 |
| 1.3.43. | BIND_VARIABLE_CAPTURE_POOL_PCTDEL .....      | 16 |
| 1.3.44. | BIND_VARIABLE_CAPTURE_POOL_SIZE .....        | 17 |
| 1.3.45. | BLOCK_CHANGE_TRACKING .....                  | 17 |
| 1.3.46. | BOOT_WITH_AUTO_DOWN_CLEAN .....              | 17 |
| 1.3.47. | BOOT_WITH_UNKNOWN_PARAMETER .....            | 18 |
| 1.3.48. | BOOT_WITH_WALLET_CHECK .....                 | 18 |
| 1.3.49. | BTIP_FILE_PATH .....                         | 18 |
| 1.3.50. | CACHE_SIZE_FACTOR .....                      | 19 |
| 1.3.51. | CALLSTACK_DUMP_BACKUP_LIMIT_CNT .....        | 19 |
| 1.3.52. | CALLSTACK_DUMP_DEST .....                    | 20 |
| 1.3.53. | CALLSTACK_DUMP_TOTAL_SIZE_LIMIT .....        | 20 |
| 1.3.54. | CCC_CATH_CNT .....                           | 20 |
| 1.3.55. | CCC_MASTERING_UNIT .....                     | 21 |
| 1.3.56. | CCC_RECL_MAX_RESOURCES .....                 | 21 |
| 1.3.57. | CCC_RSBTBL_SIZE .....                        | 21 |
| 1.3.58. | CC_CU_BLK_CNT .....                          | 22 |
| 1.3.59. | CC_CU_PCTUSE .....                           | 22 |
| 1.3.60. | CC_CU_WRITEOUT_THRESHOLD .....               | 22 |
| 1.3.61. | CC_EXPECTED_RATE .....                       | 23 |
| 1.3.62. | CC_TYPICAL_ROW_SIZE .....                    | 23 |
| 1.3.63. | CERTIFICATE_FILE .....                       | 23 |
| 1.3.64. | CHECK_RESULT_FOR_MEMLOG_DUMP .....           | 24 |
| 1.3.65. | CLOSE_REMOTE_SESSION_ON_SESSION_RESET .....  | 24 |
| 1.3.66. | CLUSTER_DATABASE .....                       | 24 |
| 1.3.67. | CMPT_CNT_FOR_RECONFIGURATION .....           | 25 |
| 1.3.68. | CM_AGENT_THREAD_CHECK_EXPIRE .....           | 25 |
| 1.3.69. | CM_AGENT_THREAD_INTERVAL .....               | 25 |
| 1.3.70. | CM_AUTO_RECONNECT_NODE_WITH_ACTIVE_FHB ..... | 26 |
| 1.3.71. | CM_AUTO_RECONNECT_TIMEOUT .....              | 26 |
| 1.3.72. | CM_AUTO_REVIVAL .....                        | 27 |
| 1.3.73. | CM_CLUSTER_AUTO_RESTART .....                | 27 |
| 1.3.74. | CM_CLUSTER_WORKER_CNT .....                  | 27 |
| 1.3.75. | CM_CSP_FENCE_WAIT_CNT .....                  | 28 |
| 1.3.76. | CM_DOWN_ON_TIME_DISACCORDANCE .....          | 28 |
| 1.3.77. | CM_ENABLE_CLUSTER_WORKER .....               | 28 |
| 1.3.78. | CM_ENABLE_FAST_ACT_LOOP .....                | 29 |

|          |                                           |    |
|----------|-------------------------------------------|----|
| 1.3.79.  | CM_ENABLE_FAST_INST_DOWN_MODE .....       | 29 |
| 1.3.80.  | CM_ENABLE_FAST_MASTER_ELECTION .....      | 29 |
| 1.3.81.  | CM_ENABLE_FAST_NET_ERROR_DETECTION .....  | 30 |
| 1.3.82.  | CM_ENABLE_SET_LIMITS_FOR_INSTANCE .....   | 30 |
| 1.3.83.  | CM_ENABLE_WAIT_DOWN_NOTI_FOR_SVC .....    | 30 |
| 1.3.84.  | CM_FENCE .....                            | 31 |
| 1.3.85.  | CM_FILE_AUTO_RESTART_SSVR .....           | 31 |
| 1.3.86.  | CM_FORCE_INSTANCE_KILL .....              | 31 |
| 1.3.87.  | CM_GUARD_LOG_DEST .....                   | 32 |
| 1.3.88.  | CM_HEARTBEAT_EXPIRE .....                 | 32 |
| 1.3.89.  | CM_ID .....                               | 32 |
| 1.3.90.  | CM_LOCAL_IP_VERSION .....                 | 33 |
| 1.3.91.  | CM_LOG_DEST .....                         | 33 |
| 1.3.92.  | CM_LOG_FILE_SIZE .....                    | 33 |
| 1.3.93.  | CM_LOG_TOTAL_SIZE_LIMIT .....             | 34 |
| 1.3.94.  | CM_LOOP_TIMEOUT .....                     | 34 |
| 1.3.95.  | CM_MODE_OBSERVER .....                    | 34 |
| 1.3.96.  | CM_MST_SAFEGUARD_EXPIRE .....             | 35 |
| 1.3.97.  | CM_NAME .....                             | 35 |
| 1.3.98.  | CM_NETWORK_AUTO_RESTART .....             | 35 |
| 1.3.99.  | CM_NET_EXPIRE_MARGIN .....                | 36 |
| 1.3.100. | CM_OBSERVER_EXPIRE .....                  | 36 |
| 1.3.101. | CM_OBSERVER_PORT .....                    | 36 |
| 1.3.102. | CM_PORT .....                             | 37 |
| 1.3.103. | CM_REGARD_NETHB_AS_SERIOUS .....          | 37 |
| 1.3.104. | CM_RESOURCE_FILE .....                    | 37 |
| 1.3.105. | CM_RESOURCE_FILE_BACKUP .....             | 38 |
| 1.3.106. | CM_RESOURCE_FILE_BACKUP_INTERVAL .....    | 38 |
| 1.3.107. | CM_REVIVAL_WAIT_INST_TIMEOUT .....        | 39 |
| 1.3.108. | CM_REVIVAL_WAIT_NODE_TIMEOUT .....        | 39 |
| 1.3.109. | CM_REVIVAL_WAIT_TIMEOUT .....             | 39 |
| 1.3.110. | CM_SVC_AUTO_START_ON_BOOT .....           | 40 |
| 1.3.111. | CM_SVC_DOWN_NOTI_WAIT_TIMEOUT .....       | 40 |
| 1.3.112. | CM_TAS_VM_ADDR .....                      | 40 |
| 1.3.113. | CM_TAS_VM_PORT .....                      | 41 |
| 1.3.114. | CM_TIME_DISACCORDANCE_NOTI_INTERVAL ..... | 41 |
| 1.3.115. | CM_TIME_UNIT .....                        | 41 |
| 1.3.116. | CM_UI_PORT .....                          | 42 |
| 1.3.117. | CM_USE_ASENV .....                        | 42 |
| 1.3.118. | CM_USE_AWS_EC2 .....                      | 42 |
| 1.3.119. | CM_USE_AWS_EC2_PRIVATE .....              | 43 |
| 1.3.120. | CM_USE_AWS_EC2_PUBLIC .....               | 43 |
| 1.3.121. | CM_USE_AZURE_VM_PRIVATE .....             | 43 |

|          |                                              |    |
|----------|----------------------------------------------|----|
| 1.3.122. | CM_USE_KUBERNETES .....                      | 44 |
| 1.3.123. | CM_USE_TAS5_STORAGE .....                    | 44 |
| 1.3.124. | CM_VIP_CHECK_KUBERNETES_SERVICE_STATUS ..... | 44 |
| 1.3.125. | CM_VIP_FAILOVER_BASED_ON_SVC_STATUS .....    | 45 |
| 1.3.126. | CM_VIP_RESTART_COUNT .....                   | 45 |
| 1.3.127. | CM_VIP_STAT_CHECK_INTERVAL .....             | 45 |
| 1.3.128. | CM_VIP_USE_KUBERNETES_EXTERNAL_IP .....      | 46 |
| 1.3.129. | CM_WATCHDOG_EXPIRE .....                     | 46 |
| 1.3.130. | COMMIT_FLUSH_REQUEST .....                   | 46 |
| 1.3.131. | COMMIT_FLUSH_WAIT .....                      | 47 |
| 1.3.132. | COMMIT_POINT_STRENGTH .....                  | 47 |
| 1.3.133. | CONTROL_FILES .....                          | 47 |
| 1.3.134. | CPU_COUNT .....                              | 48 |
| 1.3.135. | CREATE_STORED_OUTLINES .....                 | 48 |
| 1.3.136. | CRYPTO_LEVEL .....                           | 48 |
| 1.3.137. | CTHR_CHECK_CONNECTION_INTERVAL .....         | 49 |
| 1.3.138. | CURSOR_SHARING .....                         | 49 |
| 1.3.139. | CWS_DDD_TIMEOUT_ON_SLOG .....                | 49 |
| 1.3.140. | CWS_RECL_MAX_RESOURCES .....                 | 50 |
| 1.3.141. | CWS_RESOLVE_CVT_DEADLK_FORCE .....           | 50 |
| 1.3.142. | CWS_RSBL_SIZE .....                          | 50 |
| 1.3.143. | DATABASE_REPLAY_CAPTURE_DEST .....           | 51 |
| 1.3.144. | DATABASE_REPLAY_CAPTURE_DIR_LIMIT .....      | 51 |
| 1.3.145. | DATABASE_REPLAY_CAPTURE_FILE_SIZE .....      | 51 |
| 1.3.146. | DATABASE_REPLAY_CAPTURE_WORKLOAD .....       | 52 |
| 1.3.147. | DATABASE_REPLAY_DUMP_MMAP_SIZE .....         | 52 |
| 1.3.148. | DBLINK_BATCH_UPDATE_ROW_COUNT .....          | 52 |
| 1.3.149. | DBLINK_CONN_TIMEOUT .....                    | 53 |
| 1.3.150. | DBLINK_CONN_VALIDATION_SKIP_INTERVAL .....   | 53 |
| 1.3.151. | DBLINK_QUERY_TIMEOUT .....                   | 53 |
| 1.3.152. | DBMS_LOG_DEST .....                          | 54 |
| 1.3.153. | DBMS_LOG_FILE_SIZE .....                     | 54 |
| 1.3.154. | DBMS_LOG_TOTAL_SIZE_LIMIT .....              | 55 |
| 1.3.155. | DBWR_CNT .....                               | 55 |
| 1.3.156. | DBWR_USE_AIO .....                           | 55 |
| 1.3.157. | DB_BLOCK_SIZE .....                          | 56 |
| 1.3.158. | DB_CACHE_SIZE .....                          | 56 |
| 1.3.159. | DB_CREATE_FILE_DEST .....                    | 56 |
| 1.3.160. | DB_FILE_MULTIBLOCK_READ_COUNT .....          | 57 |
| 1.3.161. | DB_KEEP_CACHE_SIZE .....                     | 57 |
| 1.3.162. | DB_MULTIBLOCK_PREFETCH_COUNT .....           | 57 |
| 1.3.163. | DB_NAME .....                                | 58 |
| 1.3.164. | DB_RECYCLE_CACHE_SIZE .....                  | 58 |

|          |                                      |    |
|----------|--------------------------------------|----|
| 1.3.165. | DB_UNIQUE_NAME .....                 | 58 |
| 1.3.166. | DD_CACHE_MAX .....                   | 59 |
| 1.3.167. | DEFAULT_SEQ_CACHE_SIZE .....         | 59 |
| 1.3.168. | DFLT_DUMP_BACKUP_LIMIT_CNT .....     | 59 |
| 1.3.169. | DFLT_DUMP_TOTAL_SIZE_LIMIT .....     | 60 |
| 1.3.170. | DF_HDRBLK_START_OFFSET .....         | 60 |
| 1.3.171. | DIAG_DUMP_BACKUP_LIMIT_CNT .....     | 61 |
| 1.3.172. | DIAG_DUMP_DEST .....                 | 61 |
| 1.3.173. | DIAG_DUMP_TOTAL_SIZE_LIMIT .....     | 61 |
| 1.3.174. | DIAG_LOG_DEST .....                  | 62 |
| 1.3.175. | DIAG_LOG_FILE_SIZE .....             | 62 |
| 1.3.176. | DIAG_LOG_TOTAL_SIZE_LIMIT .....      | 62 |
| 1.3.177. | DIRTY_BLOCK_WRITE_ISSUE_TIME .....   | 63 |
| 1.3.178. | DISK_AFFINITY_GROUP .....            | 63 |
| 1.3.179. | DISK_PATH .....                      | 63 |
| 1.3.180. | DONT_USE_FILE_BUFFERING .....        | 64 |
| 1.3.181. | DPBUF_BLOCK_COUNT .....              | 64 |
| 1.3.182. | DPL_INDEX_METHOD .....               | 64 |
| 1.3.183. | DP_USE_O_SYNC .....                  | 65 |
| 1.3.184. | DUMP_EXSTAT_ON_STMT_CANCEL .....     | 65 |
| 1.3.185. | DUMP_PATH_CHECK_INTERVAL .....       | 65 |
| 1.3.186. | ENABLE_BCT_AT_BOOTING .....          | 66 |
| 1.3.187. | ENABLE_MULTI_LOG_BUFFER .....        | 66 |
| 1.3.188. | ENCRYPTION_WALLET_FILE .....         | 66 |
| 1.3.189. | EXTRA_LISTENER_IPS .....             | 67 |
| 1.3.190. | EXTRA_LISTENER_PORTS .....           | 67 |
| 1.3.191. | EXTRA_LISTENER_PORTS_6 .....         | 67 |
| 1.3.192. | EX_MEMORY_AUTO_MANAGEMENT .....      | 68 |
| 1.3.193. | EX_MEMORY_COMPENSATE_INTERVAL .....  | 68 |
| 1.3.194. | EX_MEMORY_DRIFT_CHECK_INTERVAL ..... | 68 |
| 1.3.195. | EX_MEMORY_DRIFT_RATIO .....          | 69 |
| 1.3.196. | EX_MEMORY_SOFT_LIMIT_RATIO .....     | 69 |
| 1.3.197. | FDPOOL_HASH_SIZE .....               | 69 |
| 1.3.198. | FLASHBACK_LOG_ARCHIVE_DEST .....     | 70 |
| 1.3.199. | FLASHBACK_LOG_ARCHIVE_FORMAT .....   | 70 |
| 1.3.200. | FLASHBACK_LOG_BUFFER .....           | 70 |
| 1.3.201. | FLASHBACK_MARKER_INTERVAL .....      | 71 |
| 1.3.202. | GATHER_SQL_EXEC_TIME .....           | 71 |
| 1.3.203. | GATHER_SQL_PLAN_STAT .....           | 72 |
| 1.3.204. | HASH_AREA_SIZE .....                 | 72 |
| 1.3.205. | HBASE_CONNECTOR_SCAN_CACHE .....     | 72 |
| 1.3.206. | HSM_LIB_FILE .....                   | 73 |
| 1.3.207. | HSM_NOT_USE_WRAPPING .....           | 73 |

|          |                                        |    |
|----------|----------------------------------------|----|
| 1.3.208. | HSM_TTE_KEY_BACKUP_FILE_PATH .....     | 73 |
| 1.3.209. | IFF_STORAGE_PROCESSING .....           | 74 |
| 1.3.210. | ILOG_BACKUP_SIZE_LIMIT .....           | 74 |
| 1.3.211. | ILOG_DEST .....                        | 74 |
| 1.3.212. | ILOG_FILE_BACKUP_ON_BOOT .....         | 75 |
| 1.3.213. | ILOG_FILE_SIZE .....                   | 75 |
| 1.3.214. | ILOG_KEEP_BACKUP_ASIDE .....           | 75 |
| 1.3.215. | ILOG_MAP .....                         | 76 |
| 1.3.216. | ILOG_MMAP_SIZE .....                   | 76 |
| 1.3.217. | ILOG_TOTAL_SIZE_LIMIT .....            | 77 |
| 1.3.218. | INMEMORY_QUERY .....                   | 77 |
| 1.3.219. | INMEMORY_SIZE .....                    | 77 |
| 1.3.220. | INSTANCE_NUMBER .....                  | 78 |
| 1.3.221. | INSTANCE_TYPE .....                    | 78 |
| 1.3.222. | IP_VERSION .....                       | 78 |
| 1.3.223. | JAVA_CLASS_PATH .....                  | 79 |
| 1.3.224. | JOB_HISTORY_LOG .....                  | 79 |
| 1.3.225. | JOB_SCHEDULING_MAX_TIMEOUT .....       | 79 |
| 1.3.226. | KEEP_STAT_PER_SQL .....                | 80 |
| 1.3.227. | LGWR_USE_AIO .....                     | 80 |
| 1.3.228. | LIBRARY_CACHE_MAX .....                | 80 |
| 1.3.229. | LIBRARY_DD_CACHE_MAX .....             | 81 |
| 1.3.230. | LISTENER_IP .....                      | 81 |
| 1.3.231. | LISTENER_IP_6 .....                    | 81 |
| 1.3.232. | LISTENER_PORT .....                    | 82 |
| 1.3.233. | LISTENER_PORT_6 .....                  | 82 |
| 1.3.234. | LISTENER_VIP_PORT .....                | 83 |
| 1.3.235. | LOAD_METRIC_AUTO_INTERVAL .....        | 83 |
| 1.3.236. | LOAD_METRIC_COLLECT_INTERVAL_MAX ..... | 83 |
| 1.3.237. | LOAD_METRIC_COLLECT_INTERVAL_MIN ..... | 84 |
| 1.3.238. | LOAD_METRIC_HISTORY_COUNT .....        | 84 |
| 1.3.239. | LOCAL_CLUSTER_ADDR .....               | 84 |
| 1.3.240. | LOCAL_CLUSTER_ADDR_IPV6 .....          | 85 |
| 1.3.241. | LOCAL_CLUSTER_IP_VERSION .....         | 85 |
| 1.3.242. | LOCAL_CLUSTER_PORT .....               | 85 |
| 1.3.243. | LOCAL_CLUSTER_PORT_IPV6 .....          | 86 |
| 1.3.244. | LOG_ARCHIVE_DEST .....                 | 86 |
| 1.3.245. | LOG_ARCHIVE_DEST_1 .....               | 86 |
| 1.3.246. | LOG_ARCHIVE_DEST_2 .....               | 87 |
| 1.3.247. | LOG_ARCHIVE_DEST_3 .....               | 87 |
| 1.3.248. | LOG_ARCHIVE_DEST_4 .....               | 87 |
| 1.3.249. | LOG_ARCHIVE_DEST_5 .....               | 88 |
| 1.3.250. | LOG_ARCHIVE_DEST_6 .....               | 88 |

|          |                                     |     |
|----------|-------------------------------------|-----|
| 1.3.251. | LOG_ARCHIVE_DEST_7 .....            | 88  |
| 1.3.252. | LOG_ARCHIVE_DEST_8 .....            | 89  |
| 1.3.253. | LOG_ARCHIVE_DEST_9 .....            | 89  |
| 1.3.254. | LOG_ARCHIVE_FORMAT .....            | 89  |
| 1.3.255. | LOG_ARCHIVE_OVERWRITE .....         | 90  |
| 1.3.256. | LOG_ARCHIVE_RETRY_TIME .....        | 90  |
| 1.3.257. | LOG_BUFFER .....                    | 90  |
| 1.3.258. | LOG_BUFFER_CNT .....                | 91  |
| 1.3.259. | LOG_CHECKPOINT_INTERVAL .....       | 91  |
| 1.3.260. | LOG_CHECKPOINT_TIMEOUT .....        | 91  |
| 1.3.261. | LOG_DEFAULT_DEST .....              | 92  |
| 1.3.262. | LOG_FILE_WITH_GROUP_READ_PERM ..... | 92  |
| 1.3.263. | LOG_LVL_AS .....                    | 92  |
| 1.3.264. | LOG_LVL_CM .....                    | 93  |
| 1.3.265. | LOG_LVL_LIS .....                   | 93  |
| 1.3.266. | LOG_ON_MEMORY .....                 | 93  |
| 1.3.267. | LOG_ON_MEMORY_SIZE .....            | 94  |
| 1.3.268. | LOG_REPLICATION_1_ENABLE .....      | 94  |
| 1.3.269. | LOG_REPLICATION_2_ENABLE .....      | 94  |
| 1.3.270. | LOG_REPLICATION_3_ENABLE .....      | 95  |
| 1.3.271. | LOG_REPLICATION_4_ENABLE .....      | 95  |
| 1.3.272. | LOG_REPLICATION_5_ENABLE .....      | 95  |
| 1.3.273. | LOG_REPLICATION_6_ENABLE .....      | 96  |
| 1.3.274. | LOG_REPLICATION_7_ENABLE .....      | 96  |
| 1.3.275. | LOG_REPLICATION_8_ENABLE .....      | 96  |
| 1.3.276. | LOG_REPLICATION_9_ENABLE .....      | 97  |
| 1.3.277. | LOG_REPLICATION_DEST_1 .....        | 97  |
| 1.3.278. | LOG_REPLICATION_DEST_2 .....        | 97  |
| 1.3.279. | LOG_REPLICATION_DEST_3 .....        | 98  |
| 1.3.280. | LOG_REPLICATION_DEST_4 .....        | 98  |
| 1.3.281. | LOG_REPLICATION_DEST_5 .....        | 98  |
| 1.3.282. | LOG_REPLICATION_DEST_6 .....        | 99  |
| 1.3.283. | LOG_REPLICATION_DEST_7 .....        | 99  |
| 1.3.284. | LOG_REPLICATION_DEST_8 .....        | 100 |
| 1.3.285. | LOG_REPLICATION_DEST_9 .....        | 100 |
| 1.3.286. | LOG_REPLICATION_MODE .....          | 100 |
| 1.3.287. | LONGOPS_KEEP_INTERVAL .....         | 101 |
| 1.3.288. | LONGOPS_THRESHOLD_SEC .....         | 101 |
| 1.3.289. | LONG_OPERATION_KEEP_COUNT .....     | 101 |
| 1.3.290. | LSNR_DENIED_IP .....                | 102 |
| 1.3.291. | LSNR_DENIED_IP_FILE .....           | 102 |
| 1.3.292. | LSNR_INVITED_IP .....               | 102 |
| 1.3.293. | LSNR_INVITED_IP_FILE .....          | 103 |

|          |                                     |     |
|----------|-------------------------------------|-----|
| 1.3.294. | LSNR_LISTEN_BACKLOG .....           | 103 |
| 1.3.295. | LSNR_LOG_DEST .....                 | 103 |
| 1.3.296. | LSNR_LOG_FILE_SIZE .....            | 104 |
| 1.3.297. | LSNR_LOG_TOTAL_SIZE_LIMIT .....     | 104 |
| 1.3.298. | LSNR_OUTFILE_BACKUP_LIMIT_CNT ..... | 105 |
| 1.3.299. | LSNR_OUTFILE_DEST .....             | 105 |
| 1.3.300. | LSNR_SCHEDULE_SAME_PROC_FIRST ..... | 105 |
| 1.3.301. | MAX_BG_SESSION_COUNT .....          | 106 |
| 1.3.302. | MAX_FG_SESSION_COUNT .....          | 106 |
| 1.3.303. | MAX_SESSION_COUNT .....             | 106 |
| 1.3.304. | MEMORY_MAX_TARGET .....             | 107 |
| 1.3.305. | MEMORY_MODEL .....                  | 107 |
| 1.3.306. | MEMORY_TABLESPACE_SIZE .....        | 108 |
| 1.3.307. | MEMORY_TARGET .....                 | 108 |
| 1.3.308. | MEM_LOG_BACKUP_LIMIT_CNT .....      | 108 |
| 1.3.309. | MEM_LOG_DEST .....                  | 109 |
| 1.3.310. | MEM_LOG_TOTAL_SIZE_LIMIT .....      | 109 |
| 1.3.311. | NAMED_PIPE_DIR .....                | 109 |
| 1.3.312. | NBU_ARCHIVELOG_SEARCH .....         | 110 |
| 1.3.313. | NBU_ARCHIVE_POLICY_NAME .....       | 110 |
| 1.3.314. | NBU_ARCHIVE_RETRY_CNT .....         | 110 |
| 1.3.315. | NBU_BACKUP_INST_SID .....           | 111 |
| 1.3.316. | NBU_BACKUP_POLICY_NAME .....        | 111 |
| 1.3.317. | NBU_BSA_API_VERSION .....           | 111 |
| 1.3.318. | NBU_BSA_LOG_DIRECTORY .....         | 112 |
| 1.3.319. | NBU_BSA_LOG_LEVEL .....             | 112 |
| 1.3.320. | NBU_CLIENT_COUNT .....              | 112 |
| 1.3.321. | NBU_CLIENT_HOSTNAME_0 .....         | 113 |
| 1.3.322. | NBU_CLIENT_HOSTNAME_1 .....         | 113 |
| 1.3.323. | NBU_CLIENT_HOSTNAME_2 .....         | 113 |
| 1.3.324. | NBU_CLIENT_HOSTNAME_3 .....         | 114 |
| 1.3.325. | NBU_CLIENT_HOSTNAME_4 .....         | 114 |
| 1.3.326. | NBU_CLIENT_HOSTNAME_5 .....         | 114 |
| 1.3.327. | NBU_CLIENT_HOSTNAME_6 .....         | 115 |
| 1.3.328. | NBU_CLIENT_HOSTNAME_7 .....         | 115 |
| 1.3.329. | NBU_CLIENT_HOSTNAME_8 .....         | 115 |
| 1.3.330. | NBU_CLIENT_HOSTNAME_9 .....         | 116 |
| 1.3.331. | NBU_CLIENT_HOST_NAME .....          | 116 |
| 1.3.332. | NBU_MASTER_HOST_NAME .....          | 116 |
| 1.3.333. | NBU_OBJ_OWNER_GROUP_NAME .....      | 117 |
| 1.3.334. | NBU_OBJ_OWNER_NAME .....            | 117 |
| 1.3.335. | NBU_PROC_WTHR_CNT .....             | 117 |
| 1.3.336. | NBU_SCHEDULE_NAME .....             | 118 |

|          |                                                     |     |
|----------|-----------------------------------------------------|-----|
| 1.3.337. | NBU_SKIP_HOST_CHECK .....                           | 118 |
| 1.3.338. | NBU_TWO_PHASE_CF_BACKUP .....                       | 118 |
| 1.3.339. | NETWORK_SECURITY_AGENT_MANAGER_PATH .....           | 119 |
| 1.3.340. | NLS_CALENDAR .....                                  | 119 |
| 1.3.341. | NLS_COMP .....                                      | 119 |
| 1.3.342. | NLS_CURRENCY .....                                  | 120 |
| 1.3.343. | NLS_DATE_FORMAT .....                               | 120 |
| 1.3.344. | NLS_DATE_LANGUAGE .....                             | 120 |
| 1.3.345. | NLS_DUAL_CURRENCY .....                             | 121 |
| 1.3.346. | NLS_ISO_CURRENCY .....                              | 121 |
| 1.3.347. | NLS_LANGUAGE .....                                  | 121 |
| 1.3.348. | NLS_LANG_AT_BOOT .....                              | 122 |
| 1.3.349. | NLS_LENGTH_SEMANTICS .....                          | 122 |
| 1.3.350. | NLS_NUMERIC_CHARACTERS .....                        | 122 |
| 1.3.351. | NLS_SORT .....                                      | 123 |
| 1.3.352. | NLS_TERRITORY .....                                 | 123 |
| 1.3.353. | NLS_TIMESTAMP_FORMAT .....                          | 124 |
| 1.3.354. | NLS_TIMESTAMP_TZ_FORMAT .....                       | 124 |
| 1.3.355. | NLS_TIME_FORMAT .....                               | 124 |
| 1.3.356. | NLS_TIME_TZ_FORMAT .....                            | 125 |
| 1.3.357. | NTF_LISTEN_PORT .....                               | 125 |
| 1.3.358. | OMIT_PSM_SQL_LOG_ON_MEMORY .....                    | 125 |
| 1.3.359. | OPEN_CURSORS .....                                  | 126 |
| 1.3.360. | OPTIMIZER_CAPTURE_SQL_PLAN_BASELINES .....          | 126 |
| 1.3.361. | OPTIMIZER_CAPTURE_SQL_PLAN_BASELINES_DURATION ..... | 126 |
| 1.3.362. | OPTIMIZER_LOG_OUTLINE .....                         | 127 |
| 1.3.363. | OPTIMIZER_MODE .....                                | 127 |
| 1.3.364. | OPTIMIZER_TRACE .....                               | 127 |
| 1.3.365. | OPTIMIZER_USE_INVISIBLE_INDEXES .....               | 128 |
| 1.3.366. | OPTIMIZER_USE_PENDING_STATISTICS .....              | 128 |
| 1.3.367. | OPTIMIZER_USE_SQL_PLAN_BASELINES .....              | 128 |
| 1.3.368. | OPTIMIZER_USE_SQL_PROFILES .....                    | 129 |
| 1.3.369. | OS_AUTH_PREFIX .....                                | 129 |
| 1.3.370. | OUTFILE_PATH_BACKUP_LIMIT_CNT .....                 | 129 |
| 1.3.371. | PARALLEL_DEFAULT_DOP .....                          | 130 |
| 1.3.372. | PARALLEL_GRANULE_SIZE .....                         | 130 |
| 1.3.373. | PARALLEL_THREADS_PER_CPU .....                      | 130 |
| 1.3.374. | PEP_PROC_CNT .....                                  | 131 |
| 1.3.375. | PP_SAVE_SYS_BINDVAR .....                           | 131 |
| 1.3.376. | PRIVKEY_FILE .....                                  | 131 |
| 1.3.377. | PSM_CODE_TYPE .....                                 | 132 |
| 1.3.378. | PSM_DEBUG .....                                     | 132 |
| 1.3.379. | PSM_ERROR_CONV .....                                | 132 |

|          |                                                         |     |
|----------|---------------------------------------------------------|-----|
| 1.3.380. | PSM_LOG_ON_THROW .....                                  | 133 |
| 1.3.381. | PSM_SHLIB_DIR .....                                     | 133 |
| 1.3.382. | PSM_SQL_RESULT_CACHE .....                              | 133 |
| 1.3.383. | QUERY_REWRITE_ENABLED .....                             | 134 |
| 1.3.384. | QUERY_REWRITE_INTEGRITY .....                           | 134 |
| 1.3.385. | RECO_PROC_WTHR_CNT .....                                | 134 |
| 1.3.386. | RESULT_CACHE_MAX_RATIO .....                            | 135 |
| 1.3.387. | RESULT_CACHE_MAX_SIZE .....                             | 135 |
| 1.3.388. | RESULT_CACHE_MODE .....                                 | 135 |
| 1.3.389. | RESUMABLE_SPACE_ALLOC .....                             | 136 |
| 1.3.390. | RESUMABLE_SPACE_ALLOC_TIMEOUT .....                     | 136 |
| 1.3.391. | RMGR_BACKUP_DEST .....                                  | 136 |
| 1.3.392. | RMGR_CATALOG_SID .....                                  | 137 |
| 1.3.393. | RMGR_CHECK_FREE_DISK_SIZE .....                         | 137 |
| 1.3.394. | RMGR_NBU_BACKUP_DEST .....                              | 137 |
| 1.3.395. | RMGR_PRIMARY_SID .....                                  | 138 |
| 1.3.396. | RMGR_RESTORE_BACKUP_CF .....                            | 138 |
| 1.3.397. | RMGR_TAR_PATH .....                                     | 138 |
| 1.3.398. | RSBTBL_BUCKET_CNT_PER_ROOT_LIST_FOR_RECONFIGURATION ... | 139 |
| 1.3.399. | RSRC_CHECK_INTERVAL .....                               | 139 |
| 1.3.400. | RSRC_CONTROL_THRESHOLD .....                            | 139 |
| 1.3.401. | RT_AWAKE_WAITER_INTERVAL .....                          | 140 |
| 1.3.402. | RT_UPDATE_ALLOWABLE_INFLOW_CNT_INTERVAL .....           | 140 |
| 1.3.403. | SANITY_CHECK_DURING_LOG_SWITCH .....                    | 141 |
| 1.3.404. | SECU_CHANGE_PASSWORD_WITH_CASE_SENSITIVITY .....        | 141 |
| 1.3.405. | SEGMENT_EXTEND_LAST .....                               | 141 |
| 1.3.406. | SEM_KEY .....                                           | 142 |
| 1.3.407. | SEM_SET_COUNT .....                                     | 142 |
| 1.3.408. | SEQUENCE_PIN_BUCKET_CNT .....                           | 142 |
| 1.3.409. | SERVER_LOAD_TOLERANCE .....                             | 143 |
| 1.3.410. | SERVER_SIDE_LOAD_BALANCE .....                          | 143 |
| 1.3.411. | SERVICE_CONTROL_STOP_DOWNMODE .....                     | 143 |
| 1.3.412. | SESS_IGNORE_INVALID_CLIENT_NLS_SETTING .....            | 144 |
| 1.3.413. | SHARED_PARAMETER_FILE .....                             | 144 |
| 1.3.414. | SHM_KEY .....                                           | 145 |
| 1.3.415. | SHP_ALLOC_DUMP_IDX .....                                | 145 |
| 1.3.416. | SHP_ALLOC_FULL_DUMP .....                               | 145 |
| 1.3.417. | SLAB_BLOCK_RESERVED_COUNT .....                         | 146 |
| 1.3.418. | SLOG_DEST .....                                         | 146 |
| 1.3.419. | SLOG_DISPLAY_YEAR .....                                 | 146 |
| 1.3.420. | SLOG_FILE_SIZE .....                                    | 147 |
| 1.3.421. | SLOG_HISTORY .....                                      | 147 |
| 1.3.422. | SLOG_HISTORY_BUFSIZE .....                              | 147 |

|          |                                     |     |
|----------|-------------------------------------|-----|
| 1.3.423. | SLOG_HISTORY_QSIZE .....            | 148 |
| 1.3.424. | SLOG_TOTAL_SIZE_LIMIT .....         | 148 |
| 1.3.425. | SLOG_VIEW_SIZE .....                | 148 |
| 1.3.426. | SMR_DELAY_TIME .....                | 149 |
| 1.3.427. | SMR_SGMT_DROP_TIMEOUT .....         | 149 |
| 1.3.428. | SORT_AREA_SIZE .....                | 149 |
| 1.3.429. | SQLTUNE_CATEGORY .....              | 150 |
| 1.3.430. | SQL_LOG_ON_MEMORY .....             | 150 |
| 1.3.431. | SQL_LOG_ON_MEMORY_SIZE .....        | 150 |
| 1.3.432. | SQL_STAT_HISTORY .....              | 151 |
| 1.3.433. | SQL_STAT_HISTORY_QSIZE .....        | 151 |
| 1.3.434. | SQL_STAT_HISTORY_THRESHOLD .....    | 151 |
| 1.3.435. | SQL_TRACE .....                     | 152 |
| 1.3.436. | SQL_TRACE_BACKUP_LIMIT_CNT .....    | 152 |
| 1.3.437. | SQL_TRACE_DEST .....                | 153 |
| 1.3.438. | SQL_TRACE_LEVEL .....               | 153 |
| 1.3.439. | SQL_TRACE_TOTAL_SIZE_LIMIT .....    | 153 |
| 1.3.440. | SQL_TRANSLATION_PROFILE .....       | 154 |
| 1.3.441. | SSVR_FC_DIRTY_BIN_RATIO_MAX .....   | 154 |
| 1.3.442. | SSVR_FC_FREE_BIN_RATIO .....        | 154 |
| 1.3.443. | SSVR_FC_WS_CNT .....                | 155 |
| 1.3.444. | SSVR_RECV_PORT_START .....          | 155 |
| 1.3.445. | SSVR_TPM_SENDER_INTERVAL .....      | 155 |
| 1.3.446. | SSVR_USE_AGNT .....                 | 156 |
| 1.3.447. | SSVR_USE_FC .....                   | 156 |
| 1.3.448. | SSVR_USE_IB .....                   | 156 |
| 1.3.449. | SSVR_USE_SDM .....                  | 157 |
| 1.3.450. | SSVR_USE_TCP .....                  | 157 |
| 1.3.451. | SSVR_USE_TPM .....                  | 157 |
| 1.3.452. | SSVR_WTHR_CNT .....                 | 158 |
| 1.3.453. | STANDBY_ALLOWED_IP_ADDRESSES .....  | 158 |
| 1.3.454. | STANDBY_ENABLE_CASCADING .....      | 158 |
| 1.3.455. | STANDBY_ENABLE_LOG_RECOVERY .....   | 159 |
| 1.3.456. | STANDBY_ENABLE_TAC_MULTINODE .....  | 159 |
| 1.3.457. | STANDBY_FILE_NAME_CONVERT .....     | 159 |
| 1.3.458. | STANDBY_FLUSH_WAIT .....            | 160 |
| 1.3.459. | STANDBY_LOG_ARCHIVE_FORMAT .....    | 160 |
| 1.3.460. | STANDBY_USE_OBSERVER .....          | 160 |
| 1.3.461. | STAR_TRANSFORMATION_ENABLED .....   | 161 |
| 1.3.462. | STORAGE_PROCESSING .....            | 161 |
| 1.3.463. | SUMMARY_FIRST_IN_GROUPBY_CUBE ..... | 161 |
| 1.3.464. | SVR_OUTFILE_BACKUP_LIMIT_CNT .....  | 162 |
| 1.3.465. | SVR_OUTFILE_DEST .....              | 162 |

|          |                                            |     |
|----------|--------------------------------------------|-----|
| 1.3.466. | TABLE_GLOBAL_COMPRESSION .....             | 162 |
| 1.3.467. | TAS5_STORAGE_MODE .....                    | 163 |
| 1.3.468. | TAS_DB_UUID .....                          | 163 |
| 1.3.469. | TAS_HOT_BACKUP_SYNC_WITH_TAC .....         | 163 |
| 1.3.470. | TAS_OPEN_FILE_MAX .....                    | 164 |
| 1.3.471. | TAS_SERVER_IP .....                        | 164 |
| 1.3.472. | TAS_VM_ADDR .....                          | 164 |
| 1.3.473. | TAS_VM_PORT .....                          | 165 |
| 1.3.474. | TCP_RCVBUF_SIZE .....                      | 165 |
| 1.3.475. | TCP_SNDBUF_SIZE .....                      | 165 |
| 1.3.476. | TDE_HASH_VERIFICATION .....                | 165 |
| 1.3.477. | TDE_SELF_UPDATE_MODIFICATION_DEFENCE ..... | 166 |
| 1.3.478. | TEMP_GRANULE_SIZE .....                    | 166 |
| 1.3.479. | TEMP_TS_CNT_MAX .....                      | 166 |
| 1.3.480. | THREAD .....                               | 167 |
| 1.3.481. | THROW_WHEN_GETTING_OSSTAT_FAIL .....       | 167 |
| 1.3.482. | TIBERO_PERFORMANCE_REPOSITORY .....        | 167 |
| 1.3.483. | TOTAL_SHM_MAX_SIZE .....                   | 168 |
| 1.3.484. | TOTAL_SHM_SIZE .....                       | 168 |
| 1.3.485. | TPR_AGGREGATION .....                      | 168 |
| 1.3.486. | TPR_METRIC .....                           | 169 |
| 1.3.487. | TPR_PARTITION_COUNT_IN_ADVANCE .....       | 169 |
| 1.3.488. | TPR_SEGMENT_STATISTICS .....               | 169 |
| 1.3.489. | TPR_SNAPSHOT_RETENTION .....               | 170 |
| 1.3.490. | TPR_SNAPSHOT_SAMPLING_INTERVAL .....       | 170 |
| 1.3.491. | TPR_SNAPSHOT_TOP_SEGMENT_CNT .....         | 170 |
| 1.3.492. | TPR_SNAPSHOT_TOP_SQL_CNT .....             | 171 |
| 1.3.493. | TPR_SQL_PLAN_STAT .....                    | 171 |
| 1.3.494. | TRACE_BACKUP_LIMIT_CNT .....               | 171 |
| 1.3.495. | TRACE_DUMP_BACKUP_LIMIT_CNT .....          | 172 |
| 1.3.496. | TRACE_DUMP_DEST .....                      | 172 |
| 1.3.497. | TRACE_DUMP_TOTAL_SIZE_LIMIT .....          | 173 |
| 1.3.498. | TRACE_PATH_CHECK_INTERVAL .....            | 173 |
| 1.3.499. | TRACE_TOTAL_SIZE_LIMIT .....               | 173 |
| 1.3.500. | TSAM_PREFETCH_SIZE .....                   | 174 |
| 1.3.501. | TSN_SYNC_UDP_PORT .....                    | 174 |
| 1.3.502. | TX_FORCE_COMMIT_DUMP .....                 | 174 |
| 1.3.503. | UNDO_RETENTION .....                       | 175 |
| 1.3.504. | UNDO_TABLESPACE .....                      | 175 |
| 1.3.505. | USE_ACCURATE_NUMERIC_FUNC .....            | 175 |
| 1.3.506. | USE_ACTIVE_STORAGE .....                   | 176 |
| 1.3.507. | USE_AD_AUTH .....                          | 176 |
| 1.3.508. | USE_BINARY_UPDATE .....                    | 176 |

|            |                                          |            |
|------------|------------------------------------------|------------|
| 1.3.509.   | USE_CASE_SENSITIVE_PASSWORD .....        | 177        |
| 1.3.510.   | USE_DISK_AFFINITY .....                  | 177        |
| 1.3.511.   | USE_DUMP_PATH_MANAGEMENT .....           | 177        |
| 1.3.512.   | USE_HSM .....                            | 178        |
| 1.3.513.   | USE_HUGE_PAGE .....                      | 178        |
| 1.3.514.   | USE_ILOG .....                           | 178        |
| 1.3.515.   | USE_NBU_FOR_ARCHIVELOG .....             | 179        |
| 1.3.516.   | USE_NBU_FOR_BACKUPSET .....              | 179        |
| 1.3.517.   | USE_NETWORK_SECURITY_AGENT_MANAGER ..... | 179        |
| 1.3.518.   | USE_NET_KEEPALIVE .....                  | 180        |
| 1.3.519.   | USE_OUTFILE_PATH_MANAGEMENT .....        | 180        |
| 1.3.520.   | USE_PROFILE .....                        | 180        |
| 1.3.521.   | USE_PROFILE_LOGIN_LOGOUT_TIME .....      | 181        |
| 1.3.522.   | USE_RECYCLEBIN .....                     | 181        |
| 1.3.523.   | USE_RECYCLEBIN_AUTO_PURGE .....          | 181        |
| 1.3.524.   | USE_RESOURCE_MANAGER .....               | 182        |
| 1.3.525.   | USE_SERVER_EVENT_NOTIFICATION .....      | 182        |
| 1.3.526.   | USE_SQL_STAT_HIST_IN_BG_PROC .....       | 182        |
| 1.3.527.   | USE_STANDBY_REDO_LOG .....               | 183        |
| 1.3.528.   | USE_STORED_OUTLINES .....                | 183        |
| 1.3.529.   | USE_TRACE_PATH_MANAGEMENT .....          | 183        |
| 1.3.530.   | USE_TRUNCATE_OBJ_PRIVILEGE .....         | 184        |
| 1.3.531.   | USE_TRUNCATE_PRIVILEGE .....             | 184        |
| 1.3.532.   | USE_TS_QUOTA .....                       | 184        |
| 1.3.533.   | USE_ZETA .....                           | 185        |
| 1.3.534.   | USGMT_AUTO_SHRINK_INTERVAL .....         | 185        |
| 1.3.535.   | USGMT_ONLINE_MAX .....                   | 185        |
| 1.3.536.   | USGMT_ONLINE_MIN .....                   | 186        |
| 1.3.537.   | USGMT_STEAL_IGNORE_RETENTION .....       | 186        |
| 1.3.538.   | UTL_FILE_DIR .....                       | 187        |
| 1.3.539.   | WALLET_FILE .....                        | 187        |
| 1.3.540.   | WTHR_PER_PEP .....                       | 187        |
| 1.3.541.   | WTHR_PER_PROC .....                      | 188        |
| 1.3.542.   | WTHR_PROC_CNT .....                      | 188        |
| 1.3.543.   | WTHR_PROC_CNT_MAX .....                  | 188        |
| 1.3.544.   | XA_MAX_BRANCH_CNT .....                  | 189        |
| 1.3.545.   | XA_TIMER_INTERVAL .....                  | 189        |
| <b>제2장</b> | <b>데이터 사전 .....</b>                      | <b>191</b> |
| 2.1.       | 개요 .....                                 | 191        |
| 2.2.       | 데이터 사전의 구조 .....                         | 192        |
| 2.2.1.     | 기반 테이블 .....                             | 192        |
| 2.2.2.     | 정적/동적 뷰 .....                            | 193        |

|            |                                    |            |
|------------|------------------------------------|------------|
| 2.3.       | 데이터 사전의 참조 .....                   | 193        |
| 2.3.1.     | 일반 사용자에게 의한 참조 .....               | 193        |
| 2.3.2.     | SQL 파서, 질의 최적화기에 의한 참조 .....       | 194        |
| 2.3.3.     | Tibero에 의한 참조 .....                | 195        |
| 2.4.       | 데이터 사전의 갱신 .....                   | 196        |
| 2.4.1.     | DDL 문장 .....                       | 196        |
| <b>제3장</b> | <b>Static View .....</b>           | <b>199</b> |
| 3.1.       | Overview .....                     | 199        |
| 3.2.       | List of Static Views .....         | 199        |
| 3.2.1.     | ALL_ARGUMENTS .....                | 215        |
| 3.2.2.     | ALL_CATALOG .....                  | 216        |
| 3.2.3.     | ALL_COLL_TYPES .....               | 217        |
| 3.2.4.     | ALL_COL_COMMENTS .....             | 218        |
| 3.2.5.     | ALL_COL_PENDING_STATS .....        | 218        |
| 3.2.6.     | ALL_COL_PRIVS .....                | 219        |
| 3.2.7.     | ALL_COL_PRIVS_MADE .....           | 220        |
| 3.2.8.     | ALL_COL_PRIVS_RECD .....           | 220        |
| 3.2.9.     | ALL_CONSTRAINTS .....              | 221        |
| 3.2.10.    | ALL_CONS_COLUMNS .....             | 223        |
| 3.2.11.    | ALL_CONS_COLUMNS_INFO .....        | 223        |
| 3.2.12.    | ALL_DB_LINKS .....                 | 224        |
| 3.2.13.    | ALL_DEPENDENCIES .....             | 225        |
| 3.2.14.    | ALL_DIRECTORIES .....              | 226        |
| 3.2.15.    | ALL_ENCRYPTED_COLUMNS .....        | 227        |
| 3.2.16.    | ALL_ERRORS .....                   | 227        |
| 3.2.17.    | ALL_EXTERNAL_LOCATIONS .....       | 228        |
| 3.2.18.    | ALL_EXTERNAL_TABLES .....          | 228        |
| 3.2.19.    | ALL_IDX_AND_LOB_COLUMNS_INFO ..... | 229        |
| 3.2.20.    | ALL_IDX_COLUMNS .....              | 229        |
| 3.2.21.    | ALL_IDX_COLUMNS_INFO .....         | 230        |
| 3.2.22.    | ALL_IDX_EXPRESSIONS .....          | 231        |
| 3.2.23.    | ALL_IDX_PARTITIONS .....           | 232        |
| 3.2.24.    | ALL_IDX_SUBPARTITIONS .....        | 233        |
| 3.2.25.    | ALL_INDEXES .....                  | 235        |
| 3.2.26.    | ALL_IND_PENDING_STATS .....        | 237        |
| 3.2.27.    | ALL_JAVA_ARGUMENTS .....           | 238        |
| 3.2.28.    | ALL_JAVA_CLASSES .....             | 239        |
| 3.2.29.    | ALL_JAVA_FIELDS .....              | 240        |
| 3.2.30.    | ALL_JAVA_METHODS .....             | 241        |
| 3.2.31.    | ALL_JOBS .....                     | 242        |
| 3.2.32.    | ALL_JOBS_WITH_NAME .....           | 243        |
| 3.2.33.    | ALL_LIBRARIES .....                | 244        |

|         |                                     |     |
|---------|-------------------------------------|-----|
| 3.2.34. | ALL_LOBS .....                      | 244 |
| 3.2.35. | ALL_LOB_PARTITIONS .....            | 246 |
| 3.2.36. | ALL_LOB_SUBPARTITIONS .....         | 247 |
| 3.2.37. | ALL_LOG_GROUPS .....                | 249 |
| 3.2.38. | ALL_LOG_GROUP_COLUMNS .....         | 250 |
| 3.2.39. | ALL_METHOD_PARAMS .....             | 250 |
| 3.2.40. | ALL_METHOD_RESULTS .....            | 251 |
| 3.2.41. | ALL_MVIEWS .....                    | 252 |
| 3.2.42. | ALL_MVIEW_COMMENTS .....            | 255 |
| 3.2.43. | ALL_MVIEW_DETAIL_RELATIONS .....    | 255 |
| 3.2.44. | ALL_MVIEW_LOGS .....                | 256 |
| 3.2.45. | ALL_MVIEW_REFRESH_TIMES .....       | 256 |
| 3.2.46. | ALL_OBJECTS .....                   | 257 |
| 3.2.47. | ALL_PACKAGE_COLL_TYPES .....        | 259 |
| 3.2.48. | ALL_PACKAGE_MEMBER_TYPES .....      | 259 |
| 3.2.49. | ALL_PACKAGE_TYPES .....             | 261 |
| 3.2.50. | ALL_PART_INDEXES .....              | 261 |
| 3.2.51. | ALL_PART_KEY_COLUMNS .....          | 263 |
| 3.2.52. | ALL_PART_LOBS .....                 | 263 |
| 3.2.53. | ALL_PART_TABLES .....               | 264 |
| 3.2.54. | ALL_PROCEDURES .....                | 266 |
| 3.2.55. | ALL_QUEUES .....                    | 267 |
| 3.2.56. | ALL_QUEUE_SUBSCRIBERS .....         | 268 |
| 3.2.57. | ALL_QUEUE_TABLES .....              | 269 |
| 3.2.58. | ALL_RECYCLEBIN .....                | 270 |
| 3.2.59. | ALL_SCHEDULER_CHAINS .....          | 271 |
| 3.2.60. | ALL_SCHEDULER_JOBS .....            | 272 |
| 3.2.61. | ALL_SCHEDULER_JOB_LOG .....         | 275 |
| 3.2.62. | ALL_SCHEDULER_JOB_RUN_DETAILS ..... | 276 |
| 3.2.63. | ALL_SCHEDULER_PROGRAMS .....        | 277 |
| 3.2.64. | ALL_SCHEDULER_RULES .....           | 278 |
| 3.2.65. | ALL_SCHEDULER_RULE_CTX .....        | 279 |
| 3.2.66. | ALL_SCHEDULER_RUNNING_JOBS .....    | 279 |
| 3.2.67. | ALL_SCHEDULER_SCHEDULES .....       | 283 |
| 3.2.68. | ALL_SCHEDULER_STEPS .....           | 284 |
| 3.2.69. | ALL_SCHEDULER_STEP_CTX .....        | 284 |
| 3.2.70. | ALL_SEGMENTS .....                  | 285 |
| 3.2.71. | ALL_SEQUENCES .....                 | 287 |
| 3.2.72. | ALL_SOURCE .....                    | 288 |
| 3.2.73. | ALL_STAT_EXTENSIONS .....           | 289 |
| 3.2.74. | ALL_SUBPARTITION_TEMPLATES .....    | 289 |
| 3.2.75. | ALL_SUBPART_KEY_COLUMNS .....       | 290 |
| 3.2.76. | ALL_SYNONYMS .....                  | 290 |

|          |                                     |     |
|----------|-------------------------------------|-----|
| 3.2.77.  | ALL_TABLES .....                    | 291 |
| 3.2.78.  | ALL_TAB_COMMENTS .....              | 295 |
| 3.2.79.  | ALL_TAB_HISTGRM_PENDING_STATS ..... | 295 |
| 3.2.80.  | ALL_TAB_IDENTITY_COLS .....         | 296 |
| 3.2.81.  | ALL_TAB_PENDING_STATS .....         | 296 |
| 3.2.82.  | ALL_TBL_COLS .....                  | 297 |
| 3.2.83.  | ALL_TBL_COLUMNS .....               | 299 |
| 3.2.84.  | ALL_TBL_COLUMNS_WITH_DNO .....      | 300 |
| 3.2.85.  | ALL_TBL_COL_STATISTICS .....        | 301 |
| 3.2.86.  | ALL_TBL_HISTOGRAMS .....            | 302 |
| 3.2.87.  | ALL_TBL_PARTITIONS .....            | 303 |
| 3.2.88.  | ALL_TBL_PRIVS .....                 | 305 |
| 3.2.89.  | ALL_TBL_PRIVS_MADE .....            | 306 |
| 3.2.90.  | ALL_TBL_PRIVS_RECD .....            | 307 |
| 3.2.91.  | ALL_TBL_SUBPARTITIONS .....         | 307 |
| 3.2.92.  | ALL_TRIGGERS .....                  | 310 |
| 3.2.93.  | ALL_TRIGGER_COLS .....              | 311 |
| 3.2.94.  | ALL_TYPES .....                     | 312 |
| 3.2.95.  | ALL_TYPE_ATTRS .....                | 313 |
| 3.2.96.  | ALL_TYPE_METHODS .....              | 313 |
| 3.2.97.  | ALL_UNUSED_COL_TABS .....           | 314 |
| 3.2.98.  | ALL_UPDATABLE_COLUMNS .....         | 315 |
| 3.2.99.  | ALL_USERS .....                     | 316 |
| 3.2.100. | ALL_VIEWS .....                     | 316 |
| 3.2.101. | ALL_VIEWS_INTERNAL .....            | 316 |
| 3.2.102. | CTX_PREFERENCES .....               | 317 |
| 3.2.103. | CTX_PREFERENCE_VALUES .....         | 317 |
| 3.2.104. | CTX_STOPLISTS .....                 | 318 |
| 3.2.105. | DATABASE_PROPERTIES .....           | 318 |
| 3.2.106. | DBA_2PC_NEIGHBORS .....             | 318 |
| 3.2.107. | DBA_2PC_PENDING .....               | 319 |
| 3.2.108. | DBA_AUDIT_POLICIES .....            | 319 |
| 3.2.109. | DBA_AUDIT_TRAIL .....               | 321 |
| 3.2.110. | DBA_CATALOG .....                   | 322 |
| 3.2.111. | DBA_COLL_TYPES .....                | 322 |
| 3.2.112. | DBA_COL_COMMENTS .....              | 322 |
| 3.2.113. | DBA_COL_PENDING_STATS .....         | 322 |
| 3.2.114. | DBA_COL_PRIVS .....                 | 323 |
| 3.2.115. | DBA_CONSTRAINTS .....               | 323 |
| 3.2.116. | DBA_CONS_COLUMNS .....              | 323 |
| 3.2.117. | DBA_DATA_FILES .....                | 323 |
| 3.2.118. | DBA_DB_LINKS .....                  | 324 |
| 3.2.119. | DBA_DEPENDENCIES .....              | 325 |

|          |                                    |     |
|----------|------------------------------------|-----|
| 3.2.120. | DBA_DIRECTORIES .....              | 325 |
| 3.2.121. | DBA_ENCRYPTED_COLUMNS .....        | 325 |
| 3.2.122. | DBA_ERRORS .....                   | 325 |
| 3.2.123. | DBA_EXTENTS .....                  | 325 |
| 3.2.124. | DBA_EXTERNAL_LOCATIONS .....       | 326 |
| 3.2.125. | DBA_EXTERNAL_TABLES .....          | 326 |
| 3.2.126. | DBA_FGA_AUDIT_TRAIL .....          | 327 |
| 3.2.127. | DBA_FREE_SPACE .....               | 327 |
| 3.2.128. | DBA_FREE_TABLESPACE .....          | 327 |
| 3.2.129. | DBA_HIST_ACTIVE_SESS_HISTORY ..... | 328 |
| 3.2.130. | DBA_HIST_FILESTAT .....            | 330 |
| 3.2.131. | DBA_HIST_LATCH .....               | 331 |
| 3.2.132. | DBA_HIST_LIBRARYCACHE .....        | 331 |
| 3.2.133. | DBA_HIST_OSSTAT .....              | 332 |
| 3.2.134. | DBA_HIST_PGASTAT .....             | 332 |
| 3.2.135. | DBA_HIST_PROCESS_MEM_SUMMARY ..... | 333 |
| 3.2.136. | DBA_HIST_SEG_STAT .....            | 333 |
| 3.2.137. | DBA_HIST_SGASTAT .....             | 334 |
| 3.2.138. | DBA_HIST_SNAPSHOT .....            | 334 |
| 3.2.139. | DBA_HIST_SQLBIND .....             | 334 |
| 3.2.140. | DBA_HIST_SQLSTAT .....             | 335 |
| 3.2.141. | DBA_HIST_SQLTEXT .....             | 336 |
| 3.2.142. | DBA_HIST_SQL_PLAN .....            | 337 |
| 3.2.143. | DBA_HIST_SQL_WORKAREA_HSTGRM ..... | 338 |
| 3.2.144. | DBA_HIST_SYSMETRIC_SUMMARY .....   | 339 |
| 3.2.145. | DBA_HIST_SYSTEM_EVENT .....        | 340 |
| 3.2.146. | DBA_HIST_UNDOSTAT .....            | 340 |
| 3.2.147. | DBA_HIST_WAITSTAT .....            | 341 |
| 3.2.148. | DBA_IDX_COLUMNS .....              | 341 |
| 3.2.149. | DBA_IDX_EXPRESSIONS .....          | 342 |
| 3.2.150. | DBA_IDX_PARTITIONS .....           | 342 |
| 3.2.151. | DBA_IDX_SUBPARTITIONS .....        | 342 |
| 3.2.152. | DBA_INDEXES .....                  | 342 |
| 3.2.153. | DBA_IND_PENDING_STATS .....        | 342 |
| 3.2.154. | DBA_JAVA_ARGUMENTS .....           | 343 |
| 3.2.155. | DBA_JAVA_CLASSES .....             | 343 |
| 3.2.156. | DBA_JAVA_FIELDS .....              | 343 |
| 3.2.157. | DBA_JAVA_METHODS .....             | 343 |
| 3.2.158. | DBA_JOBS .....                     | 344 |
| 3.2.159. | DBA_JOBS_WITH_NAME .....           | 344 |
| 3.2.160. | DBA_LIBRARIES .....                | 344 |
| 3.2.161. | DBA_LOBS .....                     | 344 |
| 3.2.162. | DBA_LOB_PARTITIONS .....           | 344 |

|          |                                     |     |
|----------|-------------------------------------|-----|
| 3.2.163. | DBA_LOB_SUBPARTITIONS .....         | 345 |
| 3.2.164. | DBA_LOG_GROUPS .....                | 345 |
| 3.2.165. | DBA_LOG_GROUP_COLUMNS .....         | 345 |
| 3.2.166. | DBA_METHOD_PARAMS .....             | 345 |
| 3.2.167. | DBA_METHOD_RESULTS .....            | 346 |
| 3.2.168. | DBA_MVIEWS .....                    | 346 |
| 3.2.169. | DBA_MVIEW_COMMENTS .....            | 346 |
| 3.2.170. | DBA_MVIEW_DETAIL_RELATIONS .....    | 346 |
| 3.2.171. | DBA_MVIEW_LOGS .....                | 347 |
| 3.2.172. | DBA_MVIEW_REFRESH_TIMES .....       | 347 |
| 3.2.173. | DBA_OBJECTS .....                   | 347 |
| 3.2.174. | DBA_OBJ_AUDIT_OPTS .....            | 347 |
| 3.2.175. | DBA_PACKAGE_COLL_TYPES .....        | 348 |
| 3.2.176. | DBA_PACKAGE_MEMBER_TYPES .....      | 349 |
| 3.2.177. | DBA_PACKAGE_TYPES .....             | 349 |
| 3.2.178. | DBA_PARALLEL_EXECUTE_CHUNKS .....   | 349 |
| 3.2.179. | DBA_PARALLEL_EXECUTE_TASKS .....    | 350 |
| 3.2.180. | DBA_PART_INDEXES .....              | 351 |
| 3.2.181. | DBA_PART_KEY_COLUMNS .....          | 352 |
| 3.2.182. | DBA_PART_LOBS .....                 | 352 |
| 3.2.183. | DBA_PART_TABLES .....               | 352 |
| 3.2.184. | DBA_PRIV_AUDIT_OPTS .....           | 352 |
| 3.2.185. | DBA_PROCEDURES .....                | 353 |
| 3.2.186. | DBA_PROFILES .....                  | 353 |
| 3.2.187. | DBA_QUEUES .....                    | 353 |
| 3.2.188. | DBA_QUEUE_SUBSCRIBERS .....         | 354 |
| 3.2.189. | DBA_QUEUE_TABLES .....              | 354 |
| 3.2.190. | DBA_RECYCLEBIN .....                | 354 |
| 3.2.191. | DBA_REDEFINITION_OBJECTS .....      | 355 |
| 3.2.192. | DBA_ROLES .....                     | 356 |
| 3.2.193. | DBA_ROLE_PRIVS .....                | 356 |
| 3.2.194. | DBA_SCHEDULER_CHAINS .....          | 357 |
| 3.2.195. | DBA_SCHEDULER_JOBS .....            | 357 |
| 3.2.196. | DBA_SCHEDULER_JOB_LOG .....         | 357 |
| 3.2.197. | DBA_SCHEDULER_JOB_RUN_DETAILS ..... | 357 |
| 3.2.198. | DBA_SCHEDULER_PROGRAMS .....        | 357 |
| 3.2.199. | DBA_SCHEDULER_RULES .....           | 358 |
| 3.2.200. | DBA_SCHEDULER_RULE_CTX .....        | 358 |
| 3.2.201. | DBA_SCHEDULER_RUNNING_JOBS .....    | 358 |
| 3.2.202. | DBA_SCHEDULER_SCHEDULES .....       | 358 |
| 3.2.203. | DBA_SCHEDULER_STEPS .....           | 359 |
| 3.2.204. | DBA_SCHEDULER_STEP_CTX .....        | 359 |
| 3.2.205. | DBA_SEGMENTS .....                  | 359 |

|          |                                     |     |
|----------|-------------------------------------|-----|
| 3.2.206. | DBA_SEQUENCES .....                 | 361 |
| 3.2.207. | DBA_SOURCE .....                    | 361 |
| 3.2.208. | DBA_SQL_PLAN_BASELINES .....        | 361 |
| 3.2.209. | DBA_SQL_PROFILES .....              | 363 |
| 3.2.210. | DBA_STAT_EXTENSIONS .....           | 363 |
| 3.2.211. | DBA_SUBPARTITION_TEMPLATES .....    | 364 |
| 3.2.212. | DBA_SUBPART_KEY_COLUMNS .....       | 364 |
| 3.2.213. | DBA_SYNONYMS .....                  | 364 |
| 3.2.214. | DBA_SYS_PRIVS .....                 | 364 |
| 3.2.215. | DBA_TABLES .....                    | 365 |
| 3.2.216. | DBA_TABLESPACES .....               | 365 |
| 3.2.217. | DBA_TAB_COMMENTS .....              | 368 |
| 3.2.218. | DBA_TAB_HISTGRM_PENDING_STATS ..... | 368 |
| 3.2.219. | DBA_TAB_IDENTITY_COLS .....         | 368 |
| 3.2.220. | DBA_TAB_PENDING_STATS .....         | 368 |
| 3.2.221. | DBA_TBL_COLS .....                  | 369 |
| 3.2.222. | DBA_TBL_COLUMNS .....               | 369 |
| 3.2.223. | DBA_TBL_COL_STATISTICS .....        | 369 |
| 3.2.224. | DBA_TBL_HISTOGRAMS .....            | 369 |
| 3.2.225. | DBA_TBL_PARTITIONS .....            | 370 |
| 3.2.226. | DBA_TBL_PRIVS .....                 | 370 |
| 3.2.227. | DBA_TBL_SUBPARTITIONS .....         | 370 |
| 3.2.228. | DBA_TEMP_FILES .....                | 370 |
| 3.2.229. | DBA_TRIGGERS .....                  | 371 |
| 3.2.230. | DBA_TRIGGER_COLS .....              | 371 |
| 3.2.231. | DBA_TS_QUOTAS .....                 | 372 |
| 3.2.232. | DBA_TYPES .....                     | 372 |
| 3.2.233. | DBA_TYPE_ATTRS .....                | 372 |
| 3.2.234. | DBA_TYPE_METHODS .....              | 373 |
| 3.2.235. | DBA_UNDO_EXTENTS .....              | 373 |
| 3.2.236. | DBA_UNDO_SGMTS .....                | 373 |
| 3.2.237. | DBA_UNUSED_COL_TABS .....           | 374 |
| 3.2.238. | DBA_UPDATABLE_COLUMNS .....         | 374 |
| 3.2.239. | DBA_USERS .....                     | 374 |
| 3.2.240. | DBA_VIEWS .....                     | 376 |
| 3.2.241. | DBMSHSXP_SQL_PROFILE_ATTR .....     | 376 |
| 3.2.242. | DICTIONARY .....                    | 376 |
| 3.2.243. | NLS_DATABASE_PARAMETERS .....       | 377 |
| 3.2.244. | NLS_SESSION_PARAMETERS .....        | 377 |
| 3.2.245. | PUBLICSYN .....                     | 377 |
| 3.2.246. | REDACTION_COLUMNS .....             | 378 |
| 3.2.247. | REDACTION_COLUMNS_DBMS_ERRLOG ..... | 379 |
| 3.2.248. | REDACTION_POLICIES .....            | 379 |

|          |                                      |     |
|----------|--------------------------------------|-----|
| 3.2.249. | REDACTION_VALUES_FOR_TYPE_FULL ..... | 380 |
| 3.2.250. | ROLE_ROLE_PRIVS .....                | 380 |
| 3.2.251. | ROLE_SYS_PRIVS .....                 | 381 |
| 3.2.252. | ROLE_TAB_PRIVS .....                 | 381 |
| 3.2.253. | SESSION_CONTEXT .....                | 382 |
| 3.2.254. | SESSION_PRIVS .....                  | 382 |
| 3.2.255. | SESSION_ROLES .....                  | 383 |
| 3.2.256. | SYS_OBJECTS .....                    | 383 |
| 3.2.257. | USER_ARGUMENTS .....                 | 383 |
| 3.2.258. | USER_AUDIT_TRAIL .....               | 384 |
| 3.2.259. | USER_CATALOG .....                   | 385 |
| 3.2.260. | USER_COLL_TYPES .....                | 385 |
| 3.2.261. | USER_COL_COMMENTS .....              | 385 |
| 3.2.262. | USER_COL_PENDING_STATS .....         | 385 |
| 3.2.263. | USER_COL_PRIVS .....                 | 385 |
| 3.2.264. | USER_COL_PRIVS_MADE .....            | 386 |
| 3.2.265. | USER_COL_PRIVS_RECD .....            | 386 |
| 3.2.266. | USER_CONSTRAINTS .....               | 386 |
| 3.2.267. | USER_CONS_COLUMNS .....              | 386 |
| 3.2.268. | USER_DB_LINKS .....                  | 387 |
| 3.2.269. | USER_DEPENDENCIES .....              | 387 |
| 3.2.270. | USER_ENCRYPTED_COLUMNS .....         | 387 |
| 3.2.271. | USER_ERRORS .....                    | 387 |
| 3.2.272. | USER_EXTENTS .....                   | 388 |
| 3.2.273. | USER_EXTERNAL_LOCATIONS .....        | 388 |
| 3.2.274. | USER_EXTERNAL_TABLES .....           | 388 |
| 3.2.275. | USER_FREE_SPACE .....                | 389 |
| 3.2.276. | USER_IDX_COLUMNS .....               | 389 |
| 3.2.277. | USER_IDX_EXPRESSIONS .....           | 389 |
| 3.2.278. | USER_IDX_PARTITIONS .....            | 390 |
| 3.2.279. | USER_INDEXES .....                   | 390 |
| 3.2.280. | USER_IND_PENDING_STATS .....         | 390 |
| 3.2.281. | USER_JAVA_ARGUMENTS .....            | 390 |
| 3.2.282. | USER_JAVA_CLASSES .....              | 390 |
| 3.2.283. | USER_JAVA_FIELDS .....               | 391 |
| 3.2.284. | USER_JAVA_METHODS .....              | 391 |
| 3.2.285. | USER_JOBS .....                      | 391 |
| 3.2.286. | USER_JOBS_WITH_NAME .....            | 391 |
| 3.2.287. | USER_LIBRARIES .....                 | 392 |
| 3.2.288. | USER_LOBS .....                      | 392 |
| 3.2.289. | USER_LOB_PARTITIONS .....            | 392 |
| 3.2.290. | USER_LOB_SUBPARTITIONS .....         | 392 |
| 3.2.291. | USER_LOG_GROUPS .....                | 392 |

|          |                                      |     |
|----------|--------------------------------------|-----|
| 3.2.292. | USER_LOG_GROUP_COLUMNS .....         | 393 |
| 3.2.293. | USER_METHOD_PARAMS .....             | 393 |
| 3.2.294. | USER_METHOD_RESULTS .....            | 393 |
| 3.2.295. | USER_MVIEWS .....                    | 393 |
| 3.2.296. | USER_MVIEW_COMMENTS .....            | 394 |
| 3.2.297. | USER_MVIEW_DETAIL_RELATIONS .....    | 394 |
| 3.2.298. | USER_MVIEW_LOGS .....                | 394 |
| 3.2.299. | USER_MVIEW_REFRESH_TIMES .....       | 394 |
| 3.2.300. | USER_OBJECTS .....                   | 395 |
| 3.2.301. | USER_OBJ_AUDIT_OPTS .....            | 395 |
| 3.2.302. | USER_PACKAGE_COLL_TYPES .....        | 396 |
| 3.2.303. | USER_PACKAGE_MEMBER_TYPES .....      | 396 |
| 3.2.304. | USER_PACKAGE_TYPES .....             | 396 |
| 3.2.305. | USER_PARALLEL_EXECUTE_CHUNKS .....   | 397 |
| 3.2.306. | USER_PARALLEL_EXECUTE_TASKS .....    | 398 |
| 3.2.307. | USER_PART_INDEXES .....              | 399 |
| 3.2.308. | USER_PART_KEY_COLUMNS .....          | 399 |
| 3.2.309. | USER_PART_LOBS .....                 | 399 |
| 3.2.310. | USER_PART_TABLES .....               | 400 |
| 3.2.311. | USER_PROCEEDURES .....               | 400 |
| 3.2.312. | USER_QUEUES .....                    | 400 |
| 3.2.313. | USER_QUEUE_SUBSCRIBERS .....         | 400 |
| 3.2.314. | USER_QUEUE_TABLES .....              | 400 |
| 3.2.315. | USER_RECYCLEBIN .....                | 401 |
| 3.2.316. | USER_ROLE_PRIVS .....                | 401 |
| 3.2.317. | USER_SCHEDULER_CHAINS .....          | 402 |
| 3.2.318. | USER_SCHEDULER_JOBS .....            | 402 |
| 3.2.319. | USER_SCHEDULER_JOB_LOG .....         | 402 |
| 3.2.320. | USER_SCHEDULER_JOB_RUN_DETAILS ..... | 402 |
| 3.2.321. | USER_SCHEDULER_PROGRAMS .....        | 403 |
| 3.2.322. | USER_SCHEDULER_RULES .....           | 403 |
| 3.2.323. | USER_SCHEDULER_RULE_CTX .....        | 403 |
| 3.2.324. | USER_SCHEDULER_RUNNING_JOBS .....    | 403 |
| 3.2.325. | USER_SCHEDULER_SCHEDULES .....       | 403 |
| 3.2.326. | USER_SCHEDULER_STEPS .....           | 404 |
| 3.2.327. | USER_SCHEDULER_STEP_CTX .....        | 404 |
| 3.2.328. | USER_SEGMENTS .....                  | 404 |
| 3.2.329. | USER_SEQUENCES .....                 | 404 |
| 3.2.330. | USER_SOURCE .....                    | 405 |
| 3.2.331. | USER_STAT_EXTENSIONS .....           | 405 |
| 3.2.332. | USER_SUBPARTITION_TEMPLATES .....    | 405 |
| 3.2.333. | USER_SUBPART_KEY_COLUMNS .....       | 405 |
| 3.2.334. | USER_SYNONYMS .....                  | 405 |

|            |                                      |            |
|------------|--------------------------------------|------------|
| 3.2.335.   | USER_SYS_PRIVS .....                 | 406        |
| 3.2.336.   | USER_TABLES .....                    | 406        |
| 3.2.337.   | USER_TABLESPACES .....               | 406        |
| 3.2.338.   | USER_TAB_COMMENTS .....              | 409        |
| 3.2.339.   | USER_TAB_HISTGRM_PENDING_STATS ..... | 409        |
| 3.2.340.   | USER_TAB_IDENTITY_COLS .....         | 410        |
| 3.2.341.   | USER_TAB_PENDING_STATS .....         | 410        |
| 3.2.342.   | USER_TBL_COLS .....                  | 410        |
| 3.2.343.   | USER_TBL_COLUMNS .....               | 410        |
| 3.2.344.   | USER_TBL_COL_STATISTICS .....        | 411        |
| 3.2.345.   | USER_TBL_HISTOGRAMS .....            | 411        |
| 3.2.346.   | USER_TBL_PARTITIONS .....            | 411        |
| 3.2.347.   | USER_TBL_PRIVS .....                 | 411        |
| 3.2.348.   | USER_TBL_PRIVS_MADE .....            | 411        |
| 3.2.349.   | USER_TBL_PRIVS_RECD .....            | 412        |
| 3.2.350.   | USER_TRIGGERS .....                  | 412        |
| 3.2.351.   | USER_TRIGGER_COLS .....              | 412        |
| 3.2.352.   | USER_TYPES .....                     | 413        |
| 3.2.353.   | USER_TYPE_ATTRS .....                | 413        |
| 3.2.354.   | USER_TYPE_METHODS .....              | 413        |
| 3.2.355.   | USER_UNUSED_COL_TABS .....           | 413        |
| 3.2.356.   | USER_UPDATABLE_COLUMNS .....         | 414        |
| 3.2.357.   | USER_USERS .....                     | 414        |
| 3.2.358.   | USER_VIEWS .....                     | 415        |
| <b>제4장</b> | <b>Dynamic View .....</b>            | <b>417</b> |
| 4.1.       | Overview .....                       | 417        |
| 4.2.       | List of Dynamic Views .....          | 417        |
| 4.2.1.     | V\$ACCESS .....                      | 424        |
| 4.2.2.     | V\$ACTIVE_SESSION_HISTORY .....      | 425        |
| 4.2.3.     | V\$ALL_FILESTAT .....                | 427        |
| 4.2.4.     | V\$ALL_LOG_DEST .....                | 428        |
| 4.2.5.     | V\$ARCHIVED_LOG .....                | 428        |
| 4.2.6.     | V\$ARCHIVE_DEST_FILES .....          | 429        |
| 4.2.7.     | V\$AS_ALIAS .....                    | 429        |
| 4.2.8.     | V\$AS_DISK .....                     | 430        |
| 4.2.9.     | V\$AS_DISKSPACE .....                | 431        |
| 4.2.10.    | V\$AS_DISK_HISTORY .....             | 432        |
| 4.2.11.    | V\$AS_EXTENT_MAP .....               | 432        |
| 4.2.12.    | V\$AS_FILE .....                     | 433        |
| 4.2.13.    | V\$AS_OPERATION .....                | 434        |
| 4.2.14.    | V\$AS_USER .....                     | 434        |
| 4.2.15.    | V\$AS_USERGROUP .....                | 435        |

|         |                                  |     |
|---------|----------------------------------|-----|
| 4.2.16. | V\$AS_USERGROUP_MEMBER .....     | 435 |
| 4.2.17. | V\$AUTO_COALESCE .....           | 436 |
| 4.2.18. | V\$BACKUP .....                  | 436 |
| 4.2.19. | V\$BACKUP_ARCHIVED_LOG .....     | 437 |
| 4.2.20. | V\$BACKUP_LIST .....             | 437 |
| 4.2.21. | V\$BACKUP_LIST_ARCHIVELOG .....  | 438 |
| 4.2.22. | V\$BACKUP_LIST_CONTROLFILE ..... | 438 |
| 4.2.23. | V\$BACKUP_LIST_DATAFILE .....    | 438 |
| 4.2.24. | V\$BACKUP_SET .....              | 439 |
| 4.2.25. | V\$BACKUP_SET_TABLESPACE .....   | 440 |
| 4.2.26. | V\$BG_SESSION .....              | 440 |
| 4.2.27. | V\$BG_SESSTAT .....              | 442 |
| 4.2.28. | V\$BH .....                      | 443 |
| 4.2.29. | V\$BLOCKER_SESSION .....         | 443 |
| 4.2.30. | V\$BLOCK_CHANGE_TRACKING .....   | 444 |
| 4.2.31. | V\$BPARAMETERS .....             | 444 |
| 4.2.32. | V\$BUFFER_CACHE_ADVICE .....     | 445 |
| 4.2.33. | V\$BUFFER_POOL .....             | 445 |
| 4.2.34. | V\$BUFFER_POOL_STATISTICS .....  | 445 |
| 4.2.35. | V\$CHECKPOINT .....              | 446 |
| 4.2.36. | V\$CLUSTER_INTERCONNECTS .....   | 447 |
| 4.2.37. | V\$CONTEXT .....                 | 448 |
| 4.2.38. | V\$CONTROLFILE .....             | 448 |
| 4.2.39. | V\$CORRUPT_XID .....             | 448 |
| 4.2.40. | V\$DATABASE .....                | 449 |
| 4.2.41. | V\$DATAFILE .....                | 451 |
| 4.2.42. | V\$DATAFILE_HEADER .....         | 451 |
| 4.2.43. | V\$DBFILE .....                  | 452 |
| 4.2.44. | V\$DBLINK .....                  | 453 |
| 4.2.45. | V\$DBLINK_STATUS .....           | 453 |
| 4.2.46. | V\$DB_PIPES .....                | 454 |
| 4.2.47. | V\$DML_STATS .....               | 454 |
| 4.2.48. | V\$ENCRYPTED_TABLESPACES .....   | 455 |
| 4.2.49. | V\$ENCRYPTION_WALLET .....       | 455 |
| 4.2.50. | V\$ENQUEUE_STAT .....            | 455 |
| 4.2.51. | V\$EST_CHECKPOINT_TIME .....     | 456 |
| 4.2.52. | V\$EVENT_NAME .....              | 456 |
| 4.2.53. | V\$FILESTAT .....                | 457 |
| 4.2.54. | V\$FLASHBACK_ARCHIVED_LOG .....  | 457 |
| 4.2.55. | V\$FLASHBACK_LOG .....           | 458 |
| 4.2.56. | V\$FLASHBACK_MARKER .....        | 459 |
| 4.2.57. | V\$GLOBAL_TRANSACTION .....      | 459 |
| 4.2.58. | V\$IM_COLUMN_LEVEL .....         | 460 |

|          |                                  |     |
|----------|----------------------------------|-----|
| 4.2.59.  | V\$IM_SEGMENTS .....             | 460 |
| 4.2.60.  | V\$INSTANCE .....                | 462 |
| 4.2.61.  | V\$INSTANCEMETRIC .....          | 463 |
| 4.2.62.  | V\$INSTANCEMETRIC_HISTORY .....  | 463 |
| 4.2.63.  | V\$INTERCONNECTION .....         | 464 |
| 4.2.64.  | V\$INTERCONNECT_LATENCY .....    | 465 |
| 4.2.65.  | V\$IPARAM_DESC .....             | 465 |
| 4.2.66.  | V\$LATCH .....                   | 465 |
| 4.2.67.  | V\$LIBRARYCACHE .....            | 466 |
| 4.2.68.  | V\$LICENSE .....                 | 467 |
| 4.2.69.  | V\$LISTENER_PORT .....           | 468 |
| 4.2.70.  | V\$LOCK .....                    | 468 |
| 4.2.71.  | V\$LOCKED_OBJECT .....           | 469 |
| 4.2.72.  | V\$LOG .....                     | 470 |
| 4.2.73.  | V\$LOGFILE .....                 | 470 |
| 4.2.74.  | V\$LOG_HISTORY .....             | 471 |
| 4.2.75.  | V\$MEDIA_RECOVERY_PROGRESS ..... | 472 |
| 4.2.76.  | V\$MEMORY_TABLESPACE .....       | 472 |
| 4.2.77.  | V\$MEM_MGR .....                 | 473 |
| 4.2.78.  | V\$METRIC .....                  | 473 |
| 4.2.79.  | V\$METRICNAME .....              | 474 |
| 4.2.80.  | V\$MODIFIED_PARAMETERS .....     | 475 |
| 4.2.81.  | V\$MYSTAT .....                  | 475 |
| 4.2.82.  | V\$NLS_PARAMETERS .....          | 475 |
| 4.2.83.  | V\$OBJECT_USAGE .....            | 476 |
| 4.2.84.  | V\$OPEN_CURSOR .....             | 476 |
| 4.2.85.  | V\$OSSTAT .....                  | 477 |
| 4.2.86.  | V\$OSSTAT2 .....                 | 477 |
| 4.2.87.  | V\$OUTLINE .....                 | 478 |
| 4.2.88.  | V\$PARAMETERS .....              | 478 |
| 4.2.89.  | V\$PATCH .....                   | 478 |
| 4.2.90.  | V\$PE_PESPLAN .....              | 479 |
| 4.2.91.  | V\$PE_PESSTAT .....              | 479 |
| 4.2.92.  | V\$PE_PROCESS .....              | 479 |
| 4.2.93.  | V\$PE_SESSION .....              | 480 |
| 4.2.94.  | V\$PE_SYSSTAT .....              | 481 |
| 4.2.95.  | V\$PE_TQSTAT .....               | 481 |
| 4.2.96.  | V\$PGASTAT .....                 | 482 |
| 4.2.97.  | V\$PROCESS .....                 | 483 |
| 4.2.98.  | V\$PROC_CONN_INFO .....          | 484 |
| 4.2.99.  | V\$PROXY_SQLTEXT .....           | 484 |
| 4.2.100. | V\$PROXY_SQL_PARAM .....         | 484 |
| 4.2.101. | V\$RECOVERY_FILE_STATUS .....    | 485 |

|          |                                         |     |
|----------|-----------------------------------------|-----|
| 4.2.102. | V\$RECOVERY_PROGRESS .....              | 485 |
| 4.2.103. | V\$RECOVERY_STATUS .....                | 486 |
| 4.2.104. | V\$RECOVER_FILE .....                   | 487 |
| 4.2.105. | V\$RESERVED_WORDS .....                 | 487 |
| 4.2.106. | V\$RESULT_CACHE_DEPENDENCY .....        | 487 |
| 4.2.107. | V\$RESULT_CACHE_OBJECTS .....           | 488 |
| 4.2.108. | V\$RESULT_CACHE_OBJECT_STATISTICS ..... | 489 |
| 4.2.109. | V\$RESULT_CACHE_STATISTICS .....        | 489 |
| 4.2.110. | V\$ROLLSTAT .....                       | 490 |
| 4.2.111. | V\$ROWCACHE .....                       | 491 |
| 4.2.112. | V\$RSBSTAT .....                        | 491 |
| 4.2.113. | V\$RSRC_CONSUMER_GROUP .....            | 492 |
| 4.2.114. | V\$RSRC_PLAN .....                      | 493 |
| 4.2.115. | V\$RSRC_SESSION_INFO .....              | 493 |
| 4.2.116. | V\$RUNNING_JOBS .....                   | 494 |
| 4.2.117. | V\$RUNNING_JOBS_WITH_NAME .....         | 494 |
| 4.2.118. | V\$SEGMENT_STATISTICS .....             | 495 |
| 4.2.119. | V\$SERVER_EVENT_STATUS .....            | 495 |
| 4.2.120. | V\$SERVER_EVENT_SUBTYPE .....           | 495 |
| 4.2.121. | V\$SERVER_EVENT_TYPE .....              | 496 |
| 4.2.122. | V\$SESSION .....                        | 496 |
| 4.2.123. | V\$SESSION_EVENT .....                  | 501 |
| 4.2.124. | V\$SESSION_IO .....                     | 501 |
| 4.2.125. | V\$SESSION_LATEST_STAT .....            | 502 |
| 4.2.126. | V\$SESSION_LONGOPS .....                | 502 |
| 4.2.127. | V\$SESSION_WAIT .....                   | 503 |
| 4.2.128. | V\$SESSTAT .....                        | 504 |
| 4.2.129. | V\$SESS_TIME_MODEL .....                | 504 |
| 4.2.130. | V\$SGA .....                            | 505 |
| 4.2.131. | V\$SGASTAT .....                        | 506 |
| 4.2.132. | V\$SLOG .....                           | 506 |
| 4.2.133. | V\$SPINLOCK HOLDER .....                | 507 |
| 4.2.134. | V\$SPINUSE_STAT .....                   | 507 |
| 4.2.135. | V\$SQL .....                            | 507 |
| 4.2.136. | V\$SQLAREA .....                        | 509 |
| 4.2.137. | V\$SQLSTATS .....                       | 511 |
| 4.2.138. | V\$SQLTEXT .....                        | 513 |
| 4.2.139. | V\$SQLTEXT2 .....                       | 514 |
| 4.2.140. | V\$SQLTEXT_WITH_NEWLINES .....          | 514 |
| 4.2.141. | V\$SQLTEXT_WITH_NEWLINES2 .....         | 515 |
| 4.2.142. | V\$SQL_BIND_CAPTURE .....               | 516 |
| 4.2.143. | V\$SQL_BIND_CAPTURE_ALL .....           | 516 |
| 4.2.144. | V\$SQL_EXPC_INFO .....                  | 517 |

|          |                                  |     |
|----------|----------------------------------|-----|
| 4.2.145. | V\$SQL_MONITOR .....             | 518 |
| 4.2.146. | V\$SQL_PLAN .....                | 521 |
| 4.2.147. | V\$SQL_PLAN_MONITOR .....        | 522 |
| 4.2.148. | V\$SQL_PLAN_STATISTICS .....     | 524 |
| 4.2.149. | V\$SQL_WORKAREA .....            | 526 |
| 4.2.150. | V\$SQL_WORKAREA_HISTOGRAM .....  | 528 |
| 4.2.151. | V\$SSVR_CLIENT .....             | 528 |
| 4.2.152. | V\$SSVR_FLASHCACHE .....         | 529 |
| 4.2.153. | V\$SSVR_GRID_DISK .....          | 529 |
| 4.2.154. | V\$SSVR_IOLOAD .....             | 530 |
| 4.2.155. | V\$SSVR_IORM_CATEGORY_PLAN ..... | 530 |
| 4.2.156. | V\$SSVR_IORM_DB_PLAN .....       | 531 |
| 4.2.157. | V\$SSVR_IOSTAT .....             | 531 |
| 4.2.158. | V\$SSVR_MEMSTAT .....            | 532 |
| 4.2.159. | V\$SSVR_SLAB_STAT .....          | 532 |
| 4.2.160. | V\$SSVR_STORAGE_DISK .....       | 532 |
| 4.2.161. | V\$STANDBY .....                 | 533 |
| 4.2.162. | V\$STANDBY_DEST .....            | 534 |
| 4.2.163. | V\$STANDBY_LOG .....             | 535 |
| 4.2.164. | V\$STANDBY_LOGFILE .....         | 535 |
| 4.2.165. | V\$STANDBY_STATUS .....          | 536 |
| 4.2.166. | V\$STATNAME .....                | 536 |
| 4.2.167. | V\$SYSSTAT .....                 | 537 |
| 4.2.168. | V\$SYSSTAT_VALUE .....           | 537 |
| 4.2.169. | V\$SYSTEM_EVENT .....            | 538 |
| 4.2.170. | V\$SYSTEM_PARAMETERS .....       | 538 |
| 4.2.171. | V\$SYS_TIME_MODEL .....          | 538 |
| 4.2.172. | V\$TABLESPACE .....              | 539 |
| 4.2.173. | V\$TACSTAT .....                 | 539 |
| 4.2.174. | V\$TEMPFILE .....                | 540 |
| 4.2.175. | V\$TEMPSEG_OP_USAGE .....        | 540 |
| 4.2.176. | V\$TEMPSEG_USAGE .....           | 541 |
| 4.2.177. | V\$TEMPSTAT .....                | 542 |
| 4.2.178. | V\$THRINFO .....                 | 543 |
| 4.2.179. | V\$THR_ACT_INFO .....            | 544 |
| 4.2.180. | V\$THR_EVENT .....               | 544 |
| 4.2.181. | V\$TIMER .....                   | 545 |
| 4.2.182. | V\$TIMEZONE_NAMES .....          | 545 |
| 4.2.183. | V\$TRANSACTION .....             | 546 |
| 4.2.184. | V\$TSN_TIME .....                | 548 |
| 4.2.185. | V\$UNDOSTAT .....                | 548 |
| 4.2.186. | V\$UNDO_FREE_SPACE .....         | 549 |
| 4.2.187. | V\$USGMT_TXENTRY .....           | 549 |

|                    |                                          |            |
|--------------------|------------------------------------------|------------|
| 4.2.188.           | V\$VERSION .....                         | 550        |
| 4.2.189.           | V\$WAITER_SESSION .....                  | 550        |
| 4.2.190.           | V\$WAITSTAT .....                        | 550        |
| 4.2.191.           | V\$WAITUSE_STAT .....                    | 551        |
| 4.2.192.           | V\$WAITUSE_THR_STAT .....                | 551        |
| <b>제5장</b>         | <b>Database Limits .....</b>             | <b>553</b> |
| 5.1.               | Overview .....                           | 553        |
| 5.2.               | List of Datatype Limits .....            | 553        |
| 5.3.               | List of Physical Database Limits .....   | 553        |
| 5.4.               | List of Logical Database Limits .....    | 555        |
| 5.5.               | List of Process and Runtime Limits ..... | 555        |
| <b>Appendix A.</b> | <b>초기화 파라미터의 설정 예제 .....</b>             | <b>557</b> |
| <b>Appendix B.</b> | <b>COMMAND .....</b>                     | <b>559</b> |
| <b>색인</b>          | <b>.....</b>                             | <b>565</b> |



# 안내서에 대하여

## 안내서의 대상

본 안내서는 Tibero<sup>®</sup>(이하 Tibero)를 사용하는 데 있어 필요한 초기화 파라미터, 데이터 사전, 정적 뷰, 동적 뷰의 개념과 사용법을 참조하려는 데이터베이스 관리자(Database Administrator, 이하 DBA), 시스템 관리자, 애플리케이션 프로그램 개발자를 대상으로 기술한다.

## 안내서의 전제 조건

본 안내서를 원활히 이해하기 위해서는 다음과 같은 사항을 미리 알고 있어야 한다.

- 데이터베이스의 이해
- DB의 이해
- SQL의 이해

## 안내서의 제한 조건

본 안내서는 Tibero를 실무에 적용하거나 운용하는 데 필요한 모든 사항을 포함하고 있지 않다. 따라서 설치, 환경설정 등 운용 및 관리에 대해서는 각 제품 안내서를 참고하기 바란다.

---

### 참고

Tibero의 설치 및 환경설정에 관한 내용은 "Tibero 설치 안내서"를 참고한다.

---

## 안내서 구성

Tibero 참조 안내서는 총 4개의 장과 Appendix로 이루어져 있다.

각 장의 주요 내용은 다음과 같다.

- 제1장: Introduction to Initialization Parameters

초기화 파라미터의 개념을 간략히 설명하고 Tibero가 제공하는 초기화 파라미터를 기술한다.

- 제2장: 데이터 사전

데이터 사전에 대한 개념과 구조를 설명하고 데이터 사전의 참조 및 갱신을 기술한다.

- 제3장: Static View

Tibero의 데이터 사전에서 제공하는 정적 뷰를 간략히 설명하고 이를 구성하는 컬럼을 기술한다.

- 제4장: Dynamic View

Tibero의 데이터 사전에서 제공하는 동적 뷰를 간략히 설명하고 이를 구성하는 컬럼을 기술한다.

- 제5장: Database Limit

Tibero의 데이터베이스 기능 및 개체와 관련된 값의 제한을 기술한다.

- Appendix A.: 초기화 파라미터의 설정 예제

Tibero를 설치한 후 기본적으로 설정되는 초기화 파라미터를 기술한다.

## 안내서 규약

| 표기                 | 의미                              |
|--------------------|---------------------------------|
| <<AaBbCc123>>      | 프로그램 소스 코드의 파일명                 |
| <Ctrl>+C           | Ctrl과 C를 동시에 누름                 |
| [Button]           | GUI의 버튼 또는 메뉴 이름                |
| 진하게                | 강조                              |
| " "(따옴표)           | 다른 관련 안내서 또는 안내서 내의 다른 장 및 절 언급 |
| '입력항목'             | 화면 UI에서 입력 항목에 대한 설명            |
| 하이퍼링크              | 메일 계정, 웹 사이트                    |
| >                  | 메뉴의 진행 순서                       |
| +----              | 하위 디렉터리 또는 파일 있음                |
| ----               | 하위 디렉터리 또는 파일 없음                |
| <u>참고</u>          | 참고 또는 주의사항                      |
| <u>주의</u>          | 주의할 사항                          |
| [그림 1.1]           | 그림 이름                           |
| [예 1.1]            | 예제 이름                           |
| AaBbCc123          | Java 코드, XML 문서                 |
| [command argument] | 옵션 파라미터                         |
| < xyz >            | '<'와 '>' 사이의 내용이 실제 값으로 변경됨     |
|                    | 선택 사항. 예) A B: A나 B 중 하나        |
| ...                | 파라미터 등이 반복되어서 나옴                |
| \${ }              | 환경변수                            |

# 시스템 사용 환경

|               |                                             |                                        |
|---------------|---------------------------------------------|----------------------------------------|
|               | 요구 사항                                       |                                        |
| Platform      | Solaris (Solaris 11)                        |                                        |
|               | AIX (AIX 7.1/AIX 7.2/AIX 7.3)               |                                        |
|               | GNU (X86, 64, IA64)                         |                                        |
|               | Red Hat Enterprise Linux 7 kernel 3.10.0 이상 |                                        |
|               | Windows(x86) 64bit                          |                                        |
|               | Hardware                                    | 최 소<br>2.5GB<br>하 드<br>디 스<br>크 공<br>간 |
| 1GB 이상 메모리 공간 |                                             |                                        |
| Compiler      | PSM (C99 지원 필요)                             |                                        |
|               | tbESQL/C (C99 지원 필요)                        |                                        |

## 관련 안내서

| 안내서                              | 설명                                                                                                                       |
|----------------------------------|--------------------------------------------------------------------------------------------------------------------------|
| Tibero<br>설치 안내서                 | 설치 과정에 필요한 시스템 요구사항과 설치 및 제거 방법을 기술한 안내서이다.                                                                              |
| Tibero<br>tbCLI 안내서              | Call Level Interface인 tbCLI의 개념과 구성요소, 프로그램 구조를 소개하고 tbCLI 프로그램을 작성하는 데 필요한 데이터 타입, 함수, 에러 메시지를 기술한 안내서이다.               |
| Tibero<br>애플리케이션 개발자 안내서         | 각종 애플리케이션 라이브러리를 이용하여 애플리케이션 프로그램을 개발하는 방법을 기술한 안내서이다.                                                                   |
| Tibero<br>External Procedure 안내서 | External Procedure를 소개하고 이를 생성하고 사용하는 방법을 기술한 안내서이다.                                                                     |
| Tibero<br>JDBC 개발자 안내서           | Tibero에서 제공하는 JDBC 기능을 이용하여 애플리케이션 프로그램을 개발하는 방법을 기술한 안내서이다.                                                             |
| Tibero<br>tbESQL/C 안내서           | C 프로그래밍 언어를 사용해 데이터베이스 작업을 수행하는 각종 애플리케이션 프로그램을 작성하는 방법을 기술한 안내서이다.                                                      |
| Tibero<br>tbESQL/COBOL 안내서       | COBOL 프로그래밍 언어를 사용해 데이터베이스 작업을 수행하는 각종 애플리케이션 프로그램을 작성하는 방법을 기술한 안내서이다.                                                  |
| Tibero<br>tbPSM 안내서              | 저장 프러시저 모듈인 tbPSM의 개념과 문법, 구성요소를 소개하고, 프로그램을 작성하는 데 필요한 제어 구조, 복합 타입, 서브프로그램, 패키지 및 SQL 문장을 실행하고 에러를 처리하는 방법을 기술한 안내서이다. |
| Tibero<br>tbPSM 참조 안내서           | 저장 프러시저 모듈인 tbPSM의 패키지를 소개하고, 이러한 패키지에 포함된 각 프러시저와 함수의 프로토타입, 파라미터, 예제 등을 기술한 참조 안내서이다.                                  |
| Tibero<br>관리자 안내서                | Tibero의 동작과 주요 기능의 원활한 수행을 보장하기 위해 DBA가 알아야 할 관리 방법을 논리적 또는 물리적 측면에서 설명하고, 관리를 지원하는 각종 도구를 기술한 안내서이다.                    |
| Tibero<br>유틸리티 안내서               | 데이터베이스와 관련된 작업을 수행하기 위해 필요한 유틸리티의 설치 및 환경설정, 사용 방법을 기술한 안내서이다.                                                           |
| Tibero<br>TAS 안내서                | Tibero Active Storage(TAS)를 사용해서 Tibero의 파일을 관리하고자 하는 관리자를 대상으로 기술한 안내서이다.                                               |

| 안내서                   | 설명                                                            |
|-----------------------|---------------------------------------------------------------|
| Tibero<br>에러 참조 안내서   | Tibero를 사용하는 도중에 발생할 수 있는 각종 에러의 원인과 해결 방법을 기술한 안내서이다.        |
| Tibero<br>SQL 참조 안내서  | 데이터베이스 작업을 수행하거나 애플리케이션 프로그램을 작성할 때 필요한 SQL 문장을 기술한 참조 안내서이다. |
| Tibero<br>TDP.NET 안내서 | Tibero Data Provider for .NET 기능을 기술하는 안내서이다.                 |
| Tibero<br>IMCS 안내서    | Tibero에서 제공하는 In-Memory Column Store(이하 IMCS) 기능을 기술하는 안내서이다. |

# 제1장 Introduction to Initialization Parameters

## 1.1. Overview

An **initialization parameter** describes a Tiberio instance and database configuration. It is provided with a text file and required to create a Tiberio.

An **initialization parameter file** is a text file, with a filename in the form XX.tip (where XX is the name of the database), needed when creating a database. It can be edited using a document editor provided with the operating system. Once the initialization parameter file is modified, the database must be restarted to apply the modification to the entire system.

However, not all initialization parameters require the restarting of the database after their modification. Some parameters can be modified while being used by an instance. These parameters are applied only to the current instance, not the initialization parameter file.

The following is the list of information that can be configured in the initialization parameter file.

- Shared memory area
- Default value for the database and instances
- Database limits
- Location of the control file
- Location of the archive file
- Location of the trace file

Below is an example of the initialization parameter file.

```
AUTHENTICATION_TIMEOUT = 0  
CLUSTER_DATABASE = N  
EX_WORKAREA_CACHE_SIZE = 100
```

## 1.2. Class Attributes

This section explains the class attributes necessary for users to understand the information configured in the initialization parameter file.

| Group                 | Class      | Description                                                                                                                                      |
|-----------------------|------------|--------------------------------------------------------------------------------------------------------------------------------------------------|
| Static/Dynamic        | Static     | A parameter that cannot be modified during system operation.                                                                                     |
|                       | Dynamic    | A parameter that can be modified during system operation.                                                                                        |
| Persistent/Adjustable | Persistent | A parameter that cannot be modified after database creation.                                                                                     |
|                       | Adjustable | A parameter that can be modified after database creation.                                                                                        |
| Mandatory/Optional    | Mandatory  | A parameter required to start up a system. If these parameter are not defined in the initialization parameter file, the system cannot be booted. |
|                       | Optional   | A parameter that uses a default value if not defined in the initialization parameter.                                                            |
| System/Session        | System     | A parameter that the entire system shares.                                                                                                       |
|                       | Session    | A parameter that can be configured separately for each session.                                                                                  |

## 1.3. Initialization Parameter

This section describes initialization parameters in alphabetical order.

### 1.3.1. ACF\_CLGC\_CNT

Specifies the number of CCC garbage collection (GC) threads created within the ACSD process in TAC.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | MAX(1, MIN(get_ncpu() / 8, 32))      |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.2. ACF\_CMPT\_CNT

ACF\_CMPT\_CNT specifies the number of cluster message processing threads used by the ACSD (Active Cluster Service Deamon) process in parallel.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | MAX(32, MIN(get_ncpu() * 3, 160))    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.3. ACF\_NMGR\_CM\_AUTO\_REVIVAL

ACF\_NMGR\_CM\_AUTO\_REVIVAL specifies whether or not to apply the logic for CM and CM\_GUARD to revive one another when the other is killed. (instance side)

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.4. ACF\_NMGR\_PREVENT\_REVIVAL\_ON\_IR

ACF\_NMGR\_PREVENT\_REVIVAL\_ON\_IR specifies whether or not to execute the Revival function when a DBWR loses its connection with CM even if the status of DBWR is 'IR HOLD' or 'IR WAITING'.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | Y                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.5. ACTIVE\_SESSION\_HISTORY

ACTIVE\_SESSION\_HISTORY determines whether to gather statistics on active sessions.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.6. ACTIVE\_SESSION\_TIMEOUT

The maximum time, in seconds, that a session will remain active and wait for a client request. If there is no client request during this time, the server will disconnect the session. If this is set to 0, the server will never disconnect the session.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Integer                                |
| Default Value  | 0                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          |                                        |

### 1.3.7. ACTIVE\_TX\_SESSION\_TIMEOUT

Maximum time period during which an active session that contains an active transaction will wait client requests (in seconds). If no incoming request is received during this time period, the server will terminate the session and any transaction within that session will be rolled back. If set to 0, this feature is disabled. If ACTIVE\_TX\_SESSION\_TIMEOUT and ACTIVE\_SESSION\_TIMEOUT are both configured, ACTIVE\_TX\_SESSION\_TIMEOUT overrides ACTIVE\_SESSION\_TIMEOUT when there is an active transaction in the session. Otherwise, ACTIVE\_SESSION\_TIMEOUT will take precedence.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Integer                                |
| Default Value  | 0                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          |                                        |

### 1.3.8. ACT\_DUMP\_BACKUP\_LIMIT\_CNT

Specifies the maximum number of backups that can be stored in ACT\_DUMP\_DEST. For details, refer to the description of DFLT\_DUMP\_BACKUP\_LIMIT\_CNT. Consequently, the maximum path size may approximate  $\text{ACT\_DUMP\_TOTAL\_SIZE\_LIMIT} * (\text{ACT\_DUMP\_BACKUP\_LIMIT\_CNT} + 1)$ .

- Parameter Format

| Property       | Description                                |
|----------------|--------------------------------------------|
| Parameter Type | Integer                                    |
| Default Value  | <a href="#">DFLT_DUMP_BACKUP_LIMIT_CNT</a> |
| Class          | Optional, Adjustable, Dynamic, System      |
| Range          | 2 ~ 500                                    |

### 1.3.9. ACT\_DUMP\_DEST

Specifies the directory where the activity dump files are stored. Must be specified as an absolute path.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | Depending on system environment       |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.10. ACT\_DUMP\_TOTAL\_SIZE\_LIMIT

Specifies the maximum total file size that can be stored in the directory specified in ACT\_DUMP\_DEST, excluding backup directories. For details, refer to the description of DFLT\_DUMP\_TOTAL\_SIZE\_LIMIT.

- Parameter Format

| Property       | Description                                |
|----------------|--------------------------------------------|
| Parameter Type | Integer                                    |
| Default Value  | <a href="#">DFLT_DUMP_TOTAL_SIZE_LIMIT</a> |
| Class          | Optional, Adjustable, Dynamic, System      |
| Range          | 100K - INT64_MAX                           |

### 1.3.11. ACT\_LOG\_DEST

Specifies the directory where the activity log files are stored. Must be specified as an absolute path.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | Depending on system environment       |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.12. ACT\_LOG\_FILE\_SIZE

Specifies the maximum size of each activity log file. When a single activity log file exceeds the size specified in this parameter, the current activity log file is closed and a new file is created.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 104857600                             |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 100K - 1G                             |

### 1.3.13. ACT\_LOG\_TOTAL\_SIZE\_LIMIT

Specifies the maximum total file size that can be stored in the directory specified in ACT\_LOG\_DEST. This parameter prevents activity log files from unlimited growth. When the size of files exceeds the specified value, the oldest files are removed to allow disk space reuse.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 300 * 1024 * 1024                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 100K - INT64_MAX                      |

### 1.3.14. AD\_AUTH\_KBR\_SERVICE\_NAME

kerberos service name for AD authentication.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ""                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.15. AD\_AUTH\_KEYTAB\_FILE

AD\_AUTH\_KEYTAB\_FILE specifies the location of the keytab files. It must be specified as an absolute path.

- Parameter Format

| Property       | Description                                    |
|----------------|------------------------------------------------|
| Parameter Type | String                                         |
| Default Value  | ""                                             |
| Class          | Optional, Adjustable, Dynamic, System          |
| Range          |                                                |
| Syntax         | AD_AUTH_KEYTAB_FILE=filename[:filename [...] ] |

### 1.3.16. AD\_AUTH\_PREFIX

Prefix for AD authenticated user.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | String                               |
| Default Value  | "AD\$"                               |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.17. ALL\_LOGFILE\_MEMBERS\_SANITY\_CHECK

When scanning logfiles during recovery, check all members in a logfile group to check lost or damaged precisely.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.18. AS\_ADDR

AS\_ADDR specifies the network address of the Active Storage instance to be accessed by the database instance.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | String                               |
| Default Value  | "127.0.0.1"                          |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.19. AS\_ALLOW\_ONLY\_RAW\_DISKS

AS\_ALLOW\_ONLY\_RAW\_DISKS specifies whether the Active Storage instance will only discover raw devices.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | Y                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

## 1.3.20. AS\_DDL\_DROP\_FILE

AS\_DDL\_DROP\_FILE specifies whether to delete files that exist in disk space using the DROP clause of the ALTER DISKSPACE statement

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

## 1.3.21. AS\_DISKSTRING

AS\_DISKSTRING specifies the absolute path of the disks that the Active Storage instance discovers.

- Parameter Format

| Property       | Description                                               |
|----------------|-----------------------------------------------------------|
| Parameter Type | String                                                    |
| Default Value  | ""                                                        |
| Class          | Optional, Adjustable, Dynamic, System                     |
| Range          |                                                           |
| Syntax         | AS_DISKSTRING = discovery_string [, discovery_string] ... |

## 1.3.22. AS\_DISK\_REPAIR\_WAIT\_TIME

Waiting time until the disk is automatically dropped when the AS disk monitor finds that the disk fails. In case of TAC environment, setting the same value is recommended (Unit : minutes. If set to 0, disk is not dropped)

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 0                                     |
| Class          | Optional, Adjustable, Dynamic, System |

| Property | Description |
|----------|-------------|
| Range    |             |

### 1.3.23. AS\_PORT

AS\_PORT specifies the TCP port number of the Active Storage instance to be accessed by the database instance.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 0                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          | 1024 - 65535                         |

### 1.3.24. AS\_SCAN\_SSVR\_DISK

AS\_SCAN\_SSVR\_DISK specifies whether or not to use storage server disk.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.25. AS\_WTHR\_CNT

AS\_WTHR\_CNT specifies the number of threads used for Active Storage instance rebalancing. Increasing the number of threads increases rebalancing speeds but consumes more system resources.

- Parameter Format

| Property       | Description |
|----------------|-------------|
| Parameter Type | Integer     |

| Property      | Description                          |
|---------------|--------------------------------------|
| Default Value | 10                                   |
| Class         | Optional, Adjustable, Static, System |
| Range         |                                      |

### 1.3.26. AUDIT\_FILE\_DEST

AUDIT\_FILE\_DEST specifies the directory where AUDIT TRAIL files are stored. It must be specified as an absolute path.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | Depending on system environment       |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.27. AUDIT\_FILE\_SIZE

AUDIT\_FILE\_SIZE specifies the maximum size of each AUDIT TRAIL file. If a single AUDIT TRAIL file grows more than the size specified in this parameter, the current audit trail file will be closed and a new file will be created.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 104857600                             |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 100K - 1G                             |

### 1.3.28. AUDIT\_LOG\_TOTAL\_SIZE\_LIMIT

AUDIT\_LOG\_TOTAL\_SIZE\_LIMIT specifies the maximum total file size that can be stored in the directory set in AUDIT\_LOG\_DEST. Use this parameter to prevent audit log files from growing endlessly by deleting the oldest files when the size of files exceeds the specified value and hence reusing the disk space.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 300 * 1024 * 1024                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 100K - INT64_MAX                      |

### 1.3.29. AUDIT\_SYS\_OPERATIONS

AUDIT\_SYS\_OPERATIONS determines whether to enable auditing for the SYS user.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.30. AUDIT\_TRAIL

AUDIT\_TRAIL specifies the method of auditing when the database auditing function is enabled.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | "NONE"                                |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | NONE   DB   DB_EXTENDED   OS          |

### 1.3.31. AUTHENTICATION\_TIMEOUT

AUTHENTICATION\_TIMEOUT specifies the timeout (in seconds) for client authentication when a client initializes a session. If a client has not completed the authentication protocol in the time specified, the server will close the connection.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 100                                   |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 0 - 100                               |

### 1.3.32. AUTO\_COALESCE

Specifies the automatic coalesce. 0: Do not use. 1: Only LOB indexes. 2: All indexes.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 2                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 0~2                                   |

### 1.3.33. AUTO\_COALESCE\_INTERVAL

AUTO\_COALESCE\_INTERVAL specifies the auto coalesce interval. The unit is 0.1 second.

- Parameter Format

| Property       | Description                                |
|----------------|--------------------------------------------|
| Parameter Type | Integer                                    |
| Default Value  | 100                                        |
| Class          | Optional, Adjustable, Dynamic, System      |
| Range          | 0 ~ 3600000 (OFF / 0.1 second ~ 100 hours) |

### 1.3.34. AUTO\_COALESCE\_STOP

stop automatic coalesce for segment drop

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.35. BATCHUPDATE\_MODE

Set the batch update operation mode.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 1                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.36. BATCH\_INSERT\_ROWCNT\_THRESHOLD

BATCH\_INSERT\_ROWCNT\_THRESHOLD specifies the number of rows for inserting the same information into multiple tables and indexes using the BATCH INSERT protocol.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 5                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.37. BIND\_VARIABLE\_CAPTURE

determines whether to enable Bind Variable Capture.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.38. BIND\_VARIABLE\_CAPTURE\_ALLOC\_COUNT

specifies the number of slave allocators for bind variable capture.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 4                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          | 1 ~ 16                               |

### 1.3.39. BIND\_VARIABLE\_CAPTURE\_INTERVAL

specifies the time interval, in seconds, at which Bind Variable Capture will execute.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 600                                   |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 0 ~ 86400                             |

### 1.3.40. BIND\_VARIABLE\_CAPTURE\_KEEP\_FIRST

determines whether to keep first bind variables of plans

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.41. BIND\_VARIABLE\_CAPTURE\_MAX\_COUNT

specifies the maximum number of bind variables that Bind Variable Capture can store.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 1000                                  |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 10 ~ 1000000                          |

### 1.3.42. BIND\_VARIABLE\_CAPTURE\_MAX\_VAR\_COUNT

specifies the maximum number of bind variables that can be captured from a query.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 512                                   |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 1 ~ 4096                              |

### 1.3.43. BIND\_VARIABLE\_CAPTURE\_POOL\_PCTDEL

a percentage of variables to be deleted from Bind Variable Capture pool.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 10                                   |
| Class          | Optional, Adjustable, Static, System |
| Range          | 5 ~ 20                               |

### 1.3.44. BIND\_VARIABLE\_CAPTURE\_POOL\_SIZE

specifies the size of pool for Bind Variable Capture.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 4*1024*1024                          |
| Class          | Optional, Adjustable, Static, System |
| Range          | 4M ~ TOTAL_SHM_SIZE * 0.02           |

### 1.3.45. BLOCK\_CHANGE\_TRACKING

Specifies a file path for Tiberio managed block change tracking. If specified, block change tracking feature for RMGR incremental backup is ready to be enabled.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | String                               |
| Default Value  | ''                                   |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |
| Syntax         | file                                 |

### 1.3.46. BOOT\_WITH\_AUTO\_DOWN\_CLEAN

BOOT\_WITH\_AUTO\_DOWN\_CLEAN specifies whether or not to automatically enable the tbdwn clean process during tbboot.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | Y                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.47. BOOT\_WITH\_UNKNOWN\_PARAMETER

BOOT\_WITH\_UNKNOWN\_PARAMETER specifies whether to allow booting when unknown parameters are set in the tip file.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.48. BOOT\_WITH\_WALLET\_CHECK

If TTE tablespace exists, DB must be booted with the wallet open beforehand. Recommendation: `tbboot -w`.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | Y                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.49. BTIP\_FILE\_PATH

BTIP\_FILE\_PATH specifies the binary parameter file that can be shared

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | filename                             |
| Default Value  | ""                                   |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.50. CACHE\_SIZE\_FACTOR

simulates the impact of DB\_CACHE\_SIZE increases on the index join plan

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Integer                                |
| Default Value  | 1                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          | 1 ~ 10000                              |

### 1.3.51. CALLSTACK\_DUMP\_BACKUP\_LIMIT\_CNT

Specifies the maximum number of backups that can be stored in CALLSTACK\_DUMP\_DEST. For details, refer to the description of DFLT\_DUMP\_BACKUP\_LIMIT\_CNT. Consequently, the maximum path size may approximate CALLSTACK\_DUMP\_TOTAL\_SIZE\_LIMIT \* (CALLSTACK\_DUMP\_BACKUP\_LIMIT\_CNT + 1).

- Parameter Format

| Property       | Description                                |
|----------------|--------------------------------------------|
| Parameter Type | Integer                                    |
| Default Value  | <a href="#">DFLT_DUMP_BACKUP_LIMIT_CNT</a> |
| Class          | Optional, Adjustable, Dynamic, System      |
| Range          | 2 ~ 500                                    |

### 1.3.52. CALLSTACK\_DUMP\_DEST

CALLSTACK\_DUMP\_DEST specifies the directory to which callstack dump files are stored. It must be specified as an absolute path.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | Depending on system environment       |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.53. CALLSTACK\_DUMP\_TOTAL\_SIZE\_LIMIT

Specifies the maximum total file size that can be stored in the directory specified in CALLSTACK\_DUMP\_DEST, excluding backup directories. For details, refer to the description of DFLT\_DUMP\_TOTAL\_SIZE\_LIMIT.

- Parameter Format

| Property       | Description                                |
|----------------|--------------------------------------------|
| Parameter Type | Integer                                    |
| Default Value  | <a href="#">DFLT_DUMP_TOTAL_SIZE_LIMIT</a> |
| Class          | Optional, Adjustable, Dynamic, System      |
| Range          | 100K - INT64_MAX                           |

### 1.3.54. CCC\_CATH\_CNT

CCC\_CATH\_CNT specifies the number of asynchronous cache task threads used by the CCC(Cluster Cache Control) in parallel.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | MAX(2, MIN(get_ncpu() / 16, 32))     |
| Class          | Optional, Adjustable, Static, System |
| Range          | 1 - INT32_MAX                        |

### 1.3.55. CCC\_MASTERING\_UNIT

CCC\_MASTERING\_UNIT specifies the number of continuous blocks to be specified in a single Master of the CCC (Cluster Cache Control) module. A base of 2 will be raised to the power of the integer specified in this parameter.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 7                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.56. CCC\_RECL\_MAX\_RESOURCES

CCC\_RECL\_MAX\_RESOURCES specifies the maximum number of CCC resource blocks to be maintained in memory.

- Parameter Format

| Property       | Description                                                                                                       |
|----------------|-------------------------------------------------------------------------------------------------------------------|
| Parameter Type | Integer                                                                                                           |
| Default Value  | $(TB\_CEIL\_DIV((_CCC\_RECL\_MAX\_DB\_CACHE\_SIZE / DB\_BLOCK\_SIZE), 100) * \_CCC\_RECL\_MAX\_RESOURCES\_RATIO)$ |
| Class          | Optional, Adjustable, Static, System                                                                              |
| Range          |                                                                                                                   |

### 1.3.57. CCC\_RSBTBL\_SIZE

CCC\_RSBTBL\_SIZE specifies the number of hash table buckets used for the CCC resource block. The number of blocks must be a square of 2.

- Parameter Format

| Property       | Description                                  |
|----------------|----------------------------------------------|
| Parameter Type | Integer                                      |
| Default Value  | $TB\_CEIL\_DB\_BLOCK\_HASH\_BUCKETS/8, 256)$ |
| Class          | Optional, Adjustable, Static, System         |

| Property | Description |
|----------|-------------|
| Range    |             |

### 1.3.58. CC\_CU\_BLKCNT

CC\_CU\_BLKCNT specifies the maximum CC unit size in blocks.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 4                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 1 - 32                                |

### 1.3.59. CC\_CU\_PCTUSE

CC\_CU\_PCTUSE specifies the percentage of a single block that can be used in a compression unit. This will be adjusted while loading.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Integer                                |
| Default Value  | 99                                     |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          | 0 - 99                                 |

### 1.3.60. CC\_CU\_WRITEOUT\_THRESHOLD

CC\_CU\_WRITEOUT\_THRESHOLD specifies the minimum percentage of a single block that must be met to proceed to write out. @range : 0 - 99

- Parameter Format

| Property       | Description |
|----------------|-------------|
| Parameter Type | Integer     |

| Property      | Description                            |
|---------------|----------------------------------------|
| Default Value | 85                                     |
| Class         | Optional, Adjustable, Dynamic, Session |
| Range         |                                        |

### 1.3.61. CC\_EXPECTED\_RATE

CC\_EXPECTED\_RATE specifies the percentage of a single block that can be used for row data. This will be adjusted while loading.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Integer                                |
| Default Value  | 25                                     |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          | 1 - 100                                |

### 1.3.62. CC\_TYPICAL\_ROW\_SIZE

CC\_TYPICAL\_ROW\_SIZE specifies the size of a typical row in a compression unit.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Integer                                |
| Default Value  | 100                                    |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          | 3 - 1000                               |

### 1.3.63. CERTIFICATE\_FILE

CERTIFICATE\_FILE specifies the location of the certificate files. It must be specified as an absolute path.

- Parameter Format

| Property       | Description                                 |
|----------------|---------------------------------------------|
| Parameter Type | String                                      |
| Default Value  | ""                                          |
| Class          | Optional, Adjustable, Static, System        |
| Range          |                                             |
| Syntax         | CERTIFICATE_FILE=filename[:filename [...] ] |

### 1.3.64. CHECK\_RESULT\_FOR\_MEMLOG\_DUMP

CHECK\_RESULT\_FOR\_MEMLOG\_DUMP determines whether to check a result for dumping memory log.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | Y                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.65. CLOSE\_REMOTE\_SESSION\_ON\_SESSION\_RESET

Determines whether to close remote sessions that are connected using a database link when resetting sessions (logical disconnection).

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | Y                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.66. CLUSTER\_DATABASE

CLUSTER\_DATABASE determines whether to configure TAC (Tibero Active Cluster) or not.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Persistent, Static, System |
| Range          |                                      |

### 1.3.67. CMPT\_CNT\_FOR\_RECONFIGURATION

CMPT\_CNT\_FOR\_RECONFIGURATION specifies the number of cluster message processing threads during a reconfiguration. A reconfiguration will be executed in parallel by the threads.

- Parameter Format

| Property       | Description                                     |
|----------------|-------------------------------------------------|
| Parameter Type | Integer                                         |
| Default Value  | MIN(8, MAX(1, <a href="#">ACF_CMPT_CNT</a> /4)) |
| Class          | Optional, Adjustable, Dynamic, System           |
| Range          |                                                 |

### 1.3.68. CM\_AGENT\_THREAD\_CHECK\_EXPIRE

CM\_AGENT\_THREAD\_CHECK\_EXPIRE specifies the expiration period used when the agent thread detects abnormal behavior.

- Parameter Format

| Property       | Description                                 |
|----------------|---------------------------------------------|
| Parameter Type | Integer                                     |
| Default Value  | <a href="#">CM_AGENT_THREAD_INTERVAL</a> *4 |
| Class          | Optional, Persistent, Static, System        |
| Range          |                                             |

### 1.3.69. CM\_AGENT\_THREAD\_INTERVAL

CM\_AGENT\_THREAD\_INTERVAL sets the agent check period.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 10                                   |
| Class          | Optional, Persistent, Static, System |
| Range          |                                      |

### 1.3.70. CM\_AUTO\_RECONNECT\_NODE\_WITH\_ACTIVE\_FHB

If CM\_AUTO\_RECONNECT\_NODE\_WITH\_ACTIVE\_FHB is on, CM automatically reconnects with nodes with active FHB.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.71. CM\_AUTO\_RECONNECT\_TIMEOUT

CM\_AUTO\_RECONNECT\_TIMEOUT specifies the waiting (after sending a reconn msg) timeout for the auto reconnection.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 10                                   |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.72. CM\_AUTO\_REVIVAL

CM\_AUTO\_REVIVAL specifies whether or not to apply the logic for CM and CM\_GUARD to revive one another when the other is killed.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.73. CM\_CLUSTER\_AUTO\_RESTART

CM\_CLUSTER\_AUTO\_RESTART specifies whether or not to automatically restart the cluster when the network resource connection is restored.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.74. CM\_CLUSTER\_WORKER\_CNT

CM\_CLUSTER\_WORKER\_CNT determines the number of workers for cluster when CM\_ENABLE\_CLUSTER\_WORKER is on.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 3                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.75. CM\_CSP\_FENCE\_WAIT\_CNT

Count(seconds) waiting for delay on Instance(EC2, VM, etc) state change in order to determine if i/o fencing is required.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 10                                   |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.76. CM\_DOWN\_ON\_TIME\_DISACCORDANCE

If this is Y, CM will be down on time disaccordance between other CMs.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.77. CM\_ENABLE\_CLUSTER\_WORKER

CM\_ENABLE\_CLUSTER\_WORKER determines whether to use worker for cluster.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | Y                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.78. CM\_ENABLE\_FAST\_ACT\_LOOP

CM\_ENABLE\_FAST\_ACT\_LOOP specifies whether CM action thread can be accelerated by bitq

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | Y                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.79. CM\_ENABLE\_FAST\_INST\_DOWN\_MODE

CM\_ENABLE\_FAST\_INST\_DOWN\_MODE determines whether instances are allowed to be terminated quickly without performing resource cleanup.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.80. CM\_ENABLE\_FAST\_MASTER\_ELECTION

CM\_ENABLE\_FAST\_MASTER\_ELECTION specifies whether to turn on the function to rapidly elect a node as the cluster master when the old master is in a dysfunctional situation.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.81. CM\_ENABLE\_FAST\_NET\_ERROR\_DETECTION

CM\_ENABLE\_FAST\_NET\_ERROR\_DETECTION specifies whether to turn on the function that rapidly recognizes the status change of other CM by detecting a network error.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | Y                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.82. CM\_ENABLE\_SET\_LIMITS\_FOR\_INSTANCE

If CM\_ENABLE\_SET\_LIMITS\_FOR\_INSTANCE is on, CM automatically sets limits after fork process for instance related works.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.83. CM\_ENABLE\_WAIT\_DOWN\_NOTI\_FOR\_SVC

If CM\_ENABLE\_WAIT\_DOWN\_NOTI\_FOR\_SVC is on, CM waits service interrupt until down notify comes from down node.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | Y                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.84. CM\_FENCE

CM\_FENCE determines whether to execute the CM fence daemon. To use this option, CM must be started with root privileges.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Persistent, Static, System |
| Range          |                                      |

### 1.3.85. CM\_FILE\_AUTO\_RESTART\_SSVR

CM\_FILE\_AUTO\_RESTART\_SSVR specifies whether or not to automatically restart a file resource when it is downed in an SSVR environment.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.86. CM\_FORCE\_INSTANCE\_KILL

CM\_FORCE\_INSTANCE\_KILL specifies whether CM forcibly kills the instances when they are needed to be down.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.87. CM\_GUARD\_LOG\_DEST

Save location of the TBCM Guard log file. The location must be set as an absolute path in the local disk.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | String                               |
| Default Value  | ""                                   |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.88. CM\_HEARTBEAT\_EXPIRE

CM\_HEARTBEAT\_EXPIRE specifies the length of the heartbeat expire used to control TBCM. The length must be a multiple of [CM\\_TIME\\_UNIT](#). Heartbeat expire is the reference time that TBCM uses to detect possible failures in other nodes.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 300                                  |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.89. CM\_ID

CM\_ID specifies a unique number for identification.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | -1                                   |
| Class          | Optional, Adjustable, Static, System |
| Range          | 0 - 255                              |

### 1.3.90. CM\_LOCAL\_IP\_VERSION

Configure an IP protocol family to be used for the local CM UI. To use IPv4, set the configuration to 4. To use IPv6, set the configuration to 6.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 4                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          | 4 or 6                               |

### 1.3.91. CM\_LOG\_DEST

CM\_LOG\_DEST specifies the directory to which TBCM log files are stored. It must be specified as an absolute path.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | String                               |
| Default Value  | ""                                   |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.92. CM\_LOG\_FILE\_SIZE

CM\_LOG\_FILE\_SIZE specifies the maximum size of each TBCM log file. If a single log file grows more than the size specified in this parameter, the current log file will be closed and a new file will be created.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 10485760                             |
| Class          | Optional, Adjustable, Static, System |
| Range          | 100K - 1G                            |

### 1.3.93. CM\_LOG\_TOTAL\_SIZE\_LIMIT

CM\_LOG\_TOTAL\_SIZE\_LIMIT specifies the maximum total file size of logs files that can be stored in the directory set in CM\_LOG\_DEST. Use this parameter to prevent database log files from growing endlessly by deleting the oldest files when the file size exceeds the specified value and hence reusing the database space.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 300 * 1024 * 1024                    |
| Class          | Optional, Adjustable, Static, System |
| Range          | 100K - INT64_MAX                     |

### 1.3.94. CM\_LOOP\_TIMEOUT

CM\_LOOP\_TIMEOUT specifies how much is the CTHR/Action thread loop timeout.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 200                                  |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.95. CM\_MODE\_OBSERVER

CM\_MODE\_OBSERVER specifies whether or not to start up CM in observer mode.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.96. CM\_MST\_SAFEGUARD\_EXPIRE

CM\_MST\_SAFEGUARD\_EXPIRE is provided to prevent split brain when the CM\_FAST\_NET\_DETECTION parameter is turned on. When the connection between the master and the slave is lost, turn on the master safeguard. If the master file H/B remains abnormal during this duration, turn off the master safeguard and immediately start the master election. If the master safeguard is turned on and the master file H/B is valid after waiting for 2 \* CM\_MST\_SAFEGUARD, it terminates its own cluster.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 5                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.97. CM\_NAME

CM\_NAME specifies the name which identifies a local node in the cluster environment that consists of TBCM. The name must not be the same as the name of another node and is limited to 31 bytes in length. And a semicolon cannot be used in names. If not specified, it is set to tbcm by default.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ""                                    |
| Class          | Mandatory, Adjustable, Static, System |
| Range          |                                       |

### 1.3.98. CM\_NETWORK\_AUTO\_RESTART

CM\_NETWORK\_AUTO\_RESTART specifies whether or not to automatically restart CM when the network resource connection is restored.

- Parameter Format

| Property       | Description |
|----------------|-------------|
| Parameter Type | Boolean     |

| Property      | Description                          |
|---------------|--------------------------------------|
| Default Value | N                                    |
| Class         | Optional, Adjustable, Static, System |
| Range         |                                      |

### 1.3.99. CM\_NET\_EXPIRE\_MARGIN

CM\_NET\_EXPIRE\_MARGIN specifies the network heartbeat expire for controlling CM. The value must be a multiple of CM\_TIME\_UNIT.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 5                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.100. CM\_OBSERVER\_EXPIRE

CM\_OBSERVER\_EXPIRE specifies the length of the heartbeat expire used to check between TBCM Observer and TBCM. The length must be smaller than summation of IPARAM(CM\_HEARTBEAT\_EXPIRE) and [CM\\_NET\\_EXPIRE\\_MARGIN](#). Heartbeat expire is the reference time that TBCM Observer uses to detect possible failures in other TBCM nodes.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 60                                   |
| Class          | Optional, Persistent, Static, System |
| Range          |                                      |

### 1.3.101. CM\_OBSERVER\_PORT

CM\_OBSERVER\_PORT specified the listening port to be used by CM in observer mode.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 0                                    |
| Class          | Optional, Persistent, Static, System |
| Range          | 1024 - 65535                         |

### 1.3.102. CM\_PORT

CM\_PORT specifies the port number to be used by TBCM. If two or more instances are running on a machine, the port number must be different from other instance.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | <a href="#">LISTENER_PORT</a> + 3    |
| Class          | Optional, Adjustable, Static, System |
| Range          | 1024 - 65535                         |

### 1.3.103. CM\_REGARD\_NETHB\_AS\_SERIOUS

If this parameter is on, Net H/B timeout is regarded as serious as File H/B timeout.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.104. CM\_RESOURCE\_FILE

CM\_RESOURCE\_FILE specifies the path to the file that contains information about the resources registered in CM. It must be specified as an absolute path.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | filename                             |
| Default Value  | ""                                   |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.105. CM\_RESOURCE\_FILE\_BACKUP

CM\_RESOURCE\_FILE\_BACKUP specifies the path to back up the file in the CM\_RESOURCE\_FILE path. It must be specified as an absolute path.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | filename                             |
| Default Value  | ""                                   |
| Class          | Optional, Persistent, Static, System |
| Range          |                                      |

### 1.3.106. CM\_RESOURCE\_FILE\_BACKUP\_INTERVAL

CM\_RESOURCE\_FILE\_BACKUP\_INTERVAL specifies the interval (in minutes) for backing up the file in the CM\_RESOURCE\_FILE path to CM\_RESOURCE\_FILE\_BACKUP path. If set to 0, backup is disabled.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 0                                    |
| Class          | Optional, Persistent, Static, System |
| Range          | 0 - 1440                             |

### 1.3.107. CM\_REVIVAL\_WAIT\_INST\_TIMEOUT

CM\_REVIVAL\_WAIT\_INST\_TIMEOUT specifies the timeout to wait for status info sharing between instances during a force reboot by CM\_Guard.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 30                                   |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.108. CM\_REVIVAL\_WAIT\_NODE\_TIMEOUT

CM\_REVIVAL\_WAIT\_NODE\_TIMEOUT specifies the timeout to wait for resource sharing between nodes during a force reboot by CM\_Guard.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 20                                   |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.109. CM\_REVIVAL\_WAIT\_TIMEOUT

CM\_REVIVAL\_WAIT\_INST\_TIMEOUT specifies the timeout for an existing Guard to wait for the initialization of a new CM during a force reboot by CM\_Guard.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 5                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.110. CM\_SVC\_AUTO\_START\_ON\_BOOT

CM\_SVC\_AUTO\_START\_ON\_BOOT specifies whether or not to automatically restart services together with network and cluster when CM boots.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.111. CM\_SVC\_DOWN\_NOTI\_WAIT\_TIMEOUT

CM\_SVC\_DOWN\_NOTI\_WAIT\_TIMEOUT specifies the waiting (after wait\_down\_noti is set true) timeout for the waiting down noti.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 30                                   |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.112. CM\_TAS\_VM\_ADDR

CM\_TAS\_VM\_ADDR specifies the IP information of the TAS5 Master (volume manager) that is needed by CM when the CM file is stored in TAS5.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | String                               |
| Default Value  | "127.0.0.1"                          |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.113. CM\_TAS\_VM\_PORT

CM\_TAS\_VM\_PORT specifies the PORT information of the TAS5 Master (volume manager) that is needed by CM when the CM file is stored in TAS5.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | String                               |
| Default Value  | "32000"                              |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.114. CM\_TIME\_DISACCORDANCE\_NOTI\_INTERVAL

Periodic warning messages when there is a difference in server UTC time between other CMs. (unit: second)

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 0                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.115. CM\_TIME\_UNIT

CM\_TIME\_UNIT specifies the length of the time tick to be used to control TBCM. Other time-related parameters are represented in a multiple valued set in CM\_TIME\_UNIT and specified based on a 0.1 second time base.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 10                                   |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.116. CM\_UI\_PORT

CM\_UI\_PORT specifies the port information that CM uses to receive user commands.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 0                                     |
| Class          | Mandatory, Persistent, Static, System |
| Range          | 1024 - 65535                          |

### 1.3.117. CM\_USE\_ASENV

CM\_USE\_ASENV specifies whether or not to use the AS commands and environment settings when using cmrctl from CM to control an AS instance.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.118. CM\_USE\_AWS\_EC2

Whether to use amazon AWS EC2 virtual machine.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.119. CM\_USE\_AWS\_EC2\_PRIVATE

Whether to use amazon AWS EC2 virtual machine in private subnet.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.120. CM\_USE\_AWS\_EC2\_PUBLIC

Whether to use amazon AWS EC2 virtual machine in public subnet.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.121. CM\_USE\_AZURE\_VM\_PRIVATE

Whether to use microsoft AZURE VM(virtual machine) in private subnet.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

## 1.3.122. CM\_USE\_KUBERNETES

Whether to use CM on Kubernetes.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

## 1.3.123. CM\_USE\_TAS5\_STORAGE

CM\_USE\_TAS5\_STORAGE specifies whether or not to create CM files in TAS5 storage.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

## 1.3.124. CM\_VIP\_CHECK\_KUBERNETES\_SERVICE\_STATUS

Whether CM check service status via Kubernetes REST API. When this parameter set to N, CM will not check the corresponding service of VIP is currently running or not. Recommend to turn this off when checking takes too much time or too much network bandwidth.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | Y                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.125. CM\_VIP\_FAILOVER\_BASED\_ON\_SVC\_STATUS

CM\_VIP\_FAILOVER\_BASED\_ON\_SVC\_STATUS specifies when to alias VIP (when the service instance is up or down). Y means aliasing when the service instance is up, and N means aliasing when CM is booting.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | Y                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.126. CM\_VIP\_RESTART\_COUNT

CM\_VIP\_RESTART\_COUNT specifies the number of times the VIP resource in the DOWN state performs automatic start. Automatic start is performed once per second.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 1                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.127. CM\_VIP\_STAT\_CHECK\_INTERVAL

CM\_VIP\_STAT\_CHECK\_INTERVAL specifies the interval to check the status of the VIP resource in the UP state.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 1                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.128. CM\_VIP\_USE\_KUBERNETES\_EXTERNAL\_IP

Whether to VIP use external IP on Kubernetes.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.129. CM\_WATCHDOG\_EXPIRE

CM\_WATCHDOG\_EXPIRE specifies the expiration period used when the watchdog function between RDBMS and TBCM is enabled. If TBCM does not run for the specified time, RDBMS will be automatically terminated. The value set in this parameter should be smaller than the one specified in CM\_HEARTBEAT\_EXPIRE.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 290                                  |
| Class          | Optional, Persistent, Static, System |
| Range          |                                      |

### 1.3.130. COMMIT\_FLUSH\_REQUEST

COMMIT\_FLUSH\_REQUEST determines whether to flush the logs after every transaction commit or not.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Boolean                                |
| Default Value  | Y                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          |                                        |

### 1.3.131. COMMIT\_FLUSH\_WAIT

COMMIT\_FLUSH\_WAIT determines whether database commits wait for log flushes or not.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Boolean                                |
| Default Value  | Y                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          |                                        |

### 1.3.132. COMMIT\_POINT\_STRENGTH

COMMIT\_POINT\_STRENGTH specifies the value determining the commit point site in a distributed transaction.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 1                                    |
| Class          | Optional, Persistent, Static, System |
| Range          | 0 - 255                              |

### 1.3.133. CONTROL\_FILES

CONTROL\_FILES specifies the location of the control files. It must be specified as an absolute path.

- Parameter Format

| Property       | Description                              |
|----------------|------------------------------------------|
| Parameter Type | String                                   |
| Default Value  | ""                                       |
| Class          | Mandatory, Adjustable, Static, System    |
| Range          | 1 - 8 files                              |
| Syntax         | CONTROL_FILES=filename[:filename [...] ] |

### 1.3.134. CPU\_COUNT

CPU\_COUNT specifies the number of CPUs allowed for this instance.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 0                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.135. CREATE\_STORED\_OUTLINES

Determines whether to create outline.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Boolean                                |
| Default Value  | N                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          |                                        |

### 1.3.136. CRYPTO\_LEVEL

Establishing the encryption scheme for user password storage. (0: MD5, 1: SHA256)

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 1                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 0 - 1                                 |

### 1.3.137. CTHR\_CHECK\_CONNECTION\_INTERVAL

CTHR\_CHECK\_CONNECTION\_INTERVAL specifies the time interval (in seconds) that CTHR checks client connections.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 1                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 1 - 120 (s)                           |

### 1.3.138. CURSOR\_SHARING

CURSOR\_SHARING determines the kind of SQL statement which can share the same cursor. EXACT: Cursors can be shared only when the SQL text matches exactly. FORCE: Setting this parameter to FORCE forces similar SQL statements to share cursors as long as only their literals are different.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | String                                 |
| Default Value  | "EXACT"                                |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          | EXACT   FORCE                          |

### 1.3.139. CWS\_DDD\_TIMEOUT\_ON\_SLOG

CWS\_DDD\_TIMEOUT\_ON\_SLOG specifies whether or not to record a 'DDD timedout' SLOG message.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.140. CWS\_RECL\_MAX\_RESOURCES

CWS\_RECL\_MAX\_RESOURCES specifies the maximum number of CWS (Cluster Wait-lock Service) resource blocks to be maintained in memory.

- Parameter Format

| Property       | Description                                                               |
|----------------|---------------------------------------------------------------------------|
| Parameter Type | Integer                                                                   |
| Default Value  | (TB_CEIL_DIV(_WLOCK_POOL_INIT_SIZE, 100) * _CWS_RECL_MAX_RESOURCES_RATIO) |
| Class          | Optional, Adjustable, Static, System                                      |
| Range          |                                                                           |

### 1.3.141. CWS\_RESOLVE\_CVT\_DEADLK\_FORCE

CWS\_RESOLVE\_CVT\_DEADLK\_FORCE specifies whether or not to enable CWS (Cluster Wait-lock Service) to force cancellation of requests that could potentially cause a conversion deadlock.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | Y                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.142. CWS\_RSBTBL\_SIZE

CWS\_RSBTBL\_SIZE specifies the number of hash table buckets used for the CWS resource block. The number of buckets must be a square of 2.

- Parameter Format

| Property       | Description                                  |
|----------------|----------------------------------------------|
| Parameter Type | Integer                                      |
| Default Value  | (_WLOCK_BUCKET_PER_SET*_WLOCK_BUCKETSET_CNT) |
| Class          | Optional, Adjustable, Static, System         |
| Range          |                                              |

### 1.3.143. DATABASE\_REPLAY\_CAPTURE\_DEST

directory of capture files used for database replay

- Parameter Format

| Property       | Description                                  |
|----------------|----------------------------------------------|
| Parameter Type | String                                       |
| Default Value  | ""                                           |
| Class          | Optional, Adjustable, Dynamic, System        |
| Range          |                                              |
| Syntax         | DATABASE_REPLAY_CAPTURE_DEST="/[DIRECTORY]/" |

### 1.3.144. DATABASE\_REPLAY\_CAPTURE\_DIR\_LIMIT

specifies the maximum size of all capture dump files accumulated in the directory specified as DATABASE\_REPLAY\_CAPTURE\_DEST. Agent process monitors if or not the directory is too large. If size of the directory is as big as this parameter value, then stop capturing.

- Parameter Format

| Property       | Description                                                                |
|----------------|----------------------------------------------------------------------------|
| Parameter Type | Integer                                                                    |
| Default Value  | ((MAX_SESSION_COUNT) * (int64_t)(DATABASE_REPLAY_CAPTURE_FILE_SIZE)) * 100 |
| Class          | Optional, Adjustable, Static, System                                       |
| Range          | 100K ~ UINT64_MAX                                                          |

### 1.3.145. DATABASE\_REPLAY\_CAPTURE\_FILE\_SIZE

The Maximum size of capture dump file for database replay.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 10485760                             |
| Class          | Optional, Adjustable, Static, System |

| Property | Description |
|----------|-------------|
| Range    | 100K ~ 1G   |

### 1.3.146. DATABASE\_REPLAY\_CAPTURE\_WORKLOAD

whether or not capture workload in tiberio server. All requests from the client program are captured per session without exception. During capture, DB performance may be slightly degraded due to file I/O. Rather than dynamically modifying parameter, it is recommended to shutdown DB and restart after setting parameter. It doesn't work in Active Storage and Storage Server mode. Before starting the capture, the capture destination must be set.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.147. DATABASE\_REPLAY\_DUMP\_MMAP\_SIZE

The MMAP page size to be used for MMAP when leaving the workload capture. The page size to be used in MMAP should be multiples of the OS page size. (usually 4K) It is better to set the size large enough.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 65536                                |
| Class          | Optional, Adjustable, Static, System |
| Range          | 4K ~ 10M                             |

### 1.3.148. DBLINK\_BATCH\_UPDATE\_ROW\_COUNT

Number of rows to be processed at the same time during batch update through a database link. If set to the default value 0, batch update is disabled.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Integer                                |
| Default Value  | 0                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          |                                        |

### 1.3.149. DBLINK\_CONN\_TIMEOUT

Number of seconds for a CLI connect to timeout to a DB that will be connected through a DB link.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Integer                                |
| Default Value  | 100                                    |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          | 0 - INT32_MAX                          |

### 1.3.150. DBLINK\_CONN\_VALIDATION\_SKIP\_INTERVAL

Interval to check whether a DB link is valid after the last message to a connected DB has been sent through a DB link.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Integer                                |
| Default Value  | 0                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          | 0 - INT32_MAX                          |

### 1.3.151. DBLINK\_QUERY\_TIMEOUT

Number of seconds to wait for a response after sending SQL to a DB that is connected through a DB link.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Integer                                |
| Default Value  | 0                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          | 0 - INT32_MAX                          |

### 1.3.152. DBMS\_LOG\_DEST

DBMS\_LOG\_DEST specifies the directory where the database log files are stored. It must be specified as an absolute path.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | Depending on system environment       |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.153. DBMS\_LOG\_FILE\_SIZE

DBMS\_LOG\_FILE\_SIZE specifies the maximum size of each database log file. If a single log file grows more than the size specified in this parameter, the current log file will be closed and a new file will be created.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 104857600                             |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 100K - 1G                             |

### 1.3.154. DBMS\_LOG\_TOTAL\_SIZE\_LIMIT

DBMS\_LOG\_TOTAL\_SIZE\_LIMIT specifies the maximum total file size that can be stored in the directory set in DBMS\_LOG\_DEST. Use this parameter to prevent database log files from growing endlessly by deleting the oldest files when the size of files exceeds the specified value and hence reusing the disk space.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 300 * 1024 * 1024                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 100K - INT64_MAX                      |

### 1.3.155. DBWR\_CNT

DBWR\_CNT specifies the number of threads that use database blocks on disk.

- Parameter Format

| Property       | Description                                    |
|----------------|------------------------------------------------|
| Parameter Type | Integer                                        |
| Default Value  | !_ENABLE_MULTI_DBWR ? 1 : MAX(1, get_ncpu()/4) |
| Class          | Optional, Adjustable, Static, System           |
| Range          | 1 - 1024                                       |

### 1.3.156. DBWR\_USE\_AIO

Option for DBWR to use AIO. Must be set to N in active storage mode.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

## 1.3.157. DB\_BLOCK\_SIZE

DB\_BLOCK\_SIZE specifies the size of database blocks.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 8 * 1024                             |
| Class          | Optional, Persistent, Static, System |
| Range          | 2K - 32K                             |

## 1.3.158. DB\_CACHE\_SIZE

DB\_CACHE\_SIZE specifies the size of the database cache buffer. It can be adjusted so that each DB writer manages the same number of blocks.

- Parameter Format

| Property       | Description                                                                                                                                                                                                                                                    |
|----------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Parameter Type | Integer                                                                                                                                                                                                                                                        |
| Default Value  | ( <a href="#">CLUSTER_DATABASE</a> ? TB_CEIL(( <a href="#">TOTAL_SHM_SIZE</a> - <a href="#">INMEMORY_SIZE</a> ) * 1 / 2, 1) : TB_CEIL(( <a href="#">TOTAL_SHM_SIZE</a> - <a href="#">INMEMORY_SIZE</a> ) * 2 / 3, 1)) - <a href="#">MEMORY_TABLESPACE_SIZE</a> |
| Class          | Optional, Adjustable, Static, System                                                                                                                                                                                                                           |
| Range          |                                                                                                                                                                                                                                                                |

## 1.3.159. DB\_CREATE\_FILE\_DEST

DB\_CREATE\_FILE\_DEST specifies the directory where the data files are stored. It must be specified as an absolute path.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | String                               |
| Default Value  | ""                                   |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.160. DB\_FILE\_MULTIBLOCK\_READ\_COUNT

DB\_FILE\_MULTIBLOCK\_READ\_COUNT specifies the number of blocks which can be read in one I/O operation.

- Parameter Format

| Property       | Description                                   |
|----------------|-----------------------------------------------|
| Parameter Type | Integer                                       |
| Default Value  | <a href="#">USE_ACTIVE_STORAGE</a> ? 128 : 32 |
| Class          | Optional, Adjustable, Dynamic, Session        |
| Range          | 1-1024                                        |

### 1.3.161. DB\_KEEP\_CACHE\_SIZE

DB\_KEEP\_CACHE\_SIZE specifies the size of the KEEP buffer pool.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 0                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.162. DB\_MULTIBLOCK\_PREFETCH\_COUNT

DB\_MULTIBLOCK\_PREFETCH\_COUNT specifies prefetch count

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 1                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

## 1.3.163. DB\_NAME

DB\_NAME specifies the name of the database. It must correspond to the name specified in the DB\_NAME parameter of the \$TB\_SID.tip file. If a database name is not specified, the database will be created with the TB\_SID environment variable. When the system is restarted after database creation, an error may occur if the specified database name does not match the one specified in DB\_NAME of the \$TB\_SID.tip file.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ""                                    |
| Class          | Mandatory, Persistent, Static, System |
| Range          | Limited to 40 characters              |

## 1.3.164. DB\_RECYCLE\_CACHE\_SIZE

DB\_RECYCLE\_CACHE\_SIZE specifies the size of the RECYCLE buffer pool.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 0                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

## 1.3.165. DB\_UNIQUE\_NAME

DB\_UNIQUE\_NAME specifies a globally unique name for the database. Databases with the same DB\_NAME, for example, copies of a database created for testing or physical standby databases, must have a unique DB\_UNIQUE\_NAME to be distinguished. Every database's DB\_UNIQUE\_NAME must be unique within the enterprise. @range : Limited to 40 characters and is case sensitive

- Parameter Format

| Property       | Description |
|----------------|-------------|
| Parameter Type | String      |

| Property      | Description                          |
|---------------|--------------------------------------|
| Default Value | DB_NAME                              |
| Class         | Optional, Persistent, Static, System |
| Range         |                                      |

### 1.3.166. DD\_CACHE\_MAX

DD\_CACHE\_MAX specifies the maximum size of DD cache (automatically adjusted). Deprecated.

- Parameter Format

| Property       | Description                                           |
|----------------|-------------------------------------------------------|
| Parameter Type | Integer                                               |
| Default Value  | MAX(10 << 20, (TOTAL_SHM_SIZE - DB_CACHE_SIZE) * 0.3) |
| Class          | Optional, Adjustable, Dynamic, System                 |
| Range          | 10M -                                                 |

### 1.3.167. DEFAULT\_SEQ\_CACHE\_SIZE

DEFAULT\_SEQ\_CACHE\_SIZE specifies the default value of sequence cache size.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 20                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 2 - 99999999                          |

### 1.3.168. DFLT\_DUMP\_BACKUP\_LIMIT\_CNT

Specifies the maximum number of backups that can be stored in \*\_DUMP\_DEST. This parameter prevents dump file backup directories from unlimited growth. When the backup count exceeds the specified value, the oldest backup files are removed to allow disk space reuse. Backup operations are performed under the following conditions: 1. Files exceed the maximum size limit 2. Database restart occurs Consequently, the maximum path size may approximate DFLT\_DUMP\_TOTAL\_SIZE\_LIMIT \* (DFLT\_DUMP\_BACKUP\_LIMIT\_CNT + 1). Used as the default value for TRACE\_DUMP\_BACKUP\_LIMIT\_CNT, DI

AG\_DUMP\_BACKUP\_LIMIT\_CNT, CALLSTACK\_DUMP\_BACKUP\_LIMIT\_CNT and  
ACT\_DUMP\_BACKUP\_LIMIT\_CNT.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 10                                   |
| Class          | Optional, Adjustable, Static, System |
| Range          | 2 ~ 500                              |

### 1.3.169. DFLT\_DUMP\_TOTAL\_SIZE\_LIMIT

Specifies the maximum total file size that can be stored in the directory specified in \*\_DUMP\_DEST, excluding backup directories. When the size of files exceeds the specified value, dump files are relocated to new backup directories. Used as the default value for TRACE\_DUMP\_TOTAL\_SIZE\_LIMIT, DIAG\_DUMP\_TOTAL\_SIZE\_LIMIT, CALLSTACK\_DUMP\_TOTAL\_SIZE\_LIMIT and ACT\_DUMP\_TOTAL\_SIZE\_LIMIT.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 100 * 1024 * 1024                    |
| Class          | Optional, Adjustable, Static, System |
| Range          | 100K - INT64_MAX                     |

### 1.3.170. DF\_HDRBLK\_START\_OFFSET

DF\_HDRBLK\_START\_OFFSET specifies the data file's start offset.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 0                                    |
| Class          | Optional, Persistent, Static, System |
| Range          | 0 - 65536                            |

### 1.3.171. DIAG\_DUMP\_BACKUP\_LIMIT\_CNT

Specifies the maximum number of backups that can be stored in DIAG\_DUMP\_DEST. For details, refer to the description of DFLT\_DUMP\_BACKUP\_LIMIT\_CNT. Consequently, the maximum path size may approximate  $\text{DIAG\_DUMP\_TOTAL\_SIZE\_LIMIT} * (\text{DIAG\_DUMP\_BACKUP\_LIMIT\_CNT} + 1)$ .

- Parameter Format

| Property       | Description                                |
|----------------|--------------------------------------------|
| Parameter Type | Integer                                    |
| Default Value  | <a href="#">DFLT_DUMP_BACKUP_LIMIT_CNT</a> |
| Class          | Optional, Adjustable, Dynamic, System      |
| Range          | 2 ~ 500                                    |

### 1.3.172. DIAG\_DUMP\_DEST

Specifies the directory where the diagnostic dump files are stored. Must be specified as an absolute path.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | Depending on system environment       |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.173. DIAG\_DUMP\_TOTAL\_SIZE\_LIMIT

Specifies the maximum total file size that can be stored in the directory specified in DIAG\_DUMP\_DEST, excluding backup directories. For details, refer to the description of DFLT\_DUMP\_TOTAL\_SIZE\_LIMIT.

- Parameter Format

| Property       | Description                                |
|----------------|--------------------------------------------|
| Parameter Type | Integer                                    |
| Default Value  | <a href="#">DFLT_DUMP_TOTAL_SIZE_LIMIT</a> |
| Class          | Optional, Adjustable, Dynamic, System      |
| Range          | 100K - INT64_MAX                           |

## 1.3.174. DIAG\_LOG\_DEST

Specifies the directory where the diagnostic log files are stored. Must be specified as an absolute path.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | Depending on system environment       |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

## 1.3.175. DIAG\_LOG\_FILE\_SIZE

Specifies the maximum size of each diagnostic log file. When a single diagnostic log file exceeds the size specified in this parameter, the current diagnostic log file is closed and a new file is created.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 100 * 1024 * 1024                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 100K - 1G                             |

## 1.3.176. DIAG\_LOG\_TOTAL\_SIZE\_LIMIT

Specifies the maximum total file size that can be stored in the directory specified in DIAG\_LOG\_DEST. This parameter prevents diagnostic log files from unlimited growth. When the size of files exceeds the specified value, the oldest files are removed to allow disk space reuse.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 500 * 1024 * 1024                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 100K - INT64_MAX                      |

### 1.3.177. DIRTY\_BLOCK\_WRITE\_ISSUE\_TIME

Time period within which a dirty cache block must be issued to be written to disk. (Unit: second) This cannot be used concurrently with "LOG\_CHECKPOINT\_TIMEOUT". If set to 0, this parameter is ignored.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 0                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.178. DISK\_AFFINITY\_GROUP

AS\_USE\_DISK\_AFFINITY specifies the affinity group name to be used by instance. It must be equal to one of the disk failgroup names.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | String                               |
| Default Value  | ""                                   |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.179. DISK\_PATH

Sets the paths to AS disks. Use absolute path.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | String                               |
| Default Value  | ""                                   |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |
| Syntax         | DISK_PATH=pathname[:pathname [...] ] |

## 1.3.180. DONT\_USE\_FILE\_BUFFERING

DONT\_USE\_FILE\_BUFFERING determines whether to use file cache in Windows.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

## 1.3.181. DPBUF\_BLOCK\_COUNT

DPBUF\_BLOCK\_COUNT specifies the number of blocks written per Direct Path I/O operation.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Integer                                |
| Default Value  | 64                                     |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          | 2 - 2048                               |

## 1.3.182. DPL\_INDEX\_METHOD

DPL\_INDEX\_METHOD specifies the DPL (Direct Path Loading) operation mode in the table with the index.

1: MULTI INSERT, multiple index keys are inserted at a time. 2: FAST BUILD, in

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Integer                                |
| Default Value  | 1                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          | 1   2                                  |

### 1.3.183. DP\_USE\_O\_SYNC

Specifies whether to use the O\_SYNC flag during DP operations, instead of using fsync.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.184. DUMP\_EXSTAT\_ON\_STMT\_CANCEL

DUMP\_EXSTAT\_ON\_STMT\_CANCEL specifies whether or not to dump the current SQL, bind parameters and execution stat when a statement is cancelled

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Boolean                                |
| Default Value  | N                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          |                                        |

### 1.3.185. DUMP\_PATH\_CHECK\_INTERVAL

Monitoring cycle for cumulative space usage of dump paths excluding backup sets. (unit: 1 second)  
(Target: CALLSTACK, ACT, TRACEDUMP)

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 600                                   |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 60 ~ 3600                             |

## 1.3.186. ENABLE\_BCT\_AT\_BOOTING

To always enable block change tracking feature at booting, without the corresponding DDL.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |
| Syntax         | file                                 |

## 1.3.187. ENABLE\_MULTI\_LOG\_BUFFER

ENABLE\_MULTI\_LOG\_BUFFER specifies whether to enable the multi log buffer feature.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

## 1.3.188. ENCRYPTION\_WALLET\_FILE

ENCRYPTION\_WALLET\_FILE specifies the location of wallet files used by TDE only. It must be specified as an absolute path. deprecated

- Parameter Format

| Property       | Description                                      |
|----------------|--------------------------------------------------|
| Parameter Type | String                                           |
| Default Value  | ''                                               |
| Class          | Optional, Adjustable, Static, System             |
| Range          |                                                  |
| Syntax         | ENCRYPTION_WALLET_FILE=filename[:filename [...]] |

### 1.3.189. EXTRA\_LISTENER\_IPS

Additional IP addresses used by the listener other than the primary IP specified with LISTENER\_IP. Applied only when LISTNER\_IP is set. Use a semi-colon, ';', as the delimiter.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | String                               |
| Default Value  | "-1"                                 |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.190. EXTRA\_LISTENER\_PORTS

Multiple ports can be specified to configure additional ports. Use a semi-colon, ';', as the delimiter, e.g., '8900;8901;'.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | String                               |
| Default Value  | ""                                   |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.191. EXTRA\_LISTENER\_PORTS\_6

Multiple ipv6 ports can be specified to configure additional ports. Use a semi-colon, ';', as the delimiter, e.g., '8900;8901;'.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | String                               |
| Default Value  | ""                                   |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

## 1.3.192. EX\_MEMORY\_AUTO\_MANAGEMENT

EX\_MEMORY\_AUTO\_MANAGEMENT enables either automatic PGA memory allocation with memory tuner or manual memory allocation.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Boolean                                |
| Default Value  | Y                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          |                                        |

## 1.3.193. EX\_MEMORY\_COMPENSATE\_INTERVAL

The memory tuner uses the interval specified in this parameter to check the total amount of available PGA memory and the maximum amount of PGA memory available to each operation. The interval is specified in 0.1 seconds.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 30                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | EX_MEMORY_DRIFT_CHECK_INTERVAL -      |

## 1.3.194. EX\_MEMORY\_DRIFT\_CHECK\_INTERVAL

To cope with rapid changes in the amount of memory, RDBMS checks whether changes in the amount of PGA memory has exceeded 10% of the total amount of PGA memory at every interval specified in this parameter. The interval is specified in 0.1 seconds.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 2                                     |
| Class          | Optional, Adjustable, Dynamic, System |

| Property | Description |
|----------|-------------|
| Range    | 1 -         |

### 1.3.195. EX\_MEMORY\_DRIFT\_RATIO

The memory difference RATIO for DRIFT CHECK. DRIFT will be generated when the change is  $EX\_MEMORY\_SOFT\_LIMIT * EX\_MEMORY\_DRIFT\_RATIO$ . (unit of 0.001%)

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 10000                                 |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 1 ~ 100000                            |

### 1.3.196. EX\_MEMORY\_SOFT\_LIMIT\_RATIO

The ex memory operation limit is proportional to the soft limit of available PGA memory. This parameter specifies the ratio of the soft limit in percentage to the available PGA memory.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 90                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 10 - 100                              |

### 1.3.197. FDPOOL\_HASH\_SIZE

FDPOOL\_HASH\_SIZE specifies the hash size to be used in the fdpool.

- Parameter Format

| Property       | Description |
|----------------|-------------|
| Parameter Type | Integer     |

| Property      | Description                                                                          |
|---------------|--------------------------------------------------------------------------------------|
| Default Value | MIN( MAX(WTHR_PROC_CNT * WTHR_PER_PROC * 4 * (!!CLUSTER_DATABASE + 1), 128), 100000) |
| Class         | Optional, Adjustable, Static, System                                                 |
| Range         |                                                                                      |

### 1.3.198. FLASHBACK\_LOG\_ARCHIVE\_DEST

Specifies the absolute path of the directory where the flashback archive logs.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ''                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.199. FLASHBACK\_LOG\_ARCHIVE\_FORMAT

Specifies the format of the flashback archive log file.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | "fb-t%t-r%r-s%s.arc"                  |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.200. FLASHBACK\_LOG\_BUFFER

Size of the memory space where flashback logs are buffered. Value must be set when flashback logging is set on. Recommended twice the value of LOG\_BUFFER.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 0                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          | (DB_BLOCK_SIZE) * 4 - 1G             |

### 1.3.201. FLASHBACK\_MARKER\_INTERVAL

Time period of making flashback database marker (in sec). Smaller it is, faster the DB can flashback. However, DB gets slower due to the heavy flashback logging during usual operation and also flashback logs takes up more disk space.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 86400                                 |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.202. GATHER\_SQL\_EXEC\_TIME

GATHER\_SQL\_EXEC\_TIME determines whether to measure the time spent to execute SQL statements. This parameter is used to enable EXEC\_TIME of \_VT\_OPEN\_CURSOR.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Boolean                                |
| Default Value  | N                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          |                                        |

### 1.3.203. GATHER\_SQL\_PLAN\_STAT

GATHER\_SQL\_PLAN\_STAT determines whether to collect statistics on each node in the SQL statement execution plan at the time an SQL query is executed.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Boolean                                |
| Default Value  | N                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          |                                        |

### 1.3.204. HASH\_AREA\_SIZE

HASH\_AREA\_SIZE specifies the maximum size of memory to be used for creating a hash table. If HASH\_AREA\_SIZE is large, hash joins can be completed in memory in a one-pass hash join. If HASH\_AREA\_SIZE is not large enough for a particular hash join, a two-pass hash join will be executed.

- Parameter Format

| Property       | Description                                            |
|----------------|--------------------------------------------------------|
| Parameter Type | Integer                                                |
| Default Value  | (MEMORY_TARGET - TOTAL_SHM_SIZE) * 0.3                 |
| Class          | Optional, Adjustable, Dynamic, Session                 |
| Range          | (DB_BLOCK_SIZE * DFLT_DB_FILE_MULTIBLOCK_READ_COUNT) - |

### 1.3.205. HBASE\_CONNECTOR\_SCAN\_CACHE

HBASE\_CONNECTOR\_SCAN\_CACHE specifies the HBase Connector scanner row size.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 1000                                  |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.206. HSM\_LIB\_FILE

Absolute library path with the library file name. It is used for linking with an HSM device.

- Parameter Format

| Property       | Description                             |
|----------------|-----------------------------------------|
| Parameter Type | String                                  |
| Default Value  | ""                                      |
| Class          | Optional, Adjustable, Static, System    |
| Range          |                                         |
| Syntax         | HSM_LIB_FILE=filename[:filename [...] ] |

### 1.3.207. HSM\_NOT\_USE\_WRAPPING

Option to use the PKCS #11 wrapping key option for linking an HSM device.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.208. HSM\_TTE\_KEY\_BACKUP\_FILE\_PATH

HSM\_TTE\_KEY\_BACKUP\_FILE\_PATH specifies/determines the location of the tte key backup file. It must be specified as an absolute path.

- Parameter Format

| Property       | Description                                             |
|----------------|---------------------------------------------------------|
| Parameter Type | String                                                  |
| Default Value  | ""                                                      |
| Class          | Optional, Adjustable, Static, System                    |
| Range          |                                                         |
| Syntax         | HSM_TTE_KEY_BACKUP_FILE_PATH=filename[:filename [...] ] |

## 1.3.209. IFF\_STORAGE\_PROCESSING

whether to enable INDEX FAST FULL STORAGE

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Boolean                                |
| Default Value  | N                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          |                                        |

## 1.3.210. ILOG\_BACKUP\_SIZE\_LIMIT

Specifies the maximum size of all internal backup directories accumulated in the directory specified with the ILOG\_DEST parameter. When the size of accumulated backup directories exceeds this limit, the oldest backup directory is deleted to prevent an infinite accumulation of log files.

- Parameter Format

| Property       | Description                                               |
|----------------|-----------------------------------------------------------|
| Parameter Type | Integer                                                   |
| Default Value  | ((MAX_SESSION_COUNT + 150L) * (uint64_t)(ILOG_FILE_SIZE)) |
| Class          | Optional, Adjustable, Dynamic, System                     |
| Range          | 0 ~ UINT64_MAX                                            |

## 1.3.211. ILOG\_DEST

ILOG\_DEST specifies the directory in which the internal log file is saved. The absolute path should be specified.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | String                               |
| Default Value  | Depending on system environment      |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.212. ILOG\_FILE\_BACKUP\_ON\_BOOT

ILOG\_FILE\_BACKUP\_ON\_BOOT determines if old internal log files will be backed up when RDBMS boots. If it is set to 'N', internal log records will continue to be written to an existing internal log file. Otherwise, all old internal log files will be backed up to a new backup directory.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | Y                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.213. ILOG\_FILE\_SIZE

ILOG\_FILE\_SIZE specifies the maximum size of the internal log file. If the size of a log file is larger than this size, the current file is backed up and a new log file is created. It must be multiples of the ILOG\_MMAP\_SIZE.

- Parameter Format

| Property       | Description                                          |
|----------------|------------------------------------------------------|
| Parameter Type | Integer                                              |
| Default Value  | TB_CEIL(512 * 1024, <a href="#">ILOG_MMAP_SIZE</a> ) |
| Class          | Optional, Adjustable, Dynamic, System                |
| Range          | 100K ~ 1G                                            |

### 1.3.214. ILOG\_KEEP\_BACKUP\_ASIDE

Specifies whether to manage the ilog backup directory size separately. Y: If set to Y, the backup directory size is not included in the ILOG\_TOTAL\_SIZE\_LIMIT setting. The backup directory size will be separately managed using the ILOG\_BACKUP\_SIZE\_LIMIT setting. N: If set to N, the backup directory size is included in the ILOG\_TOTAL\_SIZE\_LIMIT setting. ILOG\_TOTAL\_SIZE\_LIMIT is the maximum size of all internal log files accumulated in the ILOG\_DEST directory.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.215. ILOG\_MAP

ILOG\_MAP specifies the location of the configuration file which specifies on/off of the internal log. It must be specified as an absolute path.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ""                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.216. ILOG\_MMAP\_SIZE

Specifies the MMAP page size to be used for MMAP when leaving the internal log. The page size to be used in MMAP should be multiples of the OS page size (usually 4K). It is better to set the size large enough.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 64 * 1024                            |
| Class          | Optional, Adjustable, Static, System |
| Range          | 4K ~ 10M                             |

### 1.3.217. ILOG\_TOTAL\_SIZE\_LIMIT

Specifies the maximum aggregate size of internal log files stored in the directory specified by ILOG\_DEST parameter. Inclusion of backup directory size in this calculation is determined by the ILOG\_KEEP\_BACKUP\_ASIDE configuration. When accumulated file size exceeds this threshold, the system automatically purges oldest backup files to prevent unbounded growth of the log directory. This parameter value must exceed the product of ILOG\_FILE\_SIZE and the total thread count.

- Parameter Format

| Property       | Description                                                                                                                               |
|----------------|-------------------------------------------------------------------------------------------------------------------------------------------|
| Parameter Type | Integer                                                                                                                                   |
| Default Value  | $((WTHR\_PROC\_CNT + 1) * WTHR\_PER\_PROC + PEP\_PROC\_CNT * WTHR\_PER\_PEP + ACF\_CMPT\_CNT + 150L) * (uint64\_t)(ILOG\_FILE\_SIZE) * 2$ |
| Class          | Optional, Adjustable, Dynamic, System                                                                                                     |
| Range          | 100K ~ UINT64_MAX                                                                                                                         |

### 1.3.218. INMEMORY\_QUERY

Specifies whether in-memory columns allowed.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Boolean                                |
| Default Value  | Y                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          |                                        |

### 1.3.219. INMEMORY\_SIZE

Specifies the memory size allocated to the in-memory column store.

- Parameter Format

| Property       | Description |
|----------------|-------------|
| Parameter Type | Integer     |
| Default Value  | 0           |

| Property | Description                          |
|----------|--------------------------------------|
| Class    | Optional, Adjustable, Static, System |
| Range    | 100M -                               |

### 1.3.220. INSTANCE\_NUMBER

INSTANCE\_NUMBER specifies the instance number.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | THREAD                               |
| Class          | Optional, Adjustable, Static, System |
| Range          | 0 - 1023                             |

### 1.3.221. INSTANCE\_TYPE

INSTANCE\_TYPE specifies whether an instance is a database instance, an Active Storage instance, or a Storage Server instance.

- Parameter Format

| Property       | Description                                    |
|----------------|------------------------------------------------|
| Parameter Type | String                                         |
| Default Value  | "TIBERO"                                       |
| Class          | Optional, Persistent, Static, System           |
| Range          | TIBERO   DATABASE   AS   ACTIVE_STORAGE   SSVR |

### 1.3.222. IP\_VERSION

Configure an IP protocol family to be used for network communications with a client. To use IPv4, set the configuration to 4. To use IPv6, set the configuration to 6. To automatically select and use an available protocol, set the configuration to 0. If use both IPv4 and IPv6, set the configuration to 10.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 4                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          | 0, 4, 6, or 10                       |

### 1.3.223. JAVA\_CLASS\_PATH

JAVA\_CLASS\_PATH specifies the default directory used to store a class file in Java stored procedure. It must be specified as an absolute path. If not specified, the class file will be stored in DB\_CREATE\_FILE\_DEST/java.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | String                               |
| Default Value  | ''''                                 |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.224. JOB\_HISTORY\_LOG

JOB\_HISTORY\_LOG determines whether to write a job history after every job start or not.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | Y                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.225. JOB\_SCHEDULING\_MAX\_TIMEOUT

JOB\_SCHEDULING\_MAX\_TIMEOUT specifies the maximum interval of time (in seconds) to execute the JOB SCHEDULER.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 3600                                  |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 1 ~ 999999                            |

### 1.3.226. KEEP\_STAT\_PER\_SQL

KEEP\_STAT\_PER\_SQL determines whether to maintain statistical information on execution time for each SQL statement.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.227. LGWR\_USE\_AIO

Option for LGWR to use AIO. Must be set to N in active storage mode.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.228. LIBRARY\_CACHE\_MAX

LIBRARY\_CACHE\_MAX specifies the maximum size of library cache (automatically adjusted). Deprecated.

- Parameter Format

| Property       | Description                                           |
|----------------|-------------------------------------------------------|
| Parameter Type | Integer                                               |
| Default Value  | MAX(10 << 20, (TOTAL_SHM_SIZE - DB_CACHE_SIZE) * 0.3) |
| Class          | Optional, Adjustable, Dynamic, System                 |
| Range          | 10M -                                                 |

### 1.3.229. LIBRARY\_DD\_CACHE\_MAX

LIBRARY\_DD\_CACHE\_MAX specifies the maximum size of library cache and DD cache (automatically adjusted).

- Parameter Format

| Property       | Description                                           |
|----------------|-------------------------------------------------------|
| Parameter Type | Integer                                               |
| Default Value  | MAX(10 << 20, (TOTAL_SHM_SIZE - DB_CACHE_SIZE) * 0.6) |
| Class          | Optional, Adjustable, Dynamic, System                 |
| Range          | 10M -                                                 |

### 1.3.230. LISTENER\_IP

LISTENER\_IP specifies the IP used by listener. IPv4 addresses are supported. The primary IP address for SSL and special port bindings. This IP address cannot be deleted.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | String                               |
| Default Value  | "-1"                                 |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.231. LISTENER\_IP\_6

LISTENER\_IP\_6 specifies the IP used by listener. IPv6 addresses are supported.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | String                               |
| Default Value  | "-1"                                 |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.232. LISTENER\_PORT

LISTENER\_PORT specifies the port number to be used by listener. If two or more instances are running on a machine, the port number must be different for each instance. The same rule is in effect for SHM\_KEY parameter.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | Depending on system environment      |
| Class          | Optional, Adjustable, Static, System |
| Range          | 1024 - 65535                         |

### 1.3.233. LISTENER\_PORT\_6

LISTENER\_PORT specifies the port number to be used by listener. If two or more instances are running on a machine, the port number must be different for each instance. The same rule is in effect for SHM\_KEY parameter.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | Depending on system environment      |
| Class          | Optional, Adjustable, Static, System |
| Range          | 1024 - 65535                         |

### 1.3.234. LISTENER\_VIP\_PORT

LISTENER\_VIP\_PORT is a port used by a listener for VIP. If LISTENER\_VIP\_PORT is set differently from LISTENER\_PORT, the listener can accept connections from both ports. When connecting to a database through VIP, a client must use LISTENER\_VIP\_PORT so that sessions are cleaned up during the VIP failback process. When using an active cluster, it is recommended to set to the same value for all database instances.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | <a href="#">LISTENER_PORT</a>        |
| Class          | Optional, Adjustable, Static, System |
| Range          | 1024 - 65535                         |

### 1.3.235. LOAD\_METRIC\_AUTO\_INTERVAL

LOAD\_METRIC\_AUTO\_INTERVAL determines whether to enable auto interval to collect the information about server load balancing.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.236. LOAD\_METRIC\_COLLECT\_INTERVAL\_MAX

LOAD\_METRIC\_COLLECT\_INTERVAL\_MAX specifies the maximum interval of time to collect server load balancing information.

- Parameter Format

| Property       | Description |
|----------------|-------------|
| Parameter Type | Integer     |
| Default Value  | 10          |

| Property | Description                           |
|----------|---------------------------------------|
| Class    | Optional, Adjustable, Dynamic, System |
| Range    | 1 - 60 (s)                            |

### 1.3.237. LOAD\_METRIC\_COLLECT\_INTERVAL\_MIN

LOAD\_METRIC\_COLLECT\_INTERVAL\_MIN specifies the minimum interval of time to collect server load balancing information.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 1                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 1 - 5 (s)                             |

### 1.3.238. LOAD\_METRIC\_HISTORY\_COUNT

LOAD\_METRIC\_HISTORY\_COUNT specifies the number of data to be gathered on server load-balancing.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 10                                   |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.239. LOCAL\_CLUSTER\_ADDR

LOCAL\_CLUSTER\_ADDR specifies the network address to be used for communication between instances in the TAC node.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | String                               |
| Default Value  | ""                                   |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.240. LOCAL\_CLUSTER\_ADDR\_IPV6

LOCAL\_CLUSTER\_ADDR\_IPV6 specifies the network address to be used for communication between instances in the TAC node.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | String                               |
| Default Value  | ""                                   |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.241. LOCAL\_CLUSTER\_IP\_VERSION

Configure an IP protocol family to be used for cluster network communications. To use IPv4, set the configuration to 4. To use IPv6, set the configuration to 6.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 4                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          | 4 or 6                               |

### 1.3.242. LOCAL\_CLUSTER\_PORT

LOCAL\_CLUSTER\_PORT specifies the port number to be used for communication between instances in the TAC node.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 0                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          | 1024 - 65535                         |

### 1.3.243. LOCAL\_CLUSTER\_PORT\_IPV6

LOCAL\_CLUSTER\_PORT\_IPV6 specifies the port number to be used for communication between instances in the TAC node.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 0                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          | 1024 - 65535                         |

### 1.3.244. LOG\_ARCHIVE\_DEST

LOG\_ARCHIVE\_DEST specifies the directory where the archive logs are stored. It must be specified as an absolute path.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ''                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.245. LOG\_ARCHIVE\_DEST\_1

LOG\_ARCHIVE\_DEST\_1 specifies the directory where the archive logs are duplicated.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ""                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.246. LOG\_ARCHIVE\_DEST\_2

LOG\_ARCHIVE\_DEST\_2 specifies the directory where the archive logs are duplicated.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ""                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.247. LOG\_ARCHIVE\_DEST\_3

LOG\_ARCHIVE\_DEST\_3 specifies the directory where the archive logs are duplicated.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ""                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.248. LOG\_ARCHIVE\_DEST\_4

LOG\_ARCHIVE\_DEST\_4 specifies the directory where the archive logs are duplicated.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ''''                                  |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.249. LOG\_ARCHIVE\_DEST\_5

LOG\_ARCHIVE\_DEST\_5 specifies the directory where the archive logs are duplicated.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ''''                                  |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.250. LOG\_ARCHIVE\_DEST\_6

LOG\_ARCHIVE\_DEST\_6 specifies the directory where the archive logs are duplicated.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ''''                                  |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.251. LOG\_ARCHIVE\_DEST\_7

LOG\_ARCHIVE\_DEST\_7 specifies the directory where the archive logs are duplicated.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ''''                                  |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.252. LOG\_ARCHIVE\_DEST\_8

LOG\_ARCHIVE\_DEST\_8 specifies the directory where the archive logs are duplicated.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ''''                                  |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.253. LOG\_ARCHIVE\_DEST\_9

LOG\_ARCHIVE\_DEST\_9 specifies the directory where the archive logs are duplicated.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ''''                                  |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.254. LOG\_ARCHIVE\_FORMAT

LOG\_ARCHIVE\_FORMAT specifies the format of the archive log file. (%d:db id, %r:resetlog tsn, %s:sequence number, %S:zero filled sequence number, %t:thread number, %T:time YYYYMMDD\_HH24MISS)

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | "log-t%t-r%r-s%s.arc"                 |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.255. LOG\_ARCHIVE\_OVERWRITE

LOG\_ARCHIVE\_OVERWRITE determines whether to overwrite the pre-existing archive log file with the current archive log file.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.256. LOG\_ARCHIVE\_RETRY\_TIME

LOG\_ARCHIVE\_RETRY\_TIME specifies wait time (in seconds) to retry when it is failed to make an archive log file.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 3                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.257. LOG\_BUFFER

Size of the memory space where redo logs are stored. @range :

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 100 * 1024 * 1024                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.258. LOG\_BUFFER\_CNT

LOG\_BUFFER\_CNT specifies the number of redo buffers to be created when the multi log buffer feature is enabled by setting ENABLE\_MULTI\_LOG\_BUFFER=Y.

- Parameter Format

| Property       | Description                                                                   |
|----------------|-------------------------------------------------------------------------------|
| Parameter Type | Integer                                                                       |
| Default Value  | <a href="#">ENABLE_MULTI_LOG_BUFFER</a> ? MAX(MIN(get_ncpu() / 8, 32), 4) : 1 |
| Class          | Optional, Adjustable, Static, System                                          |
| Range          |                                                                               |

### 1.3.259. LOG\_CHECKPOINT\_INTERVAL

LOG\_CHECKPOINT\_INTERVAL specifies the maximum number of blocks which have to be written before a checkpoint is forced.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 0                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 0 - INT32_MAX                         |

### 1.3.260. LOG\_CHECKPOINT\_TIMEOUT

LOG\_CHECKPOINT\_TIMEOUT specifies a time interval, in seconds, at which the checkpoint will occur.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 1800                                  |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 0 - INT32_MAX                         |

### 1.3.261. LOG\_DEFAULT\_DEST

LOG\_DEFAULT\_DEST specifies the directory to which server log files are stored. It must be specified as an absolute path.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | String                               |
| Default Value  | ""                                   |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.262. LOG\_FILE\_WITH\_GROUP\_READ\_PERM

If set to Y, newly created redo and archive log files have the group read permission.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.263. LOG\_LVL\_AS

LOG\_LVL\_AS specifies the logging level of AS. The higher the log level, the more log records the listener stores.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 5                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 1 - 6                                 |

### 1.3.264. LOG\_LVL\_CM

LOG\_LVL\_CM specifies the logging level of TBCM. The higher the log level, the more log records CM stores.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 2                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          | 1 - 6                                |

### 1.3.265. LOG\_LVL\_LIS

LOG\_LVL\_LIS specifies the logging level of the listener. The higher the log level, the more log records the listener stores.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 6                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 1 - 6                                 |

### 1.3.266. LOG\_ON\_MEMORY

LOG\_ON\_MEMORY determines whether to write per-thread logs in cacheable memory on file when an error occurs.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.267. LOG\_ON\_MEMORY\_SIZE

LOG\_ON\_MEMORY\_SIZE specifies the size of the buffer which each thread will use to write logs to file when the LOG\_ON\_MEMORY parameter is enabled.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 102400                                |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 100K - 10M                            |

### 1.3.268. LOG\_REPLICATION\_1\_ENABLE

LOG\_REPLICATION\_1\_ENABLE enables the Standby db active.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.269. LOG\_REPLICATION\_2\_ENABLE

LOG\_REPLICATION\_2\_ENABLE enables the Standby db active.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.270. LOG\_REPLICATION\_3\_ENABLE

LOG\_REPLICATION\_3\_ENABLE enables the Standby db active.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.271. LOG\_REPLICATION\_4\_ENABLE

LOG\_REPLICATION\_4\_ENABLE enables the Standby db active.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.272. LOG\_REPLICATION\_5\_ENABLE

LOG\_REPLICATION\_5\_ENABLE enables the Standby db active.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.273. LOG\_REPLICATION\_6\_ENABLE

LOG\_REPLICATION\_6\_ENABLE enables the Standby db active.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.274. LOG\_REPLICATION\_7\_ENABLE

LOG\_REPLICATION\_7\_ENABLE enables the Standby db active.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.275. LOG\_REPLICATION\_8\_ENABLE

LOG\_REPLICATION\_8\_ENABLE enables the Standby db active.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.276. LOG\_REPLICATION\_9\_ENABLE

LOG\_REPLICATION\_9\_ENABLE enables the Standby db active.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.277. LOG\_REPLICATION\_DEST\_1

LOG\_REPLICATION\_DEST\_1 enables writing the Standby db information on the primary database.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ''                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |
| Syntax         | <ip>:<port> {LGWR ARCH} {SYNC ASYNC}  |

### 1.3.278. LOG\_REPLICATION\_DEST\_2

LOG\_REPLICATION\_DEST\_2 enables writing the Standby db information on the primary database.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ''''                                  |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |
| Syntax         | <ip>:<port> {LGWR ARCH} {SYNC ASYNC}  |

### 1.3.279. LOG\_REPLICATION\_DEST\_3

LOG\_REPLICATION\_DEST\_3 enables writing the Standby db information on the primary database.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ''''                                  |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |
| Syntax         | <ip>:<port> {LGWR ARCH} {SYNC ASYNC}  |

### 1.3.280. LOG\_REPLICATION\_DEST\_4

LOG\_REPLICATION\_DEST\_4 enables writing the Standby db information on the primary database.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ''''                                  |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |
| Syntax         | <ip>:<port> {LGWR ARCH} {SYNC ASYNC}  |

### 1.3.281. LOG\_REPLICATION\_DEST\_5

LOG\_REPLICATION\_DEST\_5 enables writing the Standby db information on the primary database.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ''''                                  |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |
| Syntax         | <ip>:<port> {LGWR ARCH} {SYNC ASYNC}  |

### 1.3.282. LOG\_REPLICATION\_DEST\_6

LOG\_REPLICATION\_DEST\_6 enables writing the Standby db information on the primary database.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ''''                                  |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |
| Syntax         | <ip>:<port> {LGWR ARCH} {SYNC ASYNC}  |

### 1.3.283. LOG\_REPLICATION\_DEST\_7

LOG\_REPLICATION\_DEST\_7 enables writing the Standby db information on the primary database.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ''''                                  |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |
| Syntax         | <ip>:<port> {LGWR ARCH} {SYNC ASYNC}  |

## 1.3.284. LOG\_REPLICATION\_DEST\_8

LOG\_REPLICATION\_DEST\_8 enables writing the Standby db information on the primary database.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ""                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |
| Syntax         | <ip>:<port> {LGWR ARCH} {SYNC ASYNC}  |

## 1.3.285. LOG\_REPLICATION\_DEST\_9

LOG\_REPLICATION\_DEST\_9 enables writing the Standby db information on the primary database.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ""                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |
| Syntax         | <ip>:<port> {LGWR ARCH} {SYNC ASYNC}  |

## 1.3.286. LOG\_REPLICATION\_MODE

LOG\_REPLICATION\_MODE allows user to manage the operation mode of the Primary DB.

- Parameter Format

| Property       | Description                                       |
|----------------|---------------------------------------------------|
| Parameter Type | String                                            |
| Default Value  | "UNPROTECTED"                                     |
| Class          | Optional, Adjustable, Dynamic, System             |
| Range          |                                                   |
| Syntax         | {UNPROTECTED PROTECTION AVAILABILITY PERFORMANCE} |

### 1.3.287. LONGOPS\_KEEP\_INTERVAL

LONGOPS\_KEEP\_INTERVAL specifies the retention time (in minutes) for history of long-running operations.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 10080                                 |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 10(10 min) - 40320(4 weeks)           |

### 1.3.288. LONGOPS\_THRESHOLD\_SEC

LONGOPS\_THRESHOLD\_SEC specifies timeout (in seconds) of performing an operation. If it takes longer than this time to handle an operation, the operation is regarded as a long-running operation and registered in v\$session\_longops.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Integer                                |
| Default Value  | 0                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          | 0(sec) ~ 10000(sec)                    |

### 1.3.289. LONG\_OPERATION\_KEEP\_COUNT

LONG\_OPERATION\_KEEP\_COUNT specifies the number of high-load SQL statements on which to gather statistics.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 10                                   |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

## 1.3.290. LSNR\_DENIED\_IP

LSNR\_DENIED\_IP specifies the list of IPs that cannot connect to the listener. This parameter can be set with the same method for setting LSNR\_INVITED\_IP. If LSNR\_INVITED\_IP is set, setting in LSNR\_DENIED\_IP will be ignored.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ''                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

## 1.3.291. LSNR\_DENIED\_IP\_FILE

LSNR\_DENIED\_IP\_FILE specifies the file containing the list of IPs that cannot connect to the listener. If the LSNR\_INVITED\_IP or the LSNR\_INVITED\_IP\_FILE is set, setting in LSNR\_DENIED\_IP\_FILE will be ignored.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ''                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

## 1.3.292. LSNR\_INVITED\_IP

LSNR\_INVITED\_IP specifies the list of IPs that can connect to the listener. Use a semicolon (;) to distinguish each IP. The IP format is address/number of bits as follows. (Ex) LSNR\_INVITED\_IP="192.168.20.0/24;192.168.10.101" 192.168.20.0/24 : Accept all IP connections starting with 192.168.20.xxx. 192.168.10.101 : If the number of bits is omitted, accept only the corresponding IP.

- Parameter Format

| Property       | Description |
|----------------|-------------|
| Parameter Type | String      |

| Property      | Description                           |
|---------------|---------------------------------------|
| Default Value | ""                                    |
| Class         | Optional, Adjustable, Dynamic, System |
| Range         |                                       |

### 1.3.293. LSNR\_INVITED\_IP\_FILE

LSNR\_INVITED\_IP\_FILE specifies the file containing the list of IPs that can connect to the listener. A single IP address is written in each line of the file. The IP format is address/number of bits like LSNR\_INVITED\_IP. @range :

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ""                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.294. LSNR\_LISTEN\_BACKLOG

LSNR\_LISTEN\_BACKLOG specifies the maximum length of the queue of pending connections.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 250                                  |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.295. LSNR\_LOG\_DEST

Specifies the directory where the listener log files are stored. Must be specified as an absolute path.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | String                               |
| Default Value  | Depending on system environment      |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.296. LSNR\_LOG\_FILE\_SIZE

LSNR\_LOG\_FILE\_SIZE specifies the maximum size of each listener log file. If a single log file grows more than the size specified in this parameter, the current log file will be closed and a new file will be created.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 10485760                             |
| Class          | Optional, Adjustable, Static, System |
| Range          | 100K - 1G                            |

### 1.3.297. LSNR\_LOG\_TOTAL\_SIZE\_LIMIT

LSNR\_LOG\_TOTAL\_SIZE\_LIMIT specifies the maximum total file size that can be stored in the directory set in LSNR\_LOG\_DEST. Use this parameter to prevent database log files from growing endlessly by deleting the oldest files when the size of files exceeds the specified value and hence reusing the disk space.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 300 * 1024 * 1024                    |
| Class          | Optional, Adjustable, Static, System |
| Range          | 100K - INT64_MAX                     |

### 1.3.298. LSNR\_OUTFILE\_BACKUP\_LIMIT\_CNT

Specifies the maximum number of backups that can be stored in LSNR\_OUTFILE\_DEST. For details, refer to the description of OUTFILE\_PATH\_BACKUP\_LIMIT\_CNT.

- Parameter Format

| Property       | Description                                   |
|----------------|-----------------------------------------------|
| Parameter Type | Integer                                       |
| Default Value  | <a href="#">OUTFILE_PATH_BACKUP_LIMIT_CNT</a> |
| Class          | Optional, Adjustable, Static, System          |
| Range          | 2 ~ 500                                       |

### 1.3.299. LSNR\_OUTFILE\_DEST

Specifies the directory where the listener outfiles are stored. Must be specified as an absolute path.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | String                               |
| Default Value  | Depending on system environment      |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.300. LSNR\_SCHEDULE\_SAME\_PROC\_FIRST

Scheduling method for passing a new session to a working process when a listener gets the session. There are two methods. If set to Y, sessions are sent to a working process until all wthr of the process are in busy state. If set to N, sessions are evenly distributed to multiple working processes.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | Y                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.301. MAX\_BG\_SESSION\_COUNT

Number of sessions for processing internal tasks and jobs among MAX\_SESSION\_COUNT that are not sessions requested by LSNR. As many as MAX\_SESSION\_COUNT - MAX\_BG\_SESSION\_COUNT sessions can be started from LSNR. MAX\_BG\_SESSION\_COUNT must be smaller than MAX\_SESSION\_COUNT and must be divisible by WTHR\_PER\_PROC.

- Parameter Format

| Property       | Description                                |
|----------------|--------------------------------------------|
| Parameter Type | Integer                                    |
| Default Value  | (MAX_SESSION_COUNT - MAX_FG_SESSION_COUNT) |
| Class          | Optional, Adjustable, Static, System       |
| Range          | 0 ~ INT32_MAX                              |

### 1.3.302. MAX\_FG\_SESSION\_COUNT

Number of sessions for user connections requested by LSNR among MAX\_SESSION\_COUNT. As many as MAX\_SESSION\_COUNT - MAX\_FG\_SESSION\_COUNT sessions can be started by internal tasks. MAX\_FG\_SESSION\_COUNT must be less than or equal to MAX\_SESSION\_COUNT and must be divisible by WTHR\_PER\_PROC.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | MAX_SESSION_COUNT                    |
| Class          | Optional, Adjustable, Static, System |
| Range          | 0 ~ INT32_MAX                        |

### 1.3.303. MAX\_SESSION\_COUNT

Specifies the maximum number of sessions provided by a server. If necessary, the number of worker processes configuring a session and the number of worker threads per process can be manually specified using WTHR\_PROC\_CNT and WTHR\_PER\_PROC respectively. WTHR\_PROC\_CNT times WTHR\_PER\_PROC must be MAX\_SESSION\_COUNT. If a user does not manually specify WTHR\_PROC\_CNT and WTHR\_PER\_PROC, the default values of the number of worker threads per process (WTHR\_PER\_PROC) is 10, and the number of worker processes (WTHR\_PROC\_CNT) is

MAX\_SESSION\_COUNT/WTNR\_PER\_PROC (round up if there is a remainder). In Windows NT. Windows Server 2003 and below, MAX\_SESSION\_COUNT should be set to an appropriate value of 1500 or less.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 20                                    |
| Class          | Mandatory, Adjustable, Static, System |
| Range          | 1 ~ INT32_MAX                         |

### 1.3.304. MEMORY\_MAX\_TARGET

Specifies the potential total amount of usable memory (shared memory plus private memory) of an instance for the foreseeable future, in bytes.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | <a href="#">MEMORY_TARGET</a>        |
| Class          | Optional, Adjustable, Static, System |
| Range          | 1M -                                 |

### 1.3.305. MEMORY\_MODEL

MEMORY\_MODEL specifies the memory model predefined by size.

- Parameter Format

| Property       | Description                                                                                                                 |
|----------------|-----------------------------------------------------------------------------------------------------------------------------|
| Parameter Type | String                                                                                                                      |
| Default Value  | <a href="#">TOTAL_SHM_SIZE</a> <= 2147483648 ? "SMALL" : <a href="#">TOTAL_SHM_SIZE</a> <= 34359738368 ? "MEDIUM" : "LARGE" |
| Class          | Optional, Adjustable, Static, System                                                                                        |
| Range          | SMALL   MEDIUM   LARGE                                                                                                      |

## 1.3.306. MEMORY\_TABLESPACE\_SIZE

MEMORY\_TABLESPACE\_SIZE specifies the maximum size of the memory tablespace.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 0                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          | 0                                    |

## 1.3.307. MEMORY\_TARGET

Specifies the current total amount of usable memory (shared memory plus private memory) of an instance, in bytes.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Integer                                |
| Default Value  | 0                                      |
| Class          | Mandatory, Adjustable, Dynamic, System |
| Range          | 1M -                                   |

## 1.3.308. MEM\_LOG\_BACKUP\_LIMIT\_CNT

Specifies the maximum number of backups that can be stored in MEM\_LOG\_DEST. For details, refer to the description of TRACE\_BACKUP\_LIMIT\_CNT. Consequently, the maximum path size may approximate MEM\_LOG\_TOTAL\_SIZE\_LIMIT \* (MEM\_LOG\_BACKUP\_LIMIT\_CNT + 1).

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Integer                                |
| Default Value  | <a href="#">TRACE_BACKUP_LIMIT_CNT</a> |
| Class          | Optional, Adjustable, Dynamic, System  |
| Range          | 2 ~ 500                                |

### 1.3.309. MEM\_LOG\_DEST

MEM\_LOG\_DEST specifies the directory in which the memory log file is saved. The absolute path should be specified.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | Depending on system environment       |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.310. MEM\_LOG\_TOTAL\_SIZE\_LIMIT

Specifies the maximum total file size that can be stored in the directory specified in MEM\_LOG\_DEST, excluding backup directories. For details, refer to the description of TRACE\_TOTAL\_SIZE\_LIMIT.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Integer                                |
| Default Value  | <a href="#">TRACE_TOTAL_SIZE_LIMIT</a> |
| Class          | Optional, Adjustable, Dynamic, System  |
| Range          | 100K - INT64_MAX                       |

### 1.3.311. NAMED\_PIPE\_DIR

NAMED\_PIPE\_DIR specifies the directory where named pipes used between processes are created while starting and closing the system.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | String                               |
| Default Value  | Depending on system environment      |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.312. NBU\_ARCHIVELOG\_SEARCH

Whether to search archive logs in NetBackup to show in v\$archive\_dest\_files or restore them for Media Recovery.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.313. NBU\_ARCHIVE\_POLICY\_NAME

NetBackup policy name for archivelogs (matches to NBBSA\_POLICY of NetBackup XBSA)

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | "TIBERO_ARCHIVE_POLICY"               |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.314. NBU\_ARCHIVE\_RETRY\_CNT

Retry counts when fails to create an archivelog at NetBackup.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 2                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.315. NBU\_BACKUP\_INST\_SID

SID of instance performing rmgr backup operations

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ""                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.316. NBU\_BACKUP\_POLICY\_NAME

NetBackup policy name for backupsets (matches to NBBSA\_POLICY of NetBackup XBSA)

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | "TIBERO_BACKUP_POLICY"                |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.317. NBU\_BSA\_API\_VERSION

Backup service API version for NetBackup connection. (matches to BSA\_API\_VERSION of NetBackup XBSA)

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | "1.1.0"                               |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.318. NBU\_BSA\_LOG\_DIRECTORY

Optional. Identifies the single name of the directory that contains the log files of the XBSA application.  
(matches to NBBSA\_LOG\_DIRECTORY of NetBackup XBSA)

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ""                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.319. NBU\_BSA\_LOG\_LEVEL

verbose level of the database logs (0 - 9) (default: 5) (matches to NBBSA\_VERBOSE\_LEVEL of NetBackup XBSA)

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 5                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.320. NBU\_CLIENT\_COUNT

the number of NetBackup clients

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 0                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.321. NBU\_CLIENT\_HOSTNAME\_0

hostname of NetBackup client 0

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ""                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.322. NBU\_CLIENT\_HOSTNAME\_1

hostname of NetBackup client 1

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ""                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.323. NBU\_CLIENT\_HOSTNAME\_2

hostname of NetBackup client 2

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ""                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.324. NBU\_CLIENT\_HOSTNAME\_3

hostname of NetBackup client 3

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ""                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.325. NBU\_CLIENT\_HOSTNAME\_4

hostname of NetBackup client 4

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ""                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.326. NBU\_CLIENT\_HOSTNAME\_5

hostname of NetBackup client 5

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ""                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.327. NBU\_CLIENT\_HOSTNAME\_6

hostname of NetBackup client 6

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ""                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.328. NBU\_CLIENT\_HOSTNAME\_7

hostname of NetBackup client 7

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ""                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.329. NBU\_CLIENT\_HOSTNAME\_8

hostname of NetBackup client 8

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ""                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.330. NBU\_CLIENT\_HOSTNAME\_9

hostname of NetBackup client 9

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ""                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.331. NBU\_CLIENT\_HOST\_NAME

Hostname of the system where NetBackup client locates. (matches to NBBSA\_CLIENT\_HOST of Net Backup XBSA)

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ""                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.332. NBU\_MASTER\_HOST\_NAME

Hostname of the system where NetBackup master locates. (matches to BSA\_SERVICE\_HOST of Net Backup XBSA)

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ""                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.333. NBU\_OBJ\_OWNER\_GROUP\_NAME

Owner group name which can access to NetBackup. (matches to NBBSA\_OBJECT\_GROUP of NetBackup XBSA)

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ""                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.334. NBU\_OBJ\_OWNER\_NAME

Owner name who can access to NetBackup. (matches to NBBSA\_OBJECT\_OWNER of NetBackup XBSA)

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ""                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.335. NBU\_PROC\_WTHR\_CNT

Number of NetBackup working processes. Its name should be NBU\_PROC\_CNT. RMGR parallel back up/recover uses 1 lock\_slave + rv slaves for DF, not --archive-only.

- Parameter Format

| Property       | Description                               |
|----------------|-------------------------------------------|
| Parameter Type | Integer                                   |
| Default Value  | (RECO_PROC_WTHR_CNT- _CACHE_RECO_DOP-3)/2 |
| Class          | Optional, Adjustable, Static, System      |
| Range          | 1 - 4096                                  |

### 1.3.336. NBU\_SCHEDULE\_NAME

NetBackup schedule name (matches to NBBSA\_SCHEDULE of NetBackup XBSA)

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ""                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.337. NBU\_SKIP\_HOST\_CHECK

skip the consistency check between hostname and SID

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.338. NBU\_TWO\_PHASE\_CF\_BACKUP

Whether to create a backup as a local file before sending the cf backup to Netbackup. @range :

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | Y                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.339. NETWORK\_SECURITY\_AGENT\_MANAGER\_PATH

NETWORK\_SECURITY\_AGENT\_MANAGER\_PATH specifies the location of the network security agent manager file. It must be specified as an absolute path.

- Parameter Format

| Property       | Description                                                    |
|----------------|----------------------------------------------------------------|
| Parameter Type | String                                                         |
| Default Value  | ""                                                             |
| Class          | Optional, Adjustable, Static, System                           |
| Range          |                                                                |
| Syntax         | NETWORK_SECURITY_AGENT_MANAGER_PATH=filename[:filename [...] ] |

### 1.3.340. NLS\_CALENDAR

NLS\_CALENDAR specifies which calendar system Tiberio uses.

- Parameter Format

| Property       | Description                                 |
|----------------|---------------------------------------------|
| Parameter Type | String                                      |
| Default Value  | "GREGORIAN"                                 |
| Class          | Optional, Adjustable, Dynamic, Session      |
| Range          | GREGORIAN   JAPANESE IMPERIAL   THAI BUDDHA |

### 1.3.341. NLS\_COMP

NLS\_COMP specifies the comparison behavior of SQL operations. The characters are compared according to their binary sequence or linguistic sequence defined in NLS\_SORT.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | String                                 |
| Default Value  | "BINARY"                               |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          | BINARY   LINGUISTIC                    |

## 1.3.342. NLS\_CURRENCY

NLS\_CURRENCY specifies the local currency symbol string for the 'L' number format.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | String                                 |
| Default Value  | "\$"                                   |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          | Maximum 10 characters in length        |

## 1.3.343. NLS\_DATE\_FORMAT

NLS\_DATE\_FORMAT specifies the default date format for date display.

- Parameter Format

| Property       | Description                                              |
|----------------|----------------------------------------------------------|
| Parameter Type | String                                                   |
| Default Value  | "YYYY/MM/DD"                                             |
| Class          | Optional, Adjustable, Dynamic, Session                   |
| Range          | Any valid date format mask up to 64 characters in length |

## 1.3.344. NLS\_DATE\_LANGUAGE

NLS\_DATE\_LANGUAGE specifies the language to use the spelling of date format elements.

- Parameter Format

| Property       | Description                                                                                                                                                                                                                                                                                                          |
|----------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Parameter Type | String                                                                                                                                                                                                                                                                                                               |
| Default Value  | "KOREAN"                                                                                                                                                                                                                                                                                                             |
| Class          | Optional, Adjustable, Dynamic, Session                                                                                                                                                                                                                                                                               |
| Range          | KOREAN   AMERICAN   JAPANESE   VIETNAMESE   BRAZILIAN PORTUGUESE   TRADITIONAL CHINESE   RUSSIAN   SIMPLIFIED CHINESE   THAI   GERMAN   SPANISH   ITALIAN   BULGARIAN   CATALAN   DANISH   DUTCH   FINNISH   POLISH   FRENCH   NORWEGIAN   LITHUANIAN   CROATIAN   CZECH   ESTONIAN   HEBREW   HUNGARIAN   ICELANDIC |

| Property | Description                                                                              |
|----------|------------------------------------------------------------------------------------------|
|          | LATVIAN   PORTUGUESE   ROMANIAN   SLOVAK   SWEDISH   TURKISH   ENGLISH   MEXICAN SPANISH |

### 1.3.345. NLS\_DUAL\_CURRENCY

NLS\_DUAL\_CURRENCY specifies the dual currency symbol string for the 'U' number format.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | String                                 |
| Default Value  | "\$"                                   |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          | Maximum 10 characters in length        |

### 1.3.346. NLS\_ISO\_CURRENCY

NLS\_ISO\_CURRENCY determines the international currency symbol string for the 'C' number format.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | String                                 |
| Default Value  | "KOREA"                                |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          | Any valid NLS_TERRITORY value          |

### 1.3.347. NLS\_LANGUAGE

NLS\_LANGUAGE specifies the current language for the session.

- Parameter Format

| Property       | Description |
|----------------|-------------|
| Parameter Type | String      |
| Default Value  | "KOREAN"    |

| Property | Description                                                                                                                                                                                                                                                                                                                                                                                |
|----------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Class    | Optional, Adjustable, Dynamic, Session                                                                                                                                                                                                                                                                                                                                                     |
| Range    | AMERICAN   KOREAN   JAPANESE   SIMPLIFIED CHINESE   BRAZILIAN PORTUGUESE   BULGARIAN   CATALAN   CZECH   DANISH   DUTCH   ENGLISH   ESTONIAN   FINNISH   FRENCH   GERMAN   HEBREW   HUNGARIAN   ICELANDIC   ITALIAN   LITHUANIAN   NORWEGIAN   POLISH   PORTUGUESE   ROMANIAN   RUSSIAN   SLOVAK   SPANISH   SWEDISH   THAI   TRADITIONAL CHINESE   TURKISH   VIETNAMESE   MEXICAN SPANISH |

### 1.3.348. NLS\_LANG\_AT\_BOOT

NLS\_LANG\_AT\_BOOT specifies the default character set before the database is mounted.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | String                               |
| Default Value  | "UTF8"                               |
| Class          | Optional, Adjustable, Static, System |
| Range          | Available Character Set              |

### 1.3.349. NLS\_LENGTH\_SEMANTICS

NLS\_LENGTH\_SEMANTICS specifies the default character length semantics.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | String                                 |
| Default Value  | "BYTE"                                 |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          | CHAR   BYTE                            |

### 1.3.350. NLS\_NUMERIC\_CHARACTERS

NLS\_NUMERIC\_CHARACTERS specifies the characters for the thousand group separator and decimal character for the session.

- Parameter Format

| Property       | Description                                                                          |
|----------------|--------------------------------------------------------------------------------------|
| Parameter Type | String                                                                               |
| Default Value  | ".,."                                                                                |
| Class          | Optional, Adjustable, Dynamic, Session                                               |
| Range          | Two different single-byte characters, excluding the following characters: +, -, <, > |

### 1.3.351. NLS\_SORT

NLS\_SORT specifies the collating sequence used by ORDER BY queries.

- Parameter Format

| Property       | Description                                                                           |
|----------------|---------------------------------------------------------------------------------------|
| Parameter Type | String                                                                                |
| Default Value  | "BINARY"                                                                              |
| Class          | Optional, Adjustable, Dynamic, Session                                                |
| Range          | BINARY   GERMAN   FRENCH   DUTCH   THAI_DICTIONARY   SCHINESE_PINYIN   UNICODE_BINARY |

### 1.3.352. NLS\_TERRITORY

NLS\_TERRITORY specifies the name of the territory for the session.

- Parameter Format

| Property       | Description                                                                                                                                                                                                                                                                                                                                             |
|----------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Parameter Type | String                                                                                                                                                                                                                                                                                                                                                  |
| Default Value  | "KOREA"                                                                                                                                                                                                                                                                                                                                                 |
| Class          | Optional, Adjustable, Dynamic, Session                                                                                                                                                                                                                                                                                                                  |
| Range          | AMERICA   KOREA   JAPAN   CHINA   AUSTRALIA   AUSTRIA   BELGIUM   BRAZIL   CANADA   CZECH REPUBLIC   EGYPT   FILAND   FRANCE   GERMANY   GREECE   HONG KONG   HUNGARY   INDONESIA   IRAQ   ITALY   MALAYSIA   MEXICO   POLAND   RUSSIA   SINGAPORE   SPAIN   SWEDEN   TAIWAN   THAILAND   THE NETHERLANDS   TURKEY   UKRAINE   UNITED KINGDOM   VIETNAM |

### 1.3.353. NLS\_TIMESTAMP\_FORMAT

NLS\_TIMESTAMP\_FORMAT specifies the default timestamp format to be used when the database is started.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | String                                 |
| Default Value  | "YYYY/MM/DD HH24:MI:SSXFF"             |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          | Maximum 64 characters in length        |

### 1.3.354. NLS\_TIMESTAMP\_TZ\_FORMAT

NLS\_TIMESTAMP\_TZ\_FORMAT specifies the default timestamp time zone format to be used when the database is started.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | String                                 |
| Default Value  | "YYYY/MM/DD HH24:MI:SSXFF TZR"         |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          | Maximum 64 characters in length        |

### 1.3.355. NLS\_TIME\_FORMAT

NLS\_TIME\_FORMAT specifies the default time format to be used when the database is started.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | String                                 |
| Default Value  | "HH24:MI:SSXFF"                        |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          | Maximum 64 characters in length        |

### 1.3.356. NLS\_TIME\_TZ\_FORMAT

NLS\_TIME\_TZ\_FORMAT specifies the default time time zone format to be used when the database is started.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | String                                 |
| Default Value  | "HH24:MI:SSXFF TZR"                    |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          | Maximum 64 characters in length        |

### 1.3.357. NTF\_LISTEN\_PORT

Listening port for NTF sender. Server event subscriber connects to this port.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | <a href="#">LISTENER_PORT</a> + 6    |
| Class          | Optional, Adjustable, Static, System |
| Range          | 1024 ~ 65535                         |

### 1.3.358. OMIT\_PSM\_SQL\_LOG\_ON\_MEMORY

OMIT\_PSM\_SQL\_LOG\_ON\_MEMORY determines whether to write per-thread SQL of PSM unit logs in cacheable memory on file when an error occurred.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

## 1.3.359. OPEN\_CURSORS

OPEN\_CURSORS specifies the maximum number of cursors that a single session can open at once.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 50                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 10 - 1000                             |
| Syntax         | OPEN_CURSORS=50                       |

## 1.3.360. OPTIMIZER\_CAPTURE\_SQL\_PLAN\_BASELINES

Enables or disables to capture and save repeated SQL statements for SQL plan baselines.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Boolean                                |
| Default Value  | N                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          |                                        |

## 1.3.361. OPTIMIZER\_CAPTURE\_SQL\_PLAN\_BASELINES\_DURATION

Time interval within which an already existing SQL statement is not captured.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Integer                                |
| Default Value  | 0                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          | 0 -                                    |

### 1.3.362. OPTIMIZER\_LOG\_OUTLINE

Specifies whether or not to log the plan outline.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Boolean                                |
| Default Value  | Y                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          |                                        |

### 1.3.363. OPTIMIZER\_MODE

OPTIMIZER\_MODE specifies the goal of the SQL query optimizer.

- Parameter Format

| Property       | Description                                     |
|----------------|-------------------------------------------------|
| Parameter Type | String                                          |
| Default Value  | "ALL_ROWS"                                      |
| Class          | Optional, Adjustable, Dynamic, Session          |
| Range          | 10 - 1000                                       |
| Syntax         | { FIRST_ROWS_[1   10   100   1000]   ALL_ROWS } |

### 1.3.364. OPTIMIZER\_TRACE

optimizer trace dump

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Boolean                                |
| Default Value  | N                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          |                                        |

### 1.3.365. OPTIMIZER\_USE\_INVISIBLE\_INDEXES

OPTIMIZER\_USE\_INVISIBLE\_INDEXES specifies the availability of invisible indexes

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Boolean                                |
| Default Value  | N                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          |                                        |

### 1.3.366. OPTIMIZER\_USE\_PENDING\_STATISTICS

OPTIMIZER\_USE\_PENDING\_STATISTICS specifies whether to use stats in pending

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Boolean                                |
| Default Value  | N                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          |                                        |

### 1.3.367. OPTIMIZER\_USE\_SQL\_PLAN\_BASELINES

Enables or disables to use the created SQL plan baselines.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Boolean                                |
| Default Value  | N                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          |                                        |

### 1.3.368. OPTIMIZER\_USE\_SQL\_PROFILES

Specifies whether or not to use the profile.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Boolean                                |
| Default Value  | Y                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          |                                        |

### 1.3.369. OS\_AUTH\_PREFIX

Prefix for OS authenticated user.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | String                               |
| Default Value  | "OSA\$"                              |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.370. OUTFILE\_PATH\_BACKUP\_LIMIT\_CNT

Specifies the maximum number of backups that can be stored in \*\_OUTFILE\_DEST. This parameter prevents outfile backup directories from unlimited growth. When the backup counts exceed the specified value, the oldest backup files are removed to allow disk space reuse. Backup operations are performed when database reboot occurs. Used as the default value for LSNR\_OUTFILE\_BACKUP\_LIMIT\_CNT and SVR\_OUTFILE\_BACKUP\_LIMIT\_CNT.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 10                                   |
| Class          | Optional, Adjustable, Static, System |
| Range          | 2 ~ 500                              |

## 1.3.371. PARALLEL\_DEFAULT\_DOP

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Integer                                |
| Default Value  | 4                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          |                                        |

## 1.3.372. PARALLEL\_GRANULE\_SIZE

PARALLEL\_GRANULE\_SIZE specifies the number of segment blocks (granule size) read into memory in a single SQL query performing a full table scan.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 3                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

## 1.3.373. PARALLEL\_THREADS\_PER\_CPU

PARALLEL\_THREADS\_PER\_CPU specifies the number of parallel execution threads per CPU.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 2                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.374. PEP\_PROC\_CNT

PEP\_PROC\_CNT specifies the number of pe processes.

- Parameter Format

| Property       | Description                                     |
|----------------|-------------------------------------------------|
| Parameter Type | Integer                                         |
| Default Value  | MAX(5, <a href="#">MAX_SESSION_COUNT</a> / 200) |
| Class          | Optional, Adjustable, Static, System            |
| Range          | 0 - 10000                                       |

### 1.3.375. PP\_SAVE\_SYS\_BINDVAR

Specifies whether or not to save system bind variables in PP cache when cursor\_sharing is set to 'force'.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Boolean                                |
| Default Value  | Y                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          |                                        |

### 1.3.376. PRIVKEY\_FILE

PRIVKEY\_FILE specifies the location of the private key files. It must be specified as an absolute path.

- Parameter Format

| Property       | Description                             |
|----------------|-----------------------------------------|
| Parameter Type | String                                  |
| Default Value  | ""                                      |
| Class          | Optional, Adjustable, Static, System    |
| Range          |                                         |
| Syntax         | PRIVKEY_FILE=filename[:filename [...] ] |

### 1.3.377. PSM\_CODE\_TYPE

PSM\_CODE\_TYPE specifies the interpreter mode in which PSM is implemented. It is efficient to use in interpreter mode for anonymous blocks.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | String                                 |
| Default Value  | "INTERPRETED"                          |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          | INTERPRETED                            |

### 1.3.378. PSM\_DEBUG

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Boolean                                |
| Default Value  | N                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          |                                        |

### 1.3.379. PSM\_ERROR\_CONV

Option to automatically convert the err\_no in pragma exception\_init(exception, err\_no).

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.380. PSM\_LOG\_ON\_THROW

Option to log object name and line number of throws in PSM.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Boolean                                |
| Default Value  | N                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          |                                        |

### 1.3.381. PSM\_SHLIB\_DIR

When a PSM subprogram or anonymous block is implemented in compile mode, PSM\_SHLIB\_DIR creates a shared library by converting PSM code into C code. The shared library location is stored as an absolute path. If not specified, the shared library will be stored in the DB\_CREATE\_FILE\_DEST/psm directory. (deprecated)

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ''''                                  |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.382. PSM\_SQL\_RESULT\_CACHE

PSM\_SQL\_RESULT\_CACHE determines whether to enable the PSM's feature for caching the SQL results.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Boolean                                |
| Default Value  | Y                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          |                                        |

### 1.3.383. QUERY\_REWRITE\_ENABLED

QUERY\_REWRITE\_ENABLED enables or disables query rewrite which is globally applied for the database.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | String                                 |
| Default Value  | "TRUE"                                 |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          | FALSE   TRUE   FORCE                   |

### 1.3.384. QUERY\_REWRITE\_INTEGRITY

Sets the level of query rewrite integrity.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | String                                 |
| Default Value  | "ENFORCED"                             |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          | STALE_TOLERATED   ENFORCED             |

### 1.3.385. RECO\_PROC\_WTHR\_CNT

it specifies the number of recovery working processes.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 35 + _CACHE_RECO_DOP                 |
| Class          | Optional, Adjustable, Static, System |
| Range          | 1 - INT32_MAX                        |

### 1.3.386. RESULT\_CACHE\_MAX\_RATIO

The maximum size ratio of a query that can be cached. For example, if RESULT\_CACHE\_MAX\_SIZE is 10MB and this value is set to 5, the maximum query size that can be cached will be 500KB, which is 5% of 10MB.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 5                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 1 - 100                               |

### 1.3.387. RESULT\_CACHE\_MAX\_SIZE

The maximum memory size that can be used by the result cache in shared memory. If this value is set to 0, the result cache function will be disabled.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | <a href="#">TOTAL_SHM_SIZE</a> / 100  |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 0 -                                   |

### 1.3.388. RESULT\_CACHE\_MODE

RESULT\_CACHE\_MODE determines which query results to store in cache. If this value is set to MANUAL, the query result will be stored only when the "RESULT\_CACHE" hint is used. If this value set to FORCE, all query results will be stored in cache.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | String                                 |
| Default Value  | "MANUAL"                               |
| Class          | Optional, Adjustable, Dynamic, Session |

| Property | Description    |
|----------|----------------|
| Range    | MANUAL   FORCE |

### 1.3.389. RESUMABLE\_SPACE\_ALLOC

RESUMABLE\_SPACE\_ALLOC determines whether to enable the resumable space allocation feature.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Boolean                                |
| Default Value  | N                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          |                                        |

### 1.3.390. RESUMABLE\_SPACE\_ALLOC\_TIMEOUT

RESUMABLE\_SPACE\_ALLOC\_TIMEOUT specifies a time interval in seconds at which the resumable space allocation will occur.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Integer                                |
| Default Value  | 0                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          | 0 - INT32_MAX seconds                  |

### 1.3.391. RMGR\_BACKUP\_DEST

RMGR\_BACKUP\_DEST specifies the directory where RMGR backup files are stored. It must be specified as an absolute path outside active storage.

- Parameter Format

| Property       | Description |
|----------------|-------------|
| Parameter Type | String      |

| Property      | Description                           |
|---------------|---------------------------------------|
| Default Value | ""                                    |
| Class         | Optional, Adjustable, Dynamic, System |
| Range         |                                       |

### 1.3.392. RMGR\_CATALOG\_SID

DB SID of the catalog database.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ""                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.393. RMGR\_CHECK\_FREE\_DISK\_SIZE

Whether or not to check the free space on the disk before performing a tbrmgr backup. This feature is not recommended if the backup path is a space configured with VERITAS Volume Manager.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | Y                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.394. RMGR\_NBU\_BACKUP\_DEST

RMGR\_NBU\_BACKUP\_DEST specifies the directory where RMGR backup files are stored when USE\_NBU\_FOR\_BACKUPSET=Y.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ''''                                  |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.395. RMGR\_PRIMARY\_SID

DB SID of the primary database. When tbrmgr connected to the standby database and RMGR\_PRIMARY\_SID is defined, tbrmgr also connects to the primary database.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ''''                                  |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.396. RMGR\_RESTORE\_BACKUP\_CF

Always restores CF, although CF exists.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.397. RMGR\_TAR\_PATH

tar path for RMGR

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ""                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.398. RSBTBL\_BUCKET\_CNT\_PER\_ROOT\_LIST\_FOR\_RECONFIGURATION

RSBTBL\_BUCKET\_CNT\_PER\_ROOT\_LIST\_FOR\_RECONFIGURATION specifies the number of rsb table buckets in a root list.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 100                                   |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.399. RSRC\_CHECK\_INTERVAL

RSRC\_CHECK\_INTERVAL specifies the resource manager thread check interval in seconds.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 3                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.400. RSRC\_CONTROL\_THRESHOLD

RSRC\_CONTROL\_THRESHOLD specifies the resource manager control threshold in percentage.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 90                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 0 ~ 100                               |

### 1.3.401. RT\_AWAKE\_WAITER\_INTERVAL

RT\_AWAKE\_WAITER\_INTERVAL specifies the time interval (in 0.1 seconds) that the flow control resource tuner awakes waiting WTHR.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 10                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 10 - 600 (1 - 60 seconds)             |

### 1.3.402. RT\_UPDATE\_ALLOWABLE\_INFLOW\_CNT\_INTERVAL

RT\_UPDATE\_ALLOWABLE\_INFLOW\_CNT\_INTERVAL specifies the time interval (in 0.1 seconds) that the flow control resource tuner updates a desired control value.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 10                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 10 - 600 (1 - 60 seconds)             |

### 1.3.403. SANITY\_CHECK\_DURING\_LOG\_SWITCH

SANITY\_CHECK\_DURING\_LOG\_SWITCH determines whether to check lost or damaged log files during LOG SWITCH.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.404. SECU\_CHANGE\_PASSWORD\_WITH\_CASE\_SENSITIVITY

Determines whether to change password with case-sensitivity when using client connection. Deprecated.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | Y                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.405. SEGMENT\_EXTEND\_LAST

SEGMENT\_EXTEND\_LAST enables the submission of the extension request on segment of a size smaller than the one of the previous request which failed due to shortages of segment space.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | Y                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.406. SEM\_KEY

SEM\_KEY specifies the key of the semaphore to be used by each instance. In general, if this parameter is not defined, it will be automatically assigned a value. This is the same as in the [SHM\\_KEY](#) parameter. However, this can cause a conflict between programs using the same semaphore set and thus this parameter should be used to give a unique semaphore key for each instance.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | Depending on system environment      |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.407. SEM\_SET\_COUNT

The number of semaphore sets which apply to the communication between processes and threads. If \_SEM\_APR is set to Y, the SEM\_SET\_COUNT value will be set to the total number of server processes or threads. If \_SEM\_ARR is set to N, the SEM\_SET\_COUNT value will be set to a user-defined value or the default value.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 1                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.408. SEQUENCE\_PIN\_BUCKET\_CNT

The number of buckets used to manage sequence pins.

- Parameter Format

| Property       | Description |
|----------------|-------------|
| Parameter Type | Integer     |
| Default Value  | 100         |

| Property | Description                          |
|----------|--------------------------------------|
| Class    | Optional, Adjustable, Static, System |
| Range    | 1 ~ INT32_MAX                        |

### 1.3.409. SERVER\_LOAD\_TOLERANCE

SERVER\_LOAD\_TOLERANCE specifies the maximum limit to be used for server load balancing.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 5                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 0 - (ms/#)                            |

### 1.3.410. SERVER\_SIDE\_LOAD\_BALANCE

SERVER\_SIDE\_LOAD\_BALANCE specifies server load-balancing behavior. NONE: Server load balancing is not performed. SHORT: The load is balanced based on differences in the processing speed of servers. LONG: The load is balanced based on differences in the session count of servers.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | "NONE"                                |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | NONE   SHORT   LONG                   |

### 1.3.411. SERVICE\_CONTROL\_STOP\_DOWNMODE

SERVICE\_CONTROL\_STOP\_DOWNMODE specifies a server down mode used to suspend a DBMS service from Windows Service Manager.

- Parameter Format

| Property       | Description                                                                                             |
|----------------|---------------------------------------------------------------------------------------------------------|
| Parameter Type | String                                                                                                  |
| Default Value  | "IMMEDIATE"                                                                                             |
| Class          | Optional, Adjustable, Static, System                                                                    |
| Range          |                                                                                                         |
| Syntax         | {normal post_tx immediate abort switchover abnormal NORMAL POST_TX IMMEDIATE ABORT SWITCHOVER ABNORMAL} |

### 1.3.412. SESS\_IGNORE\_INVALID\_CLIENT\_NLS\_SETTING

SESS\_IGNORE\_INVALID\_CLIENT\_NLS\_SETTING specifies whether to ignore result of format validation check for client's NLS parameter.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.413. SHARED\_PARAMETER\_FILE

SHARED\_PARAMETER\_FILE specifies the parameter file that can be shared. (deprecated)

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | String                               |
| Default Value  | ""                                   |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.414. SHM\_KEY

SHM\_KEY specifies the key of the shared memory to be used by each instance. In general, if this parameter is not defined, it will be automatically assigned a value. However, this can cause a conflict between programs using the same shared memory key and thus this parameter should be used to give a unique shared memory key for each instance.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | Depending on system environment      |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.415. SHP\_ALLOC\_DUMP\_IDX

The allocator index to dump when SHP\_ALLOC\_FULL\_DUMP=N. (Deprecated)

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 0                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.416. SHP\_ALLOC\_FULL\_DUMP

Determines whether to record detailed information when dumping the shared pool allocator.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

## 1.3.417. SLAB\_BLOCK\_RESERVED\_COUNT

SLAB\_BLOCK\_RESERVED\_COUNT specifies the number of blocks maintained in the slab allocator.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 4                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 2 - 1024                              |

## 1.3.418. SLOG\_DEST

SLOG\_DEST specifies the directory where system log files are stored. It must be specified as an absolute path.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | Depending on system environment       |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

## 1.3.419. SLOG\_DISPLAY\_YEAR

Option to output a year in sys.log generated by Tiberio. If set to Y, the log format is YYYY/MM/DD. If set to N, it is MM/DD.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | Y                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.420. SLOG\_FILE\_SIZE

SLOG\_FILE\_SIZE specifies the maximum size of each system log file. If a single system log file grows more than the size specified in this parameter, the current system log file will be closed and a new file will be created.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 104857600                             |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 100K - 1G                             |

### 1.3.421. SLOG\_HISTORY

SLOG\_HISTORY determines whether to gather system log history.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | Y                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.422. SLOG\_HISTORY\_BUFSIZE

SLOG\_HISTORY\_BUFSIZE specifies the size of a system log history buffer.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 256                                  |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.423. SLOG\_HISTORY\_QSIZE

SLOG\_HISTORY\_QSIZE specifies the number of system log on which histories are to be gathered.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 4                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.424. SLOG\_TOTAL\_SIZE\_LIMIT

SLOG\_TOTAL\_SIZE\_LIMIT specifies the maximum total file size that can be stored in the directory set in SLOG\_DEST. Use this parameter to prevent database log files from growing endlessly by deleting the oldest files when the size of files exceeds the specified value and hence reusing the disk space.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 1024 * 1024 * 1024                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 100K - INT64_MAX                      |

### 1.3.425. SLOG\_VIEW\_SIZE

SLOG\_VIEW\_SIZE specifies the maximum size of logs to load on v\$slog. If the value is set to 0, all logs in sys.log will be loaded on v\$slog.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 1048576                               |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 0 - 1G                                |

### 1.3.426. SMR\_DELAY\_TIME

Delay redo apply in SMR by this time. (in minutes)

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 0                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 0 - 525600                            |

### 1.3.427. SMR\_SGMT\_DROP\_TIMEOUT

Number of seconds to wait for cursors that are using segments that need to be dropped. SMR will invalidate those cursors after the number of seconds defined in SMR\_SGMT\_DROP\_TIMEOUT. If SMR\_SGMT\_DROP\_TIMEOUT is defined as 0, SMR will wait until those cursors are closed.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 0                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 0 ~ UINT64_MAX                        |

### 1.3.428. SORT\_AREA\_SIZE

SORT\_AREA\_SIZE specifies the maximum size of memory that RDBMS will allow for a one-pass sorting. The data which exceeds the specified range is sorted using two-pass sorting. If this parameter is specified too large, all sort operations will be performed using a one-pass sort, which leads to a drastic decrease in the number of SQL queries that can be simultaneously handled. If it is specified too small, all operations will be performed using two-pass and thus SQL queries take longer to execute. It is recommended to run a test with SQL queries to determine the memory usage required to execute each SQL query and check its frequency and degree of importance using the trace file. After determining an appropriate value based on this information, 10% - 50% should be added to the value for the proper memory size.

- Parameter Format

| Property       | Description                                                                              |
|----------------|------------------------------------------------------------------------------------------|
| Parameter Type | Integer                                                                                  |
| Default Value  | ( <a href="#">MEMORY_TARGET</a> - <a href="#">TOTAL_SHM_SIZE</a> ) * 0.3                 |
| Class          | Optional, Adjustable, Dynamic, Session                                                   |
| Range          | ( <a href="#">DB_BLOCK_SIZE</a> * <a href="#">DFLT_DB_FILE_MULTIBLOCK_READ_COUNT</a> ) - |

### 1.3.429. SQLTUNE\_CATEGORY

SQL tuning category. The Optimizer applies a profile that matches this category.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | String                                 |
| Default Value  | "DEFAULT"                              |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          | up to 128 characters                   |

### 1.3.430. SQL\_LOG\_ON\_MEMORY

SQL\_LOG\_ON\_MEMORY determines whether to write per-thread SQL logs in cacheable memory on file when an error occurs.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | Y                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.431. SQL\_LOG\_ON\_MEMORY\_SIZE

SQL\_LOG\_ON\_MEMORY\_SIZE specifies the size of the buffer which each thread will use to write logs to file when the SQL\_LOG\_ON\_MEMORY parameter is enabled.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 102400                                |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 100K - 10M                            |

### 1.3.432. SQL\_STAT\_HISTORY

SQL\_STAT\_HISTORY determines whether to gather statistics on SQL statements.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.433. SQL\_STAT\_HISTORY\_QSIZE

SQL\_STAT\_HISTORY\_QSIZE specifies the number of SQL statements per thread on which statistics are to be gathered.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 10                                   |
| Class          | Optional, Adjustable, Static, System |
| Range          | 1 ~ 10000                            |

### 1.3.434. SQL\_STAT\_HISTORY\_THRESHOLD

SQL\_STAT\_HISTORY\_THRESHOLD specifies the data gathering threshold for SQL statements. With SQL\_STAT\_HISTORY enabled, SQL statements are gathered if the specified execution period (ms) is exceeded.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 100                                   |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.435. SQL\_TRACE

SQL\_TRACE determines whether to record SQL trace information or not. The SQL trace enables profiling of SQL execution information. However, since using this function affects server performance, it is recommended not to use this function in a real server operating environment.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Boolean                                |
| Default Value  | N                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          |                                        |

### 1.3.436. SQL\_TRACE\_BACKUP\_LIMIT\_CNT

Specifies the maximum number of backups that can be stored in SQL\_TRACE\_DEST. For details, refer to the description of TRACE\_BACKUP\_LIMIT\_CNT. Consequently, the maximum path size may approximate  $\text{SQL\_TRACE\_TOTAL\_SIZE\_LIMIT} * (\text{SQL\_TRACE\_BACKUP\_LIMIT\_CNT} + 1)$ .

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Integer                                |
| Default Value  | <a href="#">TRACE_BACKUP_LIMIT_CNT</a> |
| Class          | Optional, Adjustable, Dynamic, System  |
| Range          | 2 ~ 500                                |

### 1.3.437. SQL\_TRACE\_DEST

SQL\_TRACE\_DEST specifies the directory where the SQL trace file will be stored. It must be specified as an absolute path.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | Depending on system environment       |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.438. SQL\_TRACE\_LEVEL

Controls the amount of information saved in the SQL trace. [1]: Traces basic information (default), [4]: Level 1 + Bind information, [8]: Level 1 + Wait information, [12]: Level 1 + Level 4 + Level 8

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Integer                                |
| Default Value  | 1                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          | 1   4   8   12                         |

### 1.3.439. SQL\_TRACE\_TOTAL\_SIZE\_LIMIT

Specifies the maximum total file size that can be stored in the directory specified in SQL\_TRACE\_DEST, excluding backup directories. For details, refer to the description of TRACE\_TOTAL\_SIZE\_LIMIT.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Integer                                |
| Default Value  | <a href="#">TRACE_TOTAL_SIZE_LIMIT</a> |
| Class          | Optional, Adjustable, Dynamic, System  |
| Range          | 100K - INT64_MAX                       |

## 1.3.440. SQL\_TRANSLATION\_PROFILE

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | String                                 |
| Default Value  | ''''                                   |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          |                                        |

## 1.3.441. SSVR\_FC\_DIRTY\_BIN\_RATIO\_MAX

SSVR\_FC\_DIRTY\_BIN\_RATIO\_MAX specifies the maximum ratio of dirty bin to keep in the storage server flash cache.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 80                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

## 1.3.442. SSVR\_FC\_FREE\_BIN\_RATIO

SSVR\_FC\_FREE\_BIN\_RATIO specifies the ratio of free bin to keep in the storage server flash cache.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 10                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.443. SSVR\_FC\_WS\_CNT

SSVR\_FC\_WS\_CNT specifies the number of working sets per device in the storage server flash cache.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | MAX(1, get_ncpu())                   |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |
| Syntax         | 1 ~ INT32_MAX                        |

### 1.3.444. SSVR\_RECV\_PORT\_START

SSVR\_RECV\_PORT\_START specifies the listening port of the storage server.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | Depending on system environment      |
| Class          | Optional, Adjustable, Static, System |
| Range          | 1024 - 65535                         |

### 1.3.445. SSVR\_TPM\_SENDER\_INTERVAL

Specifies the interval (in 0.1 s) for checking/retrying sender connections when TPM is activated in SSVR (Storage Server).

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 50                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.446. SSVR\_USE\_AGNT

Specifies whether or not to activate AGNT process that enables background utilities, such as callstack dump, in SSVR (Storage Server).

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.447. SSVR\_USE\_FC

SSVR\_USE\_FC specifies whether or not the storage server uses flash cache.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | Y                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.448. SSVR\_USE\_IB

SSVR\_USE\_IB specifies whether or not to use InfiniBand for server connections.

- Parameter Format

| Property       | Description                                                   |
|----------------|---------------------------------------------------------------|
| Parameter Type | Boolean                                                       |
| Default Value  | <a href="#">USE_ZETA</a> ? ! <a href="#">SSVR_USE_TCP</a> : N |
| Class          | Optional, Adjustable, Static, System                          |
| Range          |                                                               |

### 1.3.449. SSVR\_USE\_SDM

SSVR\_USE\_SDM specifies whether or not the storage server uses Storage Data Map (SDM).

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | Y                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.450. SSVR\_USE\_TCP

SSVR\_USE\_TCP specifies whether or not to use TCP for server connections.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.451. SSVR\_USE\_TPM

Specifies whether or not to activate TPM (Tibero Performance Monitor) in SSVR (Storage Server).

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

## 1.3.452. SSVR\_WTHR\_CNT

ssvr worker threads count

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | MAX(1, get_ncpu())                   |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

## 1.3.453. STANDBY\_ALLOWED\_IP\_ADDRESSES

IP addresses of primary allowed for standby connection. Specifying nothing means all IP addresses allowed. Existing connections are not affected when change this parameter dynamically.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ''                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |
| Syntax         | <IP_ADDRESS>;<IP_ADDRESS>,...         |

## 1.3.454. STANDBY\_ENABLE\_CASCADING

Allows LNW to send Redo Log to another node in recovery mode

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.455. STANDBY\_ENABLE\_LOG\_RECOVERY

Determines whether to send redo log for a dead node in TAC standby.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.456. STANDBY\_ENABLE\_TAC\_MULTINODE

Enable two nodes or more to boot for standby in TAC-standby.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.457. STANDBY\_FILE\_NAME\_CONVERT

STANDBY\_FILE\_NAME\_CONVERT specifies the prefix conversion rule used for converting the Primary DB path to the Standby DB path.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | String                               |
| Default Value  | ""                                   |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |
| Syntax         | {<primary path>,<standby path>}, ... |

## 1.3.458. STANDBY\_FLUSH\_WAIT

Determines whether to send an ack message after flushing the redo log received by standby.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | Y                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

## 1.3.459. STANDBY\_LOG\_ARCHIVE\_FORMAT

Specifies the format of the standby archive log file.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | "st-t%t-r%r-s%s.arc"                  |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

## 1.3.460. STANDBY\_USE\_OBSERVER

Option to enable observer in STANDBY

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.461. STAR\_TRANSFORMATION\_ENABLED

STAR\_TRANSFORMATION\_ENABLED enables or disables star transformation.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | String                                 |
| Default Value  | "FALSE"                                |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          | TRUE   FALSE                           |

### 1.3.462. STORAGE\_PROCESSING

STORAGE\_PROCESSING specifies whether or not the databases use smart scan.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Boolean                                |
| Default Value  | <a href="#">USE_ZETA</a>               |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          |                                        |

### 1.3.463. SUMMARY\_FIRST\_IN\_GROUPBY\_CUBE

Displays the summary row before the sub-total row of each group in the result set of Group By Cube().

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.464. SVR\_OUTFILE\_BACKUP\_LIMIT\_CNT

Specifies the maximum number of backups that can be stored in SVR\_OUTFILE\_DEST. For details, refer to the description of OUTFILE\_PATH\_BACKUP\_LIMIT\_CNT.

- Parameter Format

| Property       | Description                                   |
|----------------|-----------------------------------------------|
| Parameter Type | Integer                                       |
| Default Value  | <a href="#">OUTFILE_PATH_BACKUP_LIMIT_CNT</a> |
| Class          | Optional, Adjustable, Static, System          |
| Range          | 2 ~ 500                                       |

### 1.3.465. SVR\_OUTFILE\_DEST

Specifies the directory where the server outfiles are stored. Must be specified as an absolute path.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | String                               |
| Default Value  | Depending on system environment      |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.466. TABLE\_GLOBAL\_COMPRESSION

TABLE\_GLOBAL\_COMPRESSION determines whether to perform compression using a global symbol table.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Boolean                                |
| Default Value  | N                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          |                                        |

### 1.3.467. TAS5\_STORAGE\_MODE

TAS5\_STORAGE\_MODE specifies whether or not to use TAS5 with databases or storage servers.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.468. TAS\_DB\_UUID

TAS\_DB\_UUID specifies a unique ID to access cloud storage VM.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | String                               |
| Default Value  | "DBUUID00"                           |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.469. TAS\_HOT\_BACKUP\_SYNC\_WITH\_TAC

Synchronize TAS with TAC in the hot-backup state.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

## 1.3.470. TAS\_OPEN\_FILE\_MAX

TAS\_OPEN\_FILE\_MAX specifies the maximum number of files that can be opened in the cloud storage.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 4096                                 |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

## 1.3.471. TAS\_SERVER\_IP

TAS\_SERVER\_IP specifies the IP of databases or storage servers used in TCS configuration.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | String                               |
| Default Value  | "127.0.0.1"                          |
| Class          | Optional, Persistent, Static, System |
| Range          |                                      |

## 1.3.472. TAS\_VM\_ADDR

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | String                               |
| Default Value  | "127.0.0.1"                          |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.473. TAS\_VM\_PORT

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | String                               |
| Default Value  | "32000"                              |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.474. TCP\_RCVBUF\_SIZE

TCP\_RCVBUF\_SIZE specifies the size of the TCP receive buffer

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 0                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.475. TCP\_SNDBUF\_SIZE

TCP\_SNDBUF\_SIZE specifies the size of the TCP send buffer

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 0                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.476. TDE\_HASH\_VERIFICATION

Verify data integrity during decryption.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.477. TDE\_SELF\_UPDATE\_MODIFICATION\_DEFENCE

Defend against the self update modification attack. Update will be changed to delete and insert, and performance will be degraded. Rowid will also be changed: SELECT FOR UPDATE will not be supported.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.478. TEMP\_GRANULE\_SIZE

TEMP\_GRANULE\_SIZE specifies the number of blocks to be allocated for each node when managing the temporary tablespace of each node in the TAC environment.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 1000                                 |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.479. TEMP\_TS\_CNT\_MAX

The maximum number of temp tablespace that can be created.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 10                                   |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

## 1.3.480. THREAD

THREAD specifies the redo thread number to be used by instances.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 0                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          | 0 - 1023                             |

## 1.3.481. THROW\_WHEN\_GETTING\_OSSTAT\_FAIL

THROW\_WHEN\_GETTING\_OSSTAT\_FAIL specifies whether or not to throw an error when a call to get the 'osstat' information fails because the file doesn't exist in the system.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

## 1.3.482. TIBERO\_PERFORMANCE\_REPOSITORY

TIBERO\_PERFORMANCE\_REPOSITORY determines whether to activate the TPR feature.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | Y                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.483. TOTAL\_SHM\_MAX\_SIZE

TOTAL\_SHM\_MAX\_SIZE specifies the potential total size of shared memory size used by instances.

- Parameter Format

| Property       | Description                                                      |
|----------------|------------------------------------------------------------------|
| Parameter Type | Integer                                                          |
| Default Value  | <a href="#">TOTAL_SHM_SIZE</a>                                   |
| Class          | Optional, Adjustable, Static, System                             |
| Range          | 32 bit machine: 1M - UINT32_MAX, 64 bit machine: 1 M - INT64_MAX |

### 1.3.484. TOTAL\_SHM\_SIZE

TOTAL\_SHM\_SIZE specifies the current total size of shared memory size used by instances.

- Parameter Format

| Property       | Description                                                      |
|----------------|------------------------------------------------------------------|
| Parameter Type | Integer                                                          |
| Default Value  | 0                                                                |
| Class          | Mandatory, Adjustable, Dynamic, System                           |
| Range          | 32 bit machine: 1M - UINT32_MAX, 64 bit machine: 1 M - INT64_MAX |

### 1.3.485. TPR\_AGGREGATION

TPR\_AGGREGATION determines whether to aggregate information of snapshots removed due to retention over into one predetermined snapshot. (Deprecated)

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.486. TPR\_METRIC

TPR\_METRIC determines whether to collect metric statistics.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.487. TPR\_PARTITION\_COUNT\_IN\_ADVANCE

TPR\_PARTITION\_COUNT\_IN\_ADVANCE specifies the number of partitions to prepare in advance.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 2                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 0 ~ 365                               |

### 1.3.488. TPR\_SEGMENT\_STATISTICS

TPR\_SEGMENT\_STATISTICS determines whether to collect segment statistics.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.489. TPR\_SNAPSHOT\_RETENTION

TPR\_SNAPSHOT\_RETENTION specifies the period, in days, to retain TPR snapshot data.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 7                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 1 ~ 1500                              |

### 1.3.490. TPR\_SNAPSHOT\_SAMPLING\_INTERVAL

TPR\_SNAPSHOT\_SAMPLING\_INTERVAL specifies the interval, in minutes, to sample TPR snapshot data.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 60                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 1 ~ 50400                             |

### 1.3.491. TPR\_SNAPSHOT\_TOP\_SEGMENT\_CNT

TPR\_SNAPSHOT\_TOP\_SEGMENT\_CNT specifies the count of segments to display in tpr report.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 5                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          | 5 ~ 1000                             |

### 1.3.492. TPR\_SNAPSHOT\_TOP\_SQL\_CNT

TPR\_SNAPSHOT\_TOP\_SQL\_CNT specifies the count of SQLs to display in tpr report.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 10                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 5 ~ 1000                              |

### 1.3.493. TPR\_SQL\_PLAN\_STAT

TPR\_SQL\_PLAN\_STAT determines whether to collect statistics on an instance in the SQL statement execution plan at the time an SQL query is executed (only for TPR).

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.494. TRACE\_BACKUP\_LIMIT\_CNT

Specifies the maximum number of backups that can be stored in MEM\_LOG\_DEST and SQL\_TRACE\_DEST. This parameter prevents trace file backup directories from unlimited growth. When the backup counts exceed the specified value, the oldest backup files are removed to allow disk space

reuse. Backup operations are performed under the following conditions: 1. Files exceed the maximum size limit 2. Database restart occurs Consequently, the maximum path size may approximate  $\text{TRACE\_TOTAL\_SIZE\_LIMIT} * (\text{TRACE\_BACKUP\_LIMIT\_CNT} + 1)$ . Used as the default value for `MEM_LOG_BACKUP_LIMIT_CNT` and `SQL_TRACE_BACKUP_LIMIT_CNT`.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 10                                   |
| Class          | Optional, Adjustable, Static, System |
| Range          | 2 ~ 500                              |

### 1.3.495. TRACE\_DUMP\_BACKUP\_LIMIT\_CNT

Specifies the maximum number of backups that can be stored in `TRACE_DUMP_DEST`. For details, refer to the description of `DFLT_DUMP_BACKUP_LIMIT_CNT`. Consequently, the maximum path size may approximate  $\text{TRACE\_DUMP\_BACKUP\_TOTAL\_SIZE\_LIMIT} * (\text{TRACE\_DUMP\_BACKUP\_LIMIT\_CNT} + 1)$ .

- Parameter Format

| Property       | Description                                |
|----------------|--------------------------------------------|
| Parameter Type | Integer                                    |
| Default Value  | <a href="#">DFLT_DUMP_BACKUP_LIMIT_CNT</a> |
| Class          | Optional, Adjustable, Dynamic, System      |
| Range          | 2 ~ 500                                    |

### 1.3.496. TRACE\_DUMP\_DEST

Specifies the directory where the trace dump files are stored. Must be specified as an absolute path.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | Depending on system environment       |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.497. TRACE\_DUMP\_TOTAL\_SIZE\_LIMIT

Specifies the maximum total file size that can be stored in the directory specified in TRACE\_DUMP\_DEST, excluding backup directories. For details, refer to the description of DFLT\_DUMP\_TOTAL\_SIZE\_LIMIT.

- Parameter Format

| Property       | Description                                |
|----------------|--------------------------------------------|
| Parameter Type | Integer                                    |
| Default Value  | <a href="#">DFLT_DUMP_TOTAL_SIZE_LIMIT</a> |
| Class          | Optional, Adjustable, Dynamic, System      |
| Range          | 100K - INT64_MAX                           |

### 1.3.498. TRACE\_PATH\_CHECK\_INTERVAL

Monitoring cycle for cumulative space usage of trace paths excluding backup sets. (unit: 1 second) (Target: MEMLOG, SQLTRACE)

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 600                                   |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 60 ~ 3600                             |

### 1.3.499. TRACE\_TOTAL\_SIZE\_LIMIT

Specifies the maximum total file size that can be stored in the directory specified in MEM\_LOG\_DEST and SQL\_TRACE\_DEST, excluding backup directories. When the size of files exceeds the specified value, trace files are relocated to new backup directories. Used as the default value for MEM\_LOG\_TOTAL\_SIZE\_LIMIT and SQL\_TRACE\_TOTAL\_SIZE\_LIMIT.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 100 * 1024 * 1024                    |
| Class          | Optional, Adjustable, Static, System |

| Property | Description      |
|----------|------------------|
| Range    | 100K - INT64_MAX |

### 1.3.500. TSAM\_PREFETCH\_SIZE

TSAM\_PREFETCH\_SIZE specifies the size of the buffer to be prefetched when db file sequential read is enabled in TSAM API.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 4096                                  |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 0K - 64K                              |

### 1.3.501. TSN\_SYNC\_UDP\_PORT

The UDP port to be used between nodes for TSN synchronization.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 55901                                |
| Class          | Optional, Adjustable, Static, System |
| Range          | 1024 - 65535                         |

### 1.3.502. TX\_FORCE\_COMMIT\_DUMP

TX\_FORCE\_COMMIT\_DUMP determines whether to dump undo records forcibly committed.

- Parameter Format

| Property       | Description |
|----------------|-------------|
| Parameter Type | Boolean     |
| Default Value  | N           |

| Property | Description                           |
|----------|---------------------------------------|
| Class    | Optional, Adjustable, Dynamic, System |
| Range    |                                       |

### 1.3.503. UNDO\_RETENTION

UNDO\_RETENTION specifies the minimum time period in seconds for which undo can be retained.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 900                                   |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.504. UNDO\_TABLESPACE

UNDO\_TABLESPACE specifies the undo tablespace to be used by a node. If this parameter is not specified, the first available undo tablespace will be used by default. This parameter must be defined in the TAC environment.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ""                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.505. USE\_ACCURATE\_NUMERIC\_FUNC

Performs more accurate calculations on numeric functions.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Boolean                                |
| Default Value  | N                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          |                                        |

### 1.3.506. USE\_ACTIVE\_STORAGE

USE\_ACTIVE\_STORAGE specifies whether a database instance uses the Active Storage function by connecting to the Active Storage instance.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.507. USE\_AD\_AUTH

Using AD authentication or NOT (for user authentication)

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.508. USE\_BINARY\_UPDATE

USE\_BINARY\_UPDATE determines whether use or not auto-binary-update when main svr booting.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.509. USE\_CASE\_SENSITIVE\_PASSWORD

Determines whether to use case-sensitive password for users or roles.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | Y_N                                   |

### 1.3.510. USE\_DISK\_AFFINITY

AS\_USE\_DISK\_AFFINITY specifies whether disk affinity function will be used

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.511. USE\_DUMP\_PATH\_MANAGEMENT

Executes automatic backup based on path storage capacity or system boot, and performs deletion based on backup count for log files in \*\_DUMP\_DEST.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | !iparam_check_win()                   |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.512. USE\_HSM

Option to use Hardware Security Module (HSM) to save data encryption keys.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.513. USE\_HUGE\_PAGE

Option to enable the SHM\_HUGETLB flag in Linux to get shared memory segments with huge pages by using shmget().

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.514. USE\_ILOG

USE\_ILOG determines whether to record internal log.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | Y                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.515. USE\_NBU\_FOR\_ARCHIVELOG

Whether to create archive logs at NetBackup directly. If Y, no archive logs are created at local. For searching or restoring, use NBU\_ARCHIVELOG\_SEARCH.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.516. USE\_NBU\_FOR\_BACKUPSET

When using RMGR, whether to create a backupset at NetBackup or restore the backupset from NetBackup.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.517. USE\_NETWORK\_SECURITY\_AGENT\_MANAGER

specifies whether or not to use Network security agent manager.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.518. USE\_NET\_KEEPALIVE

USE\_NET\_KEEPALIVE determines whether to activate the socket option of the 'TCP keepalive'.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | Y                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.519. USE\_OUTFILE\_PATH\_MANAGEMENT

Executes automatic backup during boot and performs deletion based on backup count for paths in \*\_OUTFILE\_DEST format.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | !iparam_check_win()                   |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.520. USE\_PROFILE

Determines whether to use the PROFILE feature.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | Y                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.521. USE\_PROFILE\_LOGIN\_LOGOUT\_TIME

Determines whether to record login and logout time.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | Y                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.522. USE\_RECYCLEBIN

USE\_RECYCLEBIN determines whether to use the recyclebin function. If this feature is used, the drop table command does not actually delete a table. In order to actually delete a table, use the purge command or give the purge option in the drop table command. The flashback table feature can be used.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Boolean                                |
| Default Value  | N                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          |                                        |

### 1.3.523. USE\_RECYCLEBIN\_AUTO\_PURGE

Determines whether to purge objects in recyclebin automatically when Tablespace is full or TS\_QUOTA exceeded during segment extending.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Boolean                                |
| Default Value  | Y                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          | Y_N                                    |

### 1.3.524. USE\_RESOURCE\_MANAGER

USE\_RESOURCE\_MANAGER specifies whether to use the resource manager or not.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.525. USE\_SERVER\_EVENT\_NOTIFICATION

whether use server event notification or not.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.526. USE\_SQL\_STAT\_HIST\_IN\_BG\_PROC

Gather SQL\_STAT\_HISTORY data in background process.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.527. USE\_STANDBY\_REDO\_LOG

Whether to use standby redo log or not.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Boolean                              |
| Default Value  | N                                    |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.528. USE\_STORED\_OUTLINES

Determines whether to apply created outline.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | Boolean                                |
| Default Value  | N                                      |
| Class          | Optional, Adjustable, Dynamic, Session |
| Range          |                                        |

### 1.3.529. USE\_TRACE\_PATH\_MANAGEMENT

Executes automatic backup based on path storage capacity or system boot, and performs deletion based on backup count for log files in MEM\_LOG\_DEST and SQL\_TRACE\_DEST.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | !iparam_check_win()                   |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.530. USE\_TRUNCATE\_OBJ\_PRIVILEGE

Determines whether to use the TRUNCATE OBJ PRIVILEGE feature.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.531. USE\_TRUNCATE\_PRIVILEGE

Determines whether to use the TRUNCATE PRIVILEGE feature.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | N                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.532. USE\_TS\_QUOTA

Determines whether to use the TABLESPACE QUOTA feature.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Boolean                               |
| Default Value  | Y                                     |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.533. USE\_ZETA

USE\_ZETA specifies whether the configuration is for ZetaData or not.

- Parameter Format

| Property       | Description                                |
|----------------|--------------------------------------------|
| Parameter Type | Boolean                                    |
| Default Value  | strcmp(INSTANCE_TYPE, "SSVR") == 0 ? Y : N |
| Class          | Optional, Persistent, Static, System       |
| Range          |                                            |

### 1.3.534. USGMT\_AUTO\_SHRINK\_INTERVAL

USGMT\_AUTO\_SHRINK\_INTERVAL specifies the interval (in minutes) of occurrence of automatic undo segment shrinkage. If this parameter is set to 0, automatic undo segment shrinkage never happens.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 180                                   |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.535. USGMT\_ONLINE\_MAX

Maximum number of undo segments created during database operation. In active cluster mode, this value is the maximum number of undo segments for each node. Undo segments are created as many as USGMT\_ONLINE\_MIN when the database starts and can be created up to USGMT\_ONLINE\_MAX. If all

undo segments are occupied (at least one transaction is processed by each segment) when a transaction begins, an additional undo segment can be created. If the undo tablespace has no sufficient free space, it fails to create a new undo segment.

- Parameter Format

| Property       | Description                                   |
|----------------|-----------------------------------------------|
| Parameter Type | Integer                                       |
| Default Value  | MAX(1000, <a href="#">MAX_SESSION_COUNT</a> ) |
| Class          | Optional, Adjustable, Dynamic, System         |
| Range          |                                               |

### 1.3.536. USGMT\_ONLINE\_MIN

Number of undo segments to create when the database starts. In active cluster mode, as many undo segments as this value are created for each node. If this value is not large enough, performance may decrease because there are many attempts to create additional undo segments when multiple sessions receive many transactions at the same time. If this value is too large and the size of undo segments exceeds that of the undo tablespace, the database fails to start.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 11                                   |
| Class          | Optional, Adjustable, Static, System |
| Range          |                                      |

### 1.3.537. USGMT\_STEAL\_IGNORE\_RETENTION

It is a feature that allows the use of undo blocks without guaranteeing undo retention during undo steal. It is used to improve the utilization of undo space at the cost of a potential snapshot too old error. To minimize the chances of this error, undo segments are selected for stealing starting from those where the most recent active transaction was committed a long time ago.

- Parameter Format

| Property       | Description |
|----------------|-------------|
| Parameter Type | Boolean     |

| Property      | Description                           |
|---------------|---------------------------------------|
| Default Value | N                                     |
| Class         | Optional, Adjustable, Dynamic, System |
| Range         |                                       |

### 1.3.538. UTL\_FILE\_DIR

UTL\_FILE\_DIR specifies the default directory used by the UTL\_FILE package. If this parameter is not specified, the directory specified in PSM\_SHLIB\_DIR will be used by default. (deprecated)

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | String                                |
| Default Value  | ""                                    |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          |                                       |

### 1.3.539. WALLET\_FILE

WALLET\_FILE specifies the location of the wallet files. It must be specified as an absolute path.

- Parameter Format

| Property       | Description                            |
|----------------|----------------------------------------|
| Parameter Type | String                                 |
| Default Value  | ""                                     |
| Class          | Optional, Adjustable, Static, System   |
| Range          |                                        |
| Syntax         | WALLET_FILE=filename[:filename [...] ] |

### 1.3.540. WTHR\_PER\_PEP

WTHR\_PER\_PEP specifies the number of worker threads per PE process(PEP) .

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | MAX(4, MIN(64, get_ncpu() / 2))      |
| Class          | Optional, Adjustable, Static, System |
| Range          | 2 ~ 10000                            |

### 1.3.541. WTHR\_PER\_PROC

Number of worker threads in a worker process.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 10                                   |
| Class          | Optional, Adjustable, Static, System |
| Range          | 1 ~ 10000                            |

### 1.3.542. WTHR\_PROC\_CNT

WTHR\_PROC\_CNT specifies the number of working processes.

- Parameter Format

| Property       | Description                                     |
|----------------|-------------------------------------------------|
| Parameter Type | Integer                                         |
| Default Value  | <a href="#">MAX_SESSION_COUNT/WTHR_PER_PROC</a> |
| Class          | Optional, Adjustable, Static, System            |
| Range          | 1 - 10000                                       |

### 1.3.543. WTHR\_PROC\_CNT\_MAX

WTHR\_PROC\_CNT\_MAX specifies the maximum number of working processes. With RDBMS started, as many working processes as defined in WTHR\_PROC\_CNT are implemented. If the number of working processes becomes insufficient during the operation of RDBMS, the number of working processes will be increased within the limit specified in this parameter. If the WTHR\_PROC\_CNT\_MAX parameter is

set to the default value, the number of working processes defined in WTHR\_PROC\_CNT will be used as the maximum number of processes to run.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | <a href="#">WTHR_PROC_CNT</a>        |
| Class          | Optional, Adjustable, Static, System |
| Range          | 1 - 10000                            |

### 1.3.544. XA\_MAX\_BRANCH\_CNT

XA\_MAX\_BRANCH\_CNT specifies the maximum number of XA transaction branches that can run one instance at a time. Setting this value too large could result in excessive memory usage.

- Parameter Format

| Property       | Description                          |
|----------------|--------------------------------------|
| Parameter Type | Integer                              |
| Default Value  | 1024                                 |
| Class          | Optional, Adjustable, Static, System |
| Range          | 256 - 8192                           |

### 1.3.545. XA\_TIMER\_INTERVAL

XA\_TIMER\_INTERVAL specifies the interval for how often XA session timeouts are checked. The interval is specified in 0.1 seconds.

- Parameter Format

| Property       | Description                           |
|----------------|---------------------------------------|
| Parameter Type | Integer                               |
| Default Value  | 100                                   |
| Class          | Optional, Adjustable, Dynamic, System |
| Range          | 10 - 600 (1 - 60 seconds)             |



# 제2장 데이터 사전

본 장에서는 데이터 사전에 대한 개념과 구조를 설명하고 데이터 사전의 참조 및 갱신을 기술한다.

## 2.1. 개요

**데이터 사전(Data Dictionary)**은 데이터베이스 관리 시스템(Database Management System, 이하 DBMS)을 효율적으로 사용하기 위해 데이터베이스에 저장된 정보를 요약한 것이다. 즉 일련의 데이터를 정의하고 설명해 주는 메타데이터(metadata)를 유지하는 것을 의미한다.

데이터 사전은 데이터 정보를 분류하고 처리하기 위한 시스템과 절차로서 데이터를 이해하는 과정에서 발생하는 오류, 또는 데이터 해석상의 어려움을 제거하기 위해 사용된다.

데이터 사전은 테이블과 뷰로 정의되며, 사용자 유형별로 접근 권한이 다르다.

| 사용자 유형 | 설명                                  |
|--------|-------------------------------------|
| DBA    | 직접 접근이 가능하며, 데이터 사전을 직접 갱신할 수 있다.   |
| 일반 사용자 | 뷰를 통해 접근이 가능하며, 데이터 사전을 읽기만 할 수 있다. |

데이터 사전에 저장되는 정보는 다음과 같다.

- 데이터베이스 스키마 객체 정보  
테이블, 인덱스, 뷰 등이 있다.
- 사용자 정보  
이름, 권한 등이 있다.
- 테이블과 컬럼 속성 정보  
사용 가능한 디스크 용량, 컬럼 타입, 기본값 등이 있다.
- 테이블의 무결성 제약조건 정보  
기본 키, 외래 키, NULL 값 허용 여부 등이 있다.
- 디스크 구조 정보  
테이블 스페이스, 데이터 파일, 세그먼트, 사용자 및 테이블 등이 있다.
- 히스토그램(Histogram) 정보  
주로 많이 사용되는 컬럼 값, 컬럼 값의 분포 등이 있다.

## 2.2. 데이터 사전의 구조

Tibero 의 데이터 사전은 크게 다음과 같이 구성된다.

- 기반 테이블(Base Table)
- 정적/동적 뷰
  - 정적 뷰(Static View)
  - 동적 뷰(Dynamic View)

데이터 사전을 위와 같이 분리하여 구성하는 이유는 데이터베이스 시스템의 일관성을 유지하고 사용자에게 편의성을 제공하며 데이터베이스의 저장 공간을 절약하기 위함이다. 만약 일반 사용자의 부주의로 기반 테이블의 내용을 직접 갱신하였다면, 데이터베이스 시스템에 심각한 에러를 발생시킬 수 있다.

### 2.2.1. 기반 테이블

**기반 테이블**은 중복된 데이터를 제거하는 정규화 과정을 거친 테이블이다. 기반 테이블 안에 일부 데이터는 플래그(Flag)로 나타나며, 암호화 되어 있다. 또한, 기반 테이블의 자체 데이터도 기반 테이블에 저장된다.

기반 테이블은 정규화를 거친 데이터가 저장되므로 중복된 데이터가 거의 존재하지 않는다. 또한, 기반 테이블은 데이터 사전을 저장하는 데 필요한 디스크 공간을 줄이며 중복된 데이터가 일관성 없이 갱신될 가능성을 사전에 예방하는 역할을 수행한다.

기반 테이블의 내용을 직접 참조할 수 있는 사용자는 오직 **DBA**만이 할 수 있다. 일반 사용자는 데이터 사전을 참조하는 기반 테이블의 내용을 참조하려면 정적/동적 뷰를 이용해야 한다. 왜냐하면, 모든 데이터 사전을 참조하는 기반 테이블의 소유자는 **SYS** 사용자이기 때문이다.

---

#### 참고

**SYS** 사용자는 Tibero에서 자체적으로 생성한 사용자이며, **DBA**를 포함한 모든 사용자의 상위에 존재하는 사용자이다. 또한, 이 사용자는 Tibero 자체를 관리하는 역할을 수행한다. **DBA**나 기타 어떠한 권한을 가진 사용자도 이 사용자를 대신할 수 없다.

---

또한, 기반 테이블을 직접 갱신할 수 있는 사용자는 **DBA**뿐이며, 일반 사용자는 데이터 정의어(Data Definition Language, 이하 DDL) 문장을 통해서만 가능하다. DDL 문장을 실행하려면 반드시 대상 객체에 대한 DDL 문장의 실행 권한이 있어야 한다. 예를 들어 EMP 테이블의 외래 키 컬럼을 변경하려면 현재 사용자가 EMP 테이블에 대한 갱신 권한이 있어야 한다는 의미이다.

그리고 기반 테이블에 대한 관리를 데이터 사전이 함으로써 일반 사용자는 기반 테이블의 정의를 몰라도 간단한 DDL 문장으로 데이터 사전 작업을 수행할 수 있다.

예를 들어 일반 사용자가 기반 테이블을 생성 하려면 다음과 같은 순으로 정의해야 한다.

1. 객체 정의

기본 테이블인 **\_DD\_OBJ** 테이블에 스키마 객체를 등록한다.

## 2. 테이블 정의

기본 테이블인 **\_DD\_TBL** 테이블에 하나의 로우를 삽입한다.

## 3. 컬럼 정의

기본 테이블인 **\_DD\_COL** 테이블에 컬럼 개수에 따라 로우를 삽입한다. 만약 인덱스를 선언하였거나 제약조건을 정의한 경우, 기본 테이블에도 개수에 따라 로우를 삽입해야 한다.

## 2.2.2. 정적/동적 뷰

**정적 뷰**는 하나 이상의 기본 테이블과 다른 정적/동적 뷰를 기반으로 정의된 뷰이다. 따라서 정적 뷰는 기본 테이블과 정적/동적 뷰 간의 조인을 이용하여 생성된다.

**동적 뷰**는 Tibero 에서 동적으로 변경되는 데이터를 테이블 형태로 보여주는 뷰이다. 이처럼 동적으로 변경되는 데이터의 예를 들면, 현재 서버 프로세스의 상태, 트랜잭션의 수행 상태, 사용자 권한 등이다. 이러한 데이터는 실제로 디스크에 저장되지 않으며, 사용자가 동적 뷰를 사용할 때마다 실시간으로 테이블 형태로 생성된다.

## 2.3. 데이터 사전의 참조

Tibero에서 데이터 사전이 참조되는 경우는 다음과 같이 크게 세 가지로 나눌 수 있다.

- 일반 사용자에게 의한 참조
- SQL 파서(Parser), 질의 최적화기에 의한 참조
- Tibero에 의한 참조

### 2.3.1. 일반 사용자에게 의한 참조

일반 사용자는 정적/동적 뷰를 이용하여 데이터 사전을 참조한다. 데이터 사전을 참조하는 방법은 정적/동적 뷰를 SQL 문장에 포함하여 실행하는 것이다.

다음은 현재 사용자가 정의한 모든 테이블의 이름을 출력하는 예이다. 본 예에서는 모든 테이블에 대한 정보를 포함하는 정적 뷰가 **USER\_TABLES** 이므로, 다음과 같은 SQL 문장을 실행한다.

```
SELECT TABLE_NAME FROM USER_TABLES;
```

일반 사용자는 위와 같이 정적 뷰를 이용하여 데이터 사전을 참조할 수 있다.

스키마 객체와 관련된 정적 뷰는 사용자가 접근할 수 있는 범위에 따라 크게 세 가지로 나뉘며, 다음과 같이 접두어로 구분된다.

| 접두어   | 설명                           |
|-------|------------------------------|
| USER_ | 현재 사용자가 정의한 객체에 대한 뷰이다.      |
| ALL_  | 현재 사용자가 접근할 수 있는 객체에 대한 뷰이다. |
| DBA_  | 모든 객체에 대한 뷰이다.               |

#### 참고

정적 뷰는 위 표와 같이 접두어만 다르며, 모두 같은 컬럼으로 구성되어 있다. 단, **USER\_** 뷰는 컬럼 소유자(OWNER)를 포함하지 않는다.

## 2.3.2. SQL 파서, 질의 최적화기에 의한 참조

### SQL 파서

Tibero는 사용자로부터 입력된 SQL 문장을 해석할 때 데이터 사전을 이용한다. 이러한 과정을 의미 분석 (Semantic Analysis) 과정이라고 부른다.

**의미 분석 과정**은 SQL 문장에 포함된 스키마 객체가 존재하는지, 현재 사용자가 접근할 수 있는지, 특정 테이블에 포함된 컬럼은 어떤 것이 있는지, 컬럼의 타입에 따라 맞게 연산을 실행하는지 등의 검사를 수행한다.

다음의 SQL 문장으로 의미 분석 과정을 설명한다.

```
SELECT C11
FROM T1, T2
WHERE C11 = C21 AND C22 LIKE '20%';
```

- ① **SELECT** 문장에 포함된 스키마 객체 즉 T1, T2가 존재하는지, 그리고 이 객체에 현재 사용자가 접근할 수 있는지 검사한다.
- ② 접근에 문제가 없다면, C11, C21, C22 컬럼이 T1, T2 테이블에 각각 정의되어 있는지, 아니면 동시에 T1, T2 테이블에 정의되어 있는지(또는 모호성이 존재하는지.)를 검사한다.
- ③ C11과 C12 컬럼이 비교 연산이 가능한지, C22 컬럼이 **LIKE** 연산자를 수행할 수 있는 문자열 타입인지 검사한다.

위의 SQL 문장은 의미 분석 과정을 거친 후 각 컬럼이 포함된 테이블을 명시적으로 표시하고 다음과 같이 변환된다.

```
SELECT T1.C11
FROM T1, T2
WHERE T1.C11 = T2.C21 AND T2.C22 LIKE '20%';
```

## 질의 최적화기

Tibero는 질의 최적화기를 이용하여 SQL 문장에 대한 효율적인 실행계획(Execution Plan)을 생성하기 위해 데이터 사전에 저장된 통계 정보를 참조한다.

이러한 통계 정보는 각 스키마 객체의 기반 테이블에 저장되어 있으며, 히스토그램을 저장하는 기반 테이블에도 저장되어 있다.

예를 들어 테이블 정의를 포함하는 기반 테이블에는 테이블 안에 로우의 개수, 디스크 블록의 개수, 하나의 디스크 블록에 포함된 로우의 평균 개수 등이다. 히스토그램 테이블에는 특정 컬럼에 대하여 컬럼 값의 분포, 자주 사용되는 컬럼 값 등을 저장하고 있다.

또한, 질의 최적화기는 인덱스에 대한 정보도 참조한다. 예를 들어 테이블에 어떤 컬럼이 인덱스가 생성되어 있는지, 인덱스의 단말 노드(Terminal Node)에 저장된 레코드의 평균 개수는 몇 개인지 등을 참조한다.

다음은 질의 최적화기가 어떤 데이터 사전 정보를 이용하는지 확인할 수 있는 예이다.

```
SELECT *  
FROM T1, T2  
WHERE T1.C1 = T2.C2;
```

위 예에서 보듯이 SELECT 문장은 WHERE 절의 조건에 따라 조인 연산을 수행해야 한다. 일반적으로 조인 연산은 시스템의 메모리와 CPU 시간이 많이 요구되는 연산이다. 따라서, 될 수 있으면 적은 메모리를 사용하여 조인 연산을 실행하는 방법을 찾아야 한다.

테이블 T1과 T2에 대하여 중첩 루프 조인(Nested-Loop Join)을 수행하는 방법은 다음과 같다.

- T1을 외부 루프(Outer Loop)로 한다.
- T2를 내부 루프로 한다.

일반적으로, 포함된 로우의 개수가 적은 테이블을 외부 루프 테이블로 하는 것이 연산 비용이 적다. 따라서, 질의 최적화기는 데이터 사전의 내용을 참조하여 테이블 T1과 T2 중에서 포함된 로우의 개수가 적은 것을 외부 루프로 하여 SQL 질의를 수행하도록 한다.

질의 최적화기가 인덱스에 대한 데이터 사전의 내용을 참조한 결과 T1.C1과 T2.C2 컬럼 모두에 대하여 인덱스가 생성되었다면, 중첩 루프 조인이 아니라, 좀 더 적은 메모리 양과 CPU 작업이 필요한 정렬 병합 조인(Sort-Merge Join) 연산을 수행한다.

### 2.3.3. Tibero에 의한 참조

일반 사용자로부터 주어진 SQL 문장에 대한 의미 분석 및 질의 최적화 과정을 수행할 뿐만 아니라, SQL 문장을 실행할 때에도 데이터 사전의 내용을 참조한다.

의미 분석 및 질의 최적화 과정에서 검사하는 내용은 테이블에 저장된 실제 데이터와 관계없이 나타나는 결과이다. 그러나 SQL 문장을 실행할 때는 실제 데이터에 따라 추가로 수행해야 할 결과가 나타날 수 있다.

예를 들어 테이블 T1에 테이블 T2의 외래 키가 정의되어 있는 경우 테이블 T2에서 하나의 로우를 삭제하려고 할 때 Tibero는 데이터 사전을 참조하여 테이블 T2의 컬럼에 대한 외래 키가 정의되어 있는지 확인한다.

테이블 T1의 로우 중 T2 컬럼 값을 참조하는 로우가 존재한다면, 다음과 같은 추가 작업을 수행해 주어야 한다. 예를 들어 테이블 T2의 로우를 삭제하지 못하도록 한다든지 또는 테이블 T2의 로우를 참조하는 테이블 T1의 모든 로우를 함께 삭제하는 등의 작업을 추가로 한다. 실제 어떠한 작업을 수행할지는 데이터 사전의 내용을 참조해야 한다.

## 2.4. 데이터 사전의 갱신

데이터 사전의 갱신은 기반 테이블의 갱신을 뜻한다. 정적 뷰를 통한 갱신은 일반 사용자나 DBA도 허용되지 않으며, 동적 뷰에 대한 갱신은 자동으로 이루어지거나 다른 인터페이스를 이용해야 갱신할 수 있다. 또한, 일반 사용자는 기반 테이블에 대한 갱신을 직접 수행할 수 없다.

일반 사용자나 SYS 권한을 가진 사용자는 DDL 문장을 이용하여 안전하고 일관성 있게 데이터 사전의 내용을 갱신할 수 있다. DDL 문장을 실행하면 데이터 사전 모듈 내에서 현재 실행되고 있는 다른 트랜잭션까지 고려하여 데이터 사전의 내용을 일관성 있게 갱신해 준다.

그리고 데이터 사전의 내용과 함께 데이터베이스의 내용도 함께 갱신해야 할 경우에도 데이터 사전의 모듈 내에서 모두 수행할 수 있다.

---

### 주의

SYS 권한을 가진 사용자나 DBA만 직접 기반 테이블을 참조하고 갱신할 수 있다. 될 수 있으면 이러한 작업을 하지 않기를 권장한다.

---

### 2.4.1. DDL 문장

**DDL**은 데이터베이스에서 데이터와 데이터 간의 관계를 정의한 SQL 문장으로 데이터베이스 구조를 설정하는 구조화된 조회 언어이다. DDL은 데이터베이스의 스키마 객체를 생성(CREATE), 갱신(ALTER), 제거(DROP) 하거나 권한의 부여나 회수(GRANT, REVOKE), 주석(COMMENT), 자료의 버림(TRUNCATE) 등을 수행하는 문장 등을 포함한다.

각 문장은 CREATE, ALTER, DROP, TRUNCATE, GRANT, REVOKE, COMMENT 등으로 시작한다. 이 밖에도 많은 종류의 DDL이 존재한다.

Tibero에서 DDL 문장을 이용하여 CREATE, ALTER, DROP할 수 있는 객체는 다음과 같다.

- 데이터베이스
- 테이블
- 인덱스, 뷰, 동의어

- 사용자
- 무결성 제약조건
- 역할(Role)

위 객체 중 테이블은 생성, 갱신, 삭제는 CREATE TABLE, ALTER TABLE, DROP TABLE 문을 이용하여 수행하며, 일부 객체는 생성, 갱신, 삭제 중의 일부만 가능한 경우도 있다. 예를 들어 동의어는 CREATE SYNONYM, DROP SYNONYM 문만이 존재하고, ALTER SYNONYM 문은 존재하지 않는 경우이다.

그리고 아래와 같이 객체의 생성, 갱신, 삭제와 관련 없는 특별한 목적의 DDL 문장도 존재한다.

| DDL 문장   | 설명                           |
|----------|------------------------------|
| GRANT    | 사용자에게 사용 권한을 부여하는 문장이다.      |
| REVOKE   | 사용자에게 사용 권한을 회수하는 문장이다.      |
| SET ROLE | 사용자의 권한을 활성화 또는 비활성화하는 문장이다. |

이와 같은 DDL 문장에 의해 갱신된 데이터 사전의 내용은 기반 테이블에 저장된다. 정적 뷰를 통해 확인할 수 있으며, 일부 내용은 동적 뷰를 통해서만 확인할 수 있다.

---

## 참고

Tibero에서 지원하는 DDL 문장에 대한 내용은 "Tibero SQL 참조 안내서"를 참고한다.

---



# 제3장 Static View

## 3.1. Overview

The **static view** comprises of one or more base tables and static and dynamic views. Thus, the static view is generated from a join of static and dynamic views.

Tibero uses [DICTIONARY](#) to display the list of available static views. DBA or general users can retrieve data from the static view by executing a SELECT statement using the tbSQL tool.

## 3.2. List of Static Views

This section provides the list of static views of Tibero in alphabetical order and describes the columns which corresponds to each static view in the table.

---

### 참고

The static view of Tibero is comprised of a set of views named with a different prefix but having the same column. In this guide, the same column used for several views will be discussed only once in the list, in the first explained view.

---

The following is the list of summarized static views.

| Static View                           | Description                                                                                                                 |
|---------------------------------------|-----------------------------------------------------------------------------------------------------------------------------|
| <a href="#">ALL_ARGUMENTS</a>         | Displays all parameters of the procedure or function that the current user can call.                                        |
| <a href="#">ALL_CATALOG</a>           | Displays all tables, views, synonyms, and sequences that the current user can access.                                       |
| <a href="#">ALL_COLL_TYPES</a>        | Displays all the collection types that the current user can access.                                                         |
| <a href="#">ALL_COL_COMMENTS</a>      | Displays comments on columns of the tables or views that the current user can access.                                       |
| <a href="#">ALL_COL_PENDING_STATS</a> | Displays statistics and a histogram of columns in pending state current user can access.                                    |
| <a href="#">ALL_COL_PRIVS</a>         | Displays columns for which the current user or public user is the object owner or grantor (user who granted the privilege). |
| <a href="#">ALL_COL_PRIVS_MADE</a>    | Displays columns for which the current user is the object owner or grantor.                                                 |

| Static View                                       | Description                                                                                                                               |
|---------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------|
| <a href="#">ALL_COL_PRIVS_RECD</a>                | Displays columns for which the current user or public user is the grantee.                                                                |
| <a href="#">ALL_CONSTRAINTS</a>                   | Displays constraints defined in the table that the current user can access.                                                               |
| <a href="#">ALL_CONS_COLUMNS</a>                  | Displays columns that the current user can access and that are specified in integrity constraint definitions.                             |
| <a href="#">ALL_CONS_COLUMNS_INFO</a>             | Displays columns with constraint information that the current user can access and that are specified in integrity constraint definitions. |
| <a href="#">ALL_DB_LINKS</a>                      | Displays information for the database links that the current user can use.                                                                |
| <a href="#">ALL_DEPENDENCIES</a>                  | Displays dependencies between procedures, functions, packages, package bodies, triggers, and views that the current user can access.      |
| <a href="#">ALL_DIRECTORIES</a>                   | Displays all directories that the current user can access.                                                                                |
| <a href="#">ALL_ENCRYPTED_COLUMNS</a>             | Displays information of all encrypted columns that the current user can access.                                                           |
| <a href="#">ALL_ERRORS</a>                        | Displays the recent errors on every accessible view, procedure, function, package, package body, trigger, library, and java object.       |
| <a href="#">ALL_EXTERNAL_LOCATIONS</a>            | Displays all external table locations that the current user can access.                                                                   |
| <a href="#">ALL_EXTERNAL_TABLES</a>               | Displays all external tables that the current user can access.                                                                            |
| <a href="#">ALL_IDX_AND_LOB_COLUMNS_IN<br/>FO</a> | Displays index columns (including LOB) with index information of all tables that the current user can access                              |
| <a href="#">ALL_IDX_COLUMNS</a>                   | Displays index columns of all tables that the current user can access.                                                                    |
| <a href="#">ALL_IDX_COLUMNS_INFO</a>              | Displays index columns with index information of all tables that the current user can access.                                             |
| <a href="#">ALL_IDX_EXPRESSIONS</a>               | Displays expressions on function-based index columns of all tables that the current user can access.                                      |
| <a href="#">ALL_IDX_PARTITIONS</a>                | displays partitions of the indexes that the current user can access.                                                                      |
| <a href="#">ALL_IDX_SUBPARTITIONS</a>             | Displays subpartitions of the index that the current user can access.                                                                     |
| <a href="#">ALL_INDEXES</a>                       | Displays indexes of all tables that the current user can access.                                                                          |
| <a href="#">ALL_IND_PENDING_STATS</a>             | Displays indexes of all tables that the current user can access.                                                                          |
| <a href="#">ALL_JAVA_ARGUMENTS</a>                | Displays all parameters that the current user can access.                                                                                 |
| <a href="#">ALL_JAVA_CLASSES</a>                  | Displays all classes that the current user can access.                                                                                    |
| <a href="#">ALL_JAVA_FIELDS</a>                   | Displays all fields that the current user can access.                                                                                     |
| <a href="#">ALL_JAVA_METHODS</a>                  | Displays all methods that the current user can access.                                                                                    |
| <a href="#">ALL_JOBS</a>                          | Displays all jobs that the current user can access.                                                                                       |

| Static View                                | Description                                                                                                          |
|--------------------------------------------|----------------------------------------------------------------------------------------------------------------------|
| <a href="#">ALL_JOBS_WITH_NAME</a>         | Displays all jobs with job name that the current user can access.                                                    |
| <a href="#">ALL_LIBRARIES</a>              | Displays all libraries that the current user can access.                                                             |
| <a href="#">ALL_LOBS</a>                   | Displays large objects (LOB) which belong to all tables that the current user can access.                            |
| <a href="#">ALL_LOB_PARTITIONS</a>         | Displays all LOB partitions that the current user can access.                                                        |
| <a href="#">ALL_LOB_SUBPARTITIONS</a>      | Displays LOB subpartitions that the current user can access.                                                         |
| <a href="#">ALL_LOG_GROUPS</a>             | Displays the log group definitions on the tables accessible to the current user.                                     |
| <a href="#">ALL_LOG_GROUP_COLUMNS</a>      | Displays columns that are accessible to the current user and that are specified in log groups.                       |
| <a href="#">ALL_METHOD_PARAMS</a>          | Displays all the method parameters of the type that the current user can access.                                     |
| <a href="#">ALL_METHOD_RESULTS</a>         | Displays the result of the method of this type that the current user can access.                                     |
| <a href="#">ALL_MVIEWS</a>                 | Displays all materialized views that the current user can access.                                                    |
| <a href="#">ALL_MVIEW_COMMENTS</a>         | Displays comments on the materialized views that the current user can access.                                        |
| <a href="#">ALL_MVIEW_DETAIL_RELATIONS</a> | Describes all such detail relations defined for materialized views in the database that the current user can access. |
| <a href="#">ALL_MVIEW_LOGS</a>             | Displays all materialized view logs that the current user can access.                                                |
| <a href="#">ALL_MVIEW_REFRESH_TIMES</a>    | Describes refresh times of all materialized views accessible to the current user.                                    |
| <a href="#">ALL_OBJECTS</a>                | Displays all objects that the current user can access.                                                               |
| <a href="#">ALL_PACKAGE_COLL_TYPES</a>     | Displays collection types of the package that the current user can access,                                           |
| <a href="#">ALL_PACKAGE_MEMBER_TYPES</a>   | Displays attributes of the package type that the current user can access.                                            |
| <a href="#">ALL_PACKAGE_TYPES</a>          | Displays member types of the package that the current user can access                                                |
| <a href="#">ALL_PART_INDEXES</a>           | Displays partition information for the partitioned indexes that the current user can access.                         |
| <a href="#">ALL_PART_KEY_COLUMNS</a>       | Displays partitioning key columns of the partitioned tables or indexes that the current user can access.             |
| <a href="#">ALL_PART_LOBS</a>              | Displays LOBs which belong to all partitioned tables that the current user can access.                               |

| Static View                                   | Description                                                                                                                              |
|-----------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------|
| <a href="#">ALL_PART_TABLES</a>               | Displays partitioning information of all partitioned tables that the current user can access.                                            |
| <a href="#">ALL_PROCEDURES</a>                | Displays all procedures defined in all procedures and tables that the current user owns.                                                 |
| <a href="#">ALL_QUEUES</a>                    | Describes all queues on which the current user has enqueue or dequeue privileges.                                                        |
| <a href="#">ALL_QUEUE_SUBSCRIBERS</a>         | Shows the list of subscribers that the current user has privilege to dequeue from.                                                       |
| <a href="#">ALL_QUEUE_TABLES</a>              | Describes the queues in the queue tables accessible to the current user.                                                                 |
| <a href="#">ALL_RECYCLEBIN</a>                | Displays information of all temporarily deleted objects that the current user can access.                                                |
| <a href="#">ALL_SCHEDULER_CHAINS</a>          | Displays all scheduler chains that the current user can access.                                                                          |
| <a href="#">ALL_SCHEDULER_JOBS</a>            | Displays all scheduler jobs that the current user can access.                                                                            |
| <a href="#">ALL_SCHEDULER_JOB_LOG</a>         | Displays all scheduler job log that the current user can access.                                                                         |
| <a href="#">ALL_SCHEDULER_JOB_RUN_DETAILS</a> | Displays all scheduler job run details that the current user can access.                                                                 |
| <a href="#">ALL_SCHEDULER_PROGRAMS</a>        | Displays all scheduler programs that the current user can access.                                                                        |
| <a href="#">ALL_SCHEDULER_RULES</a>           | Displays all scheduler rules that the current user can access.                                                                           |
| <a href="#">ALL_SCHEDULER_RULE_CTX</a>        | Displays all scheduler rule ctx that the current user can access.                                                                        |
| <a href="#">ALL_SCHEDULER_RUNNING_JOBS</a>    | Displays all scheduler running jobs that the current user can access.                                                                    |
| <a href="#">ALL_SCHEDULER_SCHEDULES</a>       | Displays all scheduler schedules that the current user can access.                                                                       |
| <a href="#">ALL_SCHEDULER_STEPS</a>           | Displays all scheduler steps that the current user can access.                                                                           |
| <a href="#">ALL_SCHEDULER_STEP_CTX</a>        | Displays all scheduler step context the current user can access.                                                                         |
| <a href="#">ALL_SEGMENTS</a>                  | Displays all segments that the current user can access.                                                                                  |
| <a href="#">ALL_SEQUENCES</a>                 | Displays all sequences that the current user can access.                                                                                 |
| <a href="#">ALL_SOURCE</a>                    | Displays source texts of the stored objects that the current user can access such as procedures, functions, package bodies, or triggers. |
| <a href="#">ALL_STAT_EXTENSIONS</a>           | Displays information of all optimizer statistics extensions that the current user can access.                                            |
| <a href="#">ALL_SUBPARTITION_TEMPLATES</a>    | displays all subpartition templates that the current user can access.                                                                    |
| <a href="#">ALL_SUBPART_KEY_COLUMNS</a>       | Displays subpartitioning key columns of the partitioned tables or indexes that the current user can access.                              |

| Static View                                   | Description                                                                                                                               |
|-----------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------|
| <a href="#">ALL_SYNONYMS</a>                  | Displays all synonyms that the current user can access.                                                                                   |
| <a href="#">ALL_TABLES</a>                    | Displays all tables that the current user can access.                                                                                     |
| <a href="#">ALL_TAB_COMMENTS</a>              | Displays comments on the tables that the current user can access.                                                                         |
| <a href="#">ALL_TAB_HISTGRM_PENDING_STATS</a> | Displays histogram statistics of columns in pending state current user can access.                                                        |
| <a href="#">ALL_TAB_IDENTITY_COLS</a>         | Displays identity columns of all tables that the current user can access.                                                                 |
| <a href="#">ALL_TAB_PENDING_STATS</a>         | Displays pending statistics for the tables that the current user can access.                                                              |
| <a href="#">ALL_TBL_COLS</a>                  | Displays column information of all tables and views that the current user can access.                                                     |
| <a href="#">ALL_TBL_COLUMNS</a>               | Displays column information of all tables and views that the current user can access.                                                     |
| <a href="#">ALL_TBL_COLUMNS_WITH_DNO</a>      | Displays column information of all tables and views, which the current user can access, with data type number of the column.              |
| <a href="#">ALL_TBL_COL_STATISTICS</a>        | Displays statistics and a histogram of columns by extracting them from ALL_TBL_COLUMNS.                                                   |
| <a href="#">ALL_TBL_HISTOGRAMS</a>            | Displays histogram information of tables that the current user can access.                                                                |
| <a href="#">ALL_TBL_PARTITIONS</a>            | Displays partition information of all tables that the current user can access.                                                            |
| <a href="#">ALL_TBL_PRIVS</a>                 | Displays tables for which the current user or public user is granted object privileges.                                                   |
| <a href="#">ALL_TBL_PRIVS_MADE</a>            | Displays object privileges granted on tables for which the current user or public user is the object owner or grantor.                    |
| <a href="#">ALL_TBL_PRIVS_RECD</a>            | Displays object privileges granted on tables for which the current user or public user is the object grantee.                             |
| <a href="#">ALL_TBL_SUBPARTITIONS</a>         | Displays subpartitions of all tables that the current user can access.                                                                    |
| <a href="#">ALL_TRIGGERS</a>                  | Displays all triggers that the current user can access.                                                                                   |
| <a href="#">ALL_TRIGGER_COLS</a>              | Displays column information of all triggers that the current user can access and all triggers on tables that the current user can access. |
| <a href="#">ALL_TYPES</a>                     | Displays all types that the current user can access.                                                                                      |
| <a href="#">ALL_TYPE_ATTRS</a>                | Displays all the attributes that the current user can access.                                                                             |
| <a href="#">ALL_TYPE_METHODS</a>              | Displays all the methods of this type that the current user can access.                                                                   |
| <a href="#">ALL_UNUSED_COL_TABS</a>           | Displays all unused columns that the current user can access.                                                                             |

| Static View                           | Description                                                                                                                                                                               |
|---------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <a href="#">ALL_UPDATABLE_COLUMNS</a> | Displays all join view columns that the current user can update.                                                                                                                          |
| <a href="#">ALL_USERS</a>             | Displays all users that the current user can view.                                                                                                                                        |
| <a href="#">ALL_VIEWS</a>             | Displays all views that the current user can access.                                                                                                                                      |
| <a href="#">ALL_VIEWS_INTERNAL</a>    | Displays all views that the current user can access. (including views that can be accessed by the roles)                                                                                  |
| <a href="#">CTX_PREFERENCES</a>       | Displays preferences created by Tiberio users, as well as all the system-defined preferences.                                                                                             |
| <a href="#">CTX_PREFERENCE_VALUES</a> | Displays the values assigned to all the preferences in the Text data dictionary.                                                                                                          |
| <a href="#">CTX_STOPLISTS</a>         | Displays stoplists                                                                                                                                                                        |
| <a href="#">DATABASE_PROPERTIES</a>   | Displays database properties.                                                                                                                                                             |
| <a href="#">DBA_2PC_NEIGHBORS</a>     | Displays connection information for pending transactions.                                                                                                                                 |
| <a href="#">DBA_2PC_PENDING</a>       | Displays pending transaction information.                                                                                                                                                 |
| <a href="#">DBA_AUDIT_POLICIES</a>    | Displays all fine-grained auditing policies in the database.                                                                                                                              |
| <a href="#">DBA_AUDIT_TRAIL</a>       | Displays audit trails within the database.                                                                                                                                                |
| <a href="#">DBA_CATALOG</a>           | Displays all tables, views, synonyms, and sequences in the database.                                                                                                                      |
| <a href="#">DBA_COLL_TYPES</a>        | Displays all the collection types.                                                                                                                                                        |
| <a href="#">DBA_COL_COMMENTS</a>      | Displays comments on all columns in the database.                                                                                                                                         |
| <a href="#">DBA_COL_PENDING_STATS</a> | Displays statistics and a histogram of columns in pending state.                                                                                                                          |
| <a href="#">DBA_COL_PRIVS</a>         | Displays object privileges of all columns of the database.                                                                                                                                |
| <a href="#">DBA_CONSTRAINTS</a>       | Displays integrity constraints defined in the database table.                                                                                                                             |
| <a href="#">DBA_CONS_COLUMNS</a>      | Displays the columns for which the database integrity constraint is specified.                                                                                                            |
| <a href="#">DBA_DATA_FILES</a>        | Displays data file information.                                                                                                                                                           |
| <a href="#">DBA_DB_LINKS</a>          | Displays information for all database links.                                                                                                                                              |
| <a href="#">DBA_DEPENDENCIES</a>      | Displays dependencies of all objects.                                                                                                                                                     |
| <a href="#">DBA_DIRECTORIES</a>       | Displays all directories in the database.                                                                                                                                                 |
| <a href="#">DBA_ENCRYPTED_COLUMNS</a> | Displays information of all encrypted columns in the database. The columns are the same as those in ALL_ENCRYPTED_COLUMNS.                                                                |
| <a href="#">DBA_ERRORS</a>            | Displays the recent errors on every view, procedure, function, package, package body, trigger, library, and java object in the database. The columns are the same as those in ALL_ERRORS. |
| <a href="#">DBA_EXTENTS</a>           | Displays information of the extents contained in the segments of all the tablespaces in the database.                                                                                     |

| Static View                                        | Description                                                                                                                      |
|----------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------|
| <a href="#">DBA_EXTERNAL_LOCATIONS</a>             | Displays all external table locations in the database.                                                                           |
| <a href="#">DBA_EXTERNAL_TABLES</a>                | Displays all external tables in the database.                                                                                    |
| <a href="#">DBA_FGA_AUDIT_TRAIL</a>                | Displays TODO. TODO                                                                                                              |
| <a href="#">DBA_FREE_SPACE</a>                     | Displays information of the extents that all tablespaces can use.                                                                |
| <a href="#">DBA_FREE_TABLESPACE</a>                | Displays information of the total free space that all tablespaces can use.                                                       |
| <a href="#">DBA_HIST_ACTIVE_SESS_HISTORY</a>       | Displays history information on active sessions.                                                                                 |
| <a href="#">DBA_HIST_FILESTAT</a>                  | Displays historical statistics of data file and temp file I/O. This view includes snapshots of V\$ALL_FILESTAT.                  |
| <a href="#">DBA_HIST_LATCH</a>                     | Displays the historical information on spinlocks. This view includes snapshots of V\$LATCH.                                      |
| <a href="#">DBA_HIST_LIBRARYCACHE</a>              | Displays information of PP cache and DD cache. This view includes snapshots of V\$LIBRARYCACHE.                                  |
| <a href="#">DBA_HIST_OSSTAT</a>                    | Displays historical information on various OS statistics. This view includes snapshots of V\$OSSTAT2.                            |
| <a href="#">DBA_HIST_PGASTAT</a>                   | Displays the historical statistics of PGA. This view includes the snapshot of V\$PGASTAT.                                        |
| <a href="#">DBA_HIST_PRO<br/>CESS_MEM_SUMMARY</a>  | Displays historical information on PGA memory usage. This view gets information from V\$PROCESS.                                 |
| <a href="#">DBA_HIST_SEG_STAT</a>                  | Displays historical information of segment-level statistics. This view includes snapshots of V\$SEGMENT_STATISTICS.              |
| <a href="#">DBA_HIST_SGASTAT</a>                   | Displays the historical information on the shared pool for a database. This view includes snapshots of V\$SGASTAT.               |
| <a href="#">DBA_HIST_SNAPSHOT</a>                  | Displays information about TPR snapshots.                                                                                        |
| <a href="#">DBA_HIST_SQLBIND</a>                   | Displays historical information on bind variables used by SQL cursors. This view contains snapshots of V\$SQL_BIND_CAPTURE.      |
| <a href="#">DBA_HIST_SQLSTAT</a>                   | Displays historical information for SQL statistics. This view displays the priority of items of specific criteria.               |
| <a href="#">DBA_HIST_SQLTEXT</a>                   | Displays historical information about texts of SQL statements.                                                                   |
| <a href="#">DBA_HIST_SQL_PLAN</a>                  | Displays historical information about SQL execution plans. This view includes snapshots of V\$SQL_PLAN.                          |
| <a href="#">DBA_HIST_SQL_WORKAREA_HST<br/>GRAM</a> | Displays historical information about work areas used by SQL cursors. This view includes snapshots of V\$SQL_WORKAREA_HISTOGRAM. |
| <a href="#">DBA_HIST_SYSMETRIC_SUM<br/>MARY</a>    | Displays a history of statistical summary of all metric values in the System Metrics Long Duration group.                        |

| Static View                                | Description                                                                                                                                                                                                  |
|--------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <a href="#">DBA_HIST_SYSTEM_EVENT</a>      | Displays historical information of events performing in the system. This view includes snapshots of V\$SYSTEM_EVENT.                                                                                         |
| <a href="#">DBA_HIST_UNDOSTAT</a>          | Displays historical statistics of undo storage space. This view includes snapshots of V\$UNDOSTAT.                                                                                                           |
| <a href="#">DBA_HIST_WAITSTAT</a>          | Displays the statistical history of block wait events for each block type and class. This view includes snapshots of V\$WAITSTAT.                                                                            |
| <a href="#">DBA_IDX_COLUMNS</a>            | Displays index columns of all tables in the database.                                                                                                                                                        |
| <a href="#">DBA_IDX_EXPRESSIONS</a>        | Displays expressions on the index columns of all tables in the database.                                                                                                                                     |
| <a href="#">DBA_IDX_PARTITIONS</a>         | displays all index partitions in the database.                                                                                                                                                               |
| <a href="#">DBA_IDX_SUBPARTITIONS</a>      | Displays subpartitions of all indexes in the database.                                                                                                                                                       |
| <a href="#">DBA_INDEXES</a>                | Displays indexes of all tables in the database.                                                                                                                                                              |
| <a href="#">DBA_IND_PENDING_STATS</a>      | Displays indexes of all tables in the database.                                                                                                                                                              |
| <a href="#">DBA_JAVA_ARGUMENTS</a>         | Displays all parameters in the database.                                                                                                                                                                     |
| <a href="#">DBA_JAVA_CLASSES</a>           | Displays all classes in the database.                                                                                                                                                                        |
| <a href="#">DBA_JAVA_FIELDS</a>            | Displays all fields in the database.                                                                                                                                                                         |
| <a href="#">DBA_JAVA_METHODS</a>           | Displays all methods in the database.                                                                                                                                                                        |
| <a href="#">DBA_JOBS</a>                   | Displays all jobs in the database.                                                                                                                                                                           |
| <a href="#">DBA_JOBS_WITH_NAME</a>         | Displays all jobs with job name in the database.                                                                                                                                                             |
| <a href="#">DBA_LIBRARIES</a>              | Displays all libraries in the database.                                                                                                                                                                      |
| <a href="#">DBA_LOBS</a>                   | Displays LOBs which belong to all tables in the database.                                                                                                                                                    |
| <a href="#">DBA_LOB_PARTITIONS</a>         | Displays all LOB partitions in the database.                                                                                                                                                                 |
| <a href="#">DBA_LOB_SUBPARTITIONS</a>      | Displays all LOB subpartitions in the database.                                                                                                                                                              |
| <a href="#">DBA_LOG_GROUPS</a>             | Displays the log group definitions on all tables in the database.                                                                                                                                            |
| <a href="#">DBA_LOG_GROUP_COLUMNS</a>      | Displays all columns in the database that are specified in log groups.                                                                                                                                       |
| <a href="#">DBA_METHOD_PARAMS</a>          | Displays the method parameters of this type.                                                                                                                                                                 |
| <a href="#">DBA_METHOD_RESULTS</a>         | Displays the result of the method of this type.                                                                                                                                                              |
| <a href="#">DBA_MVIEWS</a>                 | Displays all materialized views in the database. The columns are the same as those in ALL_MVIEWS.                                                                                                            |
| <a href="#">DBA_MVIEW_COMMENTS</a>         | Displays comments on all materialized views in the database. The columns are the same as those in ALL_MVIEW_COMMENTS.                                                                                        |
| <a href="#">DBA_MVIEW_DETAIL_RELATIONS</a> | Describes the named detail relations that are either specified in the FROM list of the subquery that defines all materialized views in the database, or that are indirectly referenced through views in that |

| Static View                     | Description                                                                                                                   |
|---------------------------------|-------------------------------------------------------------------------------------------------------------------------------|
|                                 | FROM list. Inline views in the materialized view definition are not represented in this view or the related views.            |
| DBA_MVIEW_LOGS                  | Displays all materialized view logs in the database. The columns are the same as those in ALL_MVIEW_LOGS.                     |
| DBA_MVIEW_REFRESH_TIMES         | Describes refresh times of all materialized views in the database.                                                            |
| DBA_OBJECTS                     | Displays all objects in the database.                                                                                         |
| DBA_OBJ_AUDIT_OPTS              | Displays objects being audited in the database.                                                                               |
| DBA_PACKAGE_COLL_TYPES          | Displays types of the package.                                                                                                |
| DBA_PACKAGE_MEM<br>BER_TYPES    | Displays types of the package.                                                                                                |
| DBA_PACKAGE_TYPES               | Displays types of the package.                                                                                                |
| DBA_PARALLEL_EXE<br>CUTE_CHUNKS | Displays the chunks for all tasks in the database.                                                                            |
| DBA_PARALLEL_EXE<br>CUTE_TASKS  | Displays all tasks in the database.                                                                                           |
| DBA_PART_INDEXES                | Displays partitioning information for the partitioned indexes in the database.                                                |
| DBA_PART_KEY_COLUMNS            | Displays partitioning key columns for all partitioned tables or indexes in the database.                                      |
| DBA_PART_LOBS                   | Displays LOBs which belong to the partitioned tables in the database.                                                         |
| DBA_PART_TABLES                 | Displays partitioning information of the partitioned tables in the database.                                                  |
| DBA_PRIV_AUDIT_OPTS             | Displays users being audited in the database.                                                                                 |
| DBA_PROCEDURES                  | Displays all procedures in the database.                                                                                      |
| DBA_PROFILES                    | Describes all profiles in the database.                                                                                       |
| DBA_QUEUES                      | Describes all queues in the database.                                                                                         |
| DBA_QUEUE_SUBSCRIBERS           | Shows the list of subscribers on all queues in the database.                                                                  |
| DBA_QUEUE_TABLES                | Describes the queues in all queue tables in the database.                                                                     |
| DBA_RECYCLEBIN                  | Displays information of all temporarily deleted objects in the database. The columns are the same as those in ALL_RECYCLEBIN. |
| DBA_REDEFINITION_OBJECTS        | Displays all objects in database which are being redefined online.                                                            |
| DBA_ROLES                       | Displays all roles in the database.                                                                                           |
| DBA_ROLE_PRIVS                  | Displays roles granted to the users in the database.                                                                          |
| DBA_SCHEDULER_CHAINS            | Displays all scheduler chains in the database.                                                                                |

| Static View                                   | Description                                                                                                             |
|-----------------------------------------------|-------------------------------------------------------------------------------------------------------------------------|
| <a href="#">DBA_SCHEDULER_JOBS</a>            | Displays all scheduler jobs in the database.                                                                            |
| <a href="#">DBA_SCHEDULER_JOB_LOG</a>         | Displays all scheduler job log in the database.                                                                         |
| <a href="#">DBA_SCHEDULER_JOB_RUN_DETAILS</a> | Displays all scheduler job run details in the database.                                                                 |
| <a href="#">DBA_SCHEDULER_PROGRAMS</a>        | Displays all scheduler programs in the database.                                                                        |
| <a href="#">DBA_SCHEDULER_RULES</a>           | Displays all scheduler rules in the database.                                                                           |
| <a href="#">DBA_SCHEDULER_RULE_CTX</a>        | Displays all scheduler rule ctx in the database.                                                                        |
| Static View                                   | Description                                                                                                             |
| <a href="#">DBA_SCHEDULER_RUNNING_JOBS</a>    | Displays all scheduler running jobs in the database.                                                                    |
| <a href="#">DBA_SCHEDULER_SCHEDULES</a>       | Displays all scheduler schedules in the database.                                                                       |
| <a href="#">DBA_SCHEDULER_STEPS</a>           | Displays all scheduler steps in the database.                                                                           |
| <a href="#">DBA_SCHEDULER_STEP_CTX</a>        | Displays all scheduler step context in the database.                                                                    |
| <a href="#">DBA_SEGMENTS</a>                  | Displays all segments in the database.                                                                                  |
| <a href="#">DBA_SEQUENCES</a>                 | Displays all sequences in the database.                                                                                 |
| <a href="#">DBA_SOURCE</a>                    | Displays source texts of the stored objects in the database such as procedures, functions, package bodies, or triggers. |
| <a href="#">DBA_SQL_PLAN_BASELINES</a>        | Displays information about the created SQL plan baselines                                                               |
| <a href="#">DBA_SQL_PROFILES</a>              | displays information about SQL profiles currently created for specific SQL statements.                                  |
| <a href="#">DBA_STAT_EXTENSIONS</a>           | displays information about all optimizer statistics extensions in the database.                                         |
| <a href="#">DBA_SUBPARTITION_TEMPLATES</a>    | displays all subpartition templates in the database.                                                                    |
| <a href="#">DBA_SUBPART_KEY_COLUMNS</a>       | Displays subpartitioning key columns of the partitioned tables or in dexes in the database.                             |
| <a href="#">DBA_SYNONYMS</a>                  | Displays synonyms in the database.                                                                                      |
| <a href="#">DBA_SYS_PRIVS</a>                 | Displays system privileges granted to users and roles in the database.                                                  |
| <a href="#">DBA_TABLES</a>                    | Displays all tables in the database.                                                                                    |
| <a href="#">DBA_TABLESPACES</a>               | Displays all tablespaces in the database.                                                                               |
| <a href="#">DBA_TAB_COMMENTS</a>              | Displays comments on all tables and views in the database.                                                              |

| Static View                                   | Description                                                                                                         |
|-----------------------------------------------|---------------------------------------------------------------------------------------------------------------------|
| <a href="#">DBA_TAB_HISTGRM_PENDING_STATS</a> | Displays histogram statistics of columns in pending state.                                                          |
| <a href="#">DBA_TAB_IDENTITY_COLS</a>         | Displays identity columns of all tables.                                                                            |
| <a href="#">DBA_TAB_PENDING_STATS</a>         | Displays pending statistics for the tables in the database.                                                         |
| <a href="#">DBA_TBL_COLS</a>                  | Displays columns of all tables and views in the database.                                                           |
| <a href="#">DBA_TBL_COLUMNS</a>               | Displays columns of all tables and views in the database.                                                           |
| <a href="#">DBA_TBL_COL_STATISTICS</a>        | Displays statistics and a histogram of columns by extracting them from DBA_TBL_COLUMNS.                             |
| <a href="#">DBA_TBL_HISTOGRAMS</a>            | Displays histogram information on tables and views in the database.                                                 |
| <a href="#">DBA_TBL_PARTITIONS</a>            | Displays partition information of all tables in the database.                                                       |
| <a href="#">DBA_TBL_PRIVS</a>                 | Displays object privileges granted on all tables in the database.                                                   |
| <a href="#">DBA_TBL_SUBPARTITIONS</a>         | Displays subpartitions of all tables in the database.                                                               |
| <a href="#">DBA_TEMP_FILES</a>                | Displays information of the temporary data files.                                                                   |
| <a href="#">DBA_TRIGGERS</a>                  | Displays all triggers in the database.                                                                              |
| <a href="#">DBA_TRIGGER_COLS</a>              | Displays column information of all triggers in the database. The columns are the same as those in ALL_TRIGGER_COLS. |
| <a href="#">DBA_TS_QUOTAS</a>                 | Describe tablespace quotas for all user                                                                             |
| <a href="#">DBA_TYPES</a>                     | Displays all types.                                                                                                 |
| <a href="#">DBA_TYPE_ATTRS</a>                | Displays all the attributes of this type.                                                                           |
| <a href="#">DBA_TYPE_METHODS</a>              | Displays all the methods of this type.                                                                              |
| <a href="#">DBA_UNDO_EXTENTS</a>              | Displays information about the extents in all undo segments.                                                        |
| <a href="#">DBA_UNDO_SGMTS</a>                | Displays undo segments in the database.                                                                             |
| <a href="#">DBA_UNUSED_COL_TABS</a>           | Displays all unused columns in the database.                                                                        |
| <a href="#">DBA_UPDATABLE_COLUMNS</a>         | Displays all join view columns that the DBA can update.                                                             |
| <a href="#">DBA_USERS</a>                     | Displays all users in the database.                                                                                 |
| <a href="#">DBA_VIEWS</a>                     | Displays all views in the database.                                                                                 |
| <a href="#">DBMSHXP_SQL_PROFILE_ATTR</a>      | displays outline hints that sql profile defines.                                                                    |
| <a href="#">DICTIONARY</a>                    | Displays comments on tables and views in the data dictionary.                                                       |
| <a href="#">NLS_DATABASE_PARAMETERS</a>       | Displays NLS parameters of the current database.                                                                    |
| <a href="#">NLS_SESSION_PARAMETERS</a>        | Displays NLS parameters of the current user session.                                                                |
| <a href="#">PUBLICSYN</a>                     | Displays all public synonyms in the database.                                                                       |
| <a href="#">REDACTION_COLUMNS</a>             | Describes all redacted columns in the database                                                                      |

| Static View                     | Description                                                                                                     |
|---------------------------------|-----------------------------------------------------------------------------------------------------------------|
| REDACTION_COLUMNS_DBMS_ERRORLOG | Logs DML errors that occur on columns protected by data redaction policies (Currently not supported)            |
| REDACTION_POLICIES              | Displays all redaction policies in the database.                                                                |
| REDACTION_VALUES_FOR_TYPE_FULL  | Shows all of the current values for full redaction.                                                             |
| ROLE_ROLE_PRIVS                 | Displays the roles granted to other roles which the current user can access.                                    |
| ROLE_SYS_PRIVS                  | Displays system privileges granted to roles which the current user can access.                                  |
| ROLE_TAB_PRIVS                  | Displays object privileges granted to roles which the current user can access.                                  |
| SESSION_CONTEXT                 | Displays context attributes in the current session.                                                             |
| SESSION_PRIVS                   | Displays the system privileges granted to current session user.                                                 |
| SESSION_ROLES                   | Displays roles granted to the current session user.                                                             |
| SYS_OBJECTS                     | Displays IDs and types of all schema objects, and IDs of the segment header blocks in the database.             |
| USER_ARGUMENTS                  | Displays all parameters of the procedure or function that the current user owns.                                |
| USER_AUDIT_TRAIL                | Displays audit trails associated with the current user.                                                         |
| USER_CATALOG                    | Displays all tables, views, synonyms, and sequences that the current user owns.                                 |
| USER_COLL_TYPES                 | Displays all the collection types that the current user owns.                                                   |
| USER_COL_COMMENTS               | Displays comments on all columns in the tables or views that the current user owns.                             |
| USER_COL_PENDING_STATS          | Displays statistics and a histogram of columns in pending state current user owns.                              |
| USER_COL_PRIVS                  | Displays object privileges of the columns for which the current user is the object owner, grantor, or grantee.  |
| USER_COL_PRIVS_MADE             | Displays columns for which the current user is the grantor.                                                     |
| USER_COL_PRIVS_RECD             | Displays columns for which the current user is the grantee.                                                     |
| USER_CONSTRAINTS                | Displays constraints defined in the table that the current user owns.                                           |
| USER_CONS_COLUMNS               | Displays columns that are owned by the current user and that are specified in integrity constraint definitions. |

| Static View                             | Description                                                                                                                                                                                                       |
|-----------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <a href="#">USER_DB_LINKS</a>           | Displays information for the database links that the current user owns.                                                                                                                                           |
| <a href="#">USER_DEPENDENCIES</a>       | Displays dependencies between procedures, functions, packages, package bodies, triggers, and views that the current user owns.                                                                                    |
| <a href="#">USER_ENCRYPTED_COLUMNS</a>  | Displays information of all encrypted columns that the current user owns. The columns are the same as those in ALL_ENCRYPTED_COLUMNS(except for OWNER)                                                            |
| <a href="#">USER_ERRORS</a>             | Displays the recent errors in every view, procedure, function, package, package body, trigger, library, and java object owned by current user. The columns are the same as those in ALL_ERRORS.(except for OWNER) |
| <a href="#">USER_EXTENTS</a>            | Displays information of the extents contained in the segments that the current user owns.                                                                                                                         |
| <a href="#">USER_EXTERNAL_LOCATIONS</a> | Displays all external table locations that the current user owns.                                                                                                                                                 |
| <a href="#">USER_EXTERNAL_TABLES</a>    | Displays all external tables that the current user owns.                                                                                                                                                          |
| <a href="#">USER_FREE_SPACE</a>         | Displays information of the extents available for the tablespaces that the current user can access.                                                                                                               |
| <a href="#">USER_IDX_COLUMNS</a>        | Displays index columns of all tables that the current user owns.                                                                                                                                                  |
| <a href="#">USER_IDX_EXPRESSIONS</a>    | Displays expressions on function-based index columns of all tables that the current user owns.                                                                                                                    |
| <a href="#">USER_IDX_PARTITIONS</a>     | displays partitions of the indexes that the current user owns.                                                                                                                                                    |
| <a href="#">USER_INDEXES</a>            | Displays indexes of all tables that the current user owns.                                                                                                                                                        |
| <a href="#">USER_IND_PENDING_STATS</a>  | Displays indexes of all tables that the current user owns.                                                                                                                                                        |
| <a href="#">USER_JAVA_ARGUMENTS</a>     | Displays all parameters that the current user owns.                                                                                                                                                               |
| <a href="#">USER_JAVA_CLASSES</a>       | Displays all classes that the current user owns.                                                                                                                                                                  |
| <a href="#">USER_JAVA_FIELDS</a>        | Displays all fields that the current user owns.                                                                                                                                                                   |
| <a href="#">USER_JAVA_METHODS</a>       | Displays all methods that the current user owns.                                                                                                                                                                  |
| <a href="#">USER_JOBS</a>               | Displays all jobs that the current user owns.                                                                                                                                                                     |
| <a href="#">USER_JOBS_WITH_NAME</a>     | Displays all jobs with job name that the current user owns.                                                                                                                                                       |
| <a href="#">USER_LIBRARIES</a>          | Displays all libraries that the current user owns.                                                                                                                                                                |
| <a href="#">USER_LOBS</a>               | Displays large objects (LOB) which belong to all tables that the current user owns.                                                                                                                               |
| <a href="#">USER_LOB_PARTITIONS</a>     | Displays all LOB partitions which belong to all tables that the current user owns.                                                                                                                                |

| Static View                                  | Description                                                                                                          |
|----------------------------------------------|----------------------------------------------------------------------------------------------------------------------|
| <a href="#">USER_LOB_SUBPARTITIONS</a>       | Displays all LOB subpartitions which belong to all tables that the current user owns.                                |
| <a href="#">USER_LOG_GROUPS</a>              | Displays the log group definitions on the tables owned by the current user.                                          |
| <a href="#">USER_LOG_GROUP_COLUMNS</a>       | Displays columns that are owned by the current user and that are specified in log groups.                            |
| <a href="#">USER_METHOD_PARAMS</a>           | Displays the method parameters of the type that the current user owns.                                               |
| <a href="#">USER_METHOD_RESULTS</a>          | Displays the result of the method of this type that the current user owns.                                           |
| <a href="#">USER_MVIEWS</a>                  | Displays all materialized views that the current user owns. The columns are the same as those in ALL_MVIEWS.         |
| <a href="#">USER_MVIEW_COMMENTS</a>          | Displays comments on all materialized views that the current user owns.                                              |
| <a href="#">USER_MVIEW_DETAIL_RELATIONS</a>  | Describes all such detail relations defined for materialized views in the database owned by the current user.        |
| <a href="#">USER_MVIEW_LOGS</a>              | Displays all materialized view logs that the current user owns. The columns are the same as those in ALL_MVIEW_LOGS. |
| <a href="#">USER_MVIEW_REFRESH_TIMES</a>     | Describes refresh times of the materialized views owned by the current user.                                         |
| <a href="#">USER_OBJECTS</a>                 | Displays all objects that the current user owns.                                                                     |
| <a href="#">USER_OBJ_AUDIT_OPTS</a>          | Displays the current user's own objects which are being audited.                                                     |
| <a href="#">USER_PACKAGE_COLL_TYPES</a>      | Displays types of the package that the current user owns.                                                            |
| <a href="#">USER_PACKAGE_MEMBER_TYPES</a>    | Displays types of the package that the current user owns.                                                            |
| <a href="#">USER_PACKAGE_TYPES</a>           | Displays types of the package that the current user owns.                                                            |
| <a href="#">USER_PARALLEL_EXECUTE_CHUNKS</a> | Displays the chunks for all tasks that created by current user.                                                      |
| <a href="#">USER_PARALLEL_EXECUTE_TASKS</a>  | Displays all tasks that created by current user.                                                                     |
| <a href="#">USER_PART_INDEXES</a>            | Displays partitioning information for the partitioned indexes that the current user owns.                            |
| <a href="#">USER_PART_KEY_COLUMNS</a>        | Displays partitioning key columns for all partitioned tables or indexes that the current user owns.                  |
| <a href="#">USER_PART_LOBS</a>               | Displays LOBs which belong to the partitioned tables that the current user owns.                                     |

| Static View                                    | Description                                                                                                                              |
|------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------|
| <a href="#">USER_PART_TABLES</a>               | Displays partitioning information of the partitioned tables that the current user owns.                                                  |
| <a href="#">USER_PROCEDURES</a>                | Displays all procedures that the current user owns.                                                                                      |
| <a href="#">USER_QUEUES</a>                    | Describes the operational characteristics of every queue owned by the current user.                                                      |
| <a href="#">USER_QUEUE_SUBSCRIBERS</a>         | Shows the list of subscribers on queues that are under the current user's schema.                                                        |
| <a href="#">USER_QUEUE_TABLES</a>              | Describes the queues in the queue tables created in the current user's schema.                                                           |
| <a href="#">USER_RECYCLEBIN</a>                | Displays information of all temporarily deleted objects that the current user owns. The columns are the same as those in ALL_RECYCLEBIN. |
| <a href="#">USER_ROLE_PRIVS</a>                | Displays roles granted to the current user.                                                                                              |
| <a href="#">USER_SCHEDULER_CHAINS</a>          | Displays all scheduler chains that the current user owns.                                                                                |
| <a href="#">USER_SCHEDULER_JOBS</a>            | Displays all scheduler jobs that the current user owns.                                                                                  |
| <a href="#">USER_SCHEDULER_JOB_LOG</a>         | Displays all scheduler job log that the current user owns.                                                                               |
| <a href="#">USER_SCHEDULER_JOB_RUN_DETAILS</a> | Displays all scheduler job run details that the current user owns.                                                                       |
| <a href="#">USER_SCHEDULER_PROGRAMS</a>        | Displays all scheduler programs that the current user owns.                                                                              |
| <a href="#">USER_SCHEDULER_RULES</a>           | Displays all scheduler rules that the current user owns.                                                                                 |
| <a href="#">USER_SCHEDULER_RULE_CTX</a>        | Displays all scheduler rule ctx that the current user owns.                                                                              |
| <a href="#">USER_SCHEDULER_RUNNING_JOBS</a>    | Displays all scheduler running jobs that the current user owns.                                                                          |
| <a href="#">USER_SCHEDULER_SCHEDULES</a>       | Displays all scheduler schedules that the current user owns.                                                                             |
| <a href="#">USER_SCHEDULER_STEPS</a>           | Displays all scheduler steps that the current user owns.                                                                                 |
| <a href="#">USER_SCHEDULER_STEP_CTX</a>        | Displays all scheduler step context the current user owns.                                                                               |
| <a href="#">USER_SEGMENTS</a>                  | Displays all segments which belong to the current user.                                                                                  |
| <a href="#">USER_SEQUENCES</a>                 | Displays all sequences that the current user owns.                                                                                       |
| <a href="#">USER_SOURCE</a>                    | Displays source texts of the stored objects in the database such as procedures, functions, package bodies, or triggers.                  |
| <a href="#">USER_STAT_EXTENSIONS</a>           | Displays information of all optimizer statistics extensions that the current user owns                                                   |

| Static View                                    | Description                                                                                                                                                                         |
|------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <a href="#">USER_SUBPARTITION_TEMPLATES</a>    | displays subaprtition templates that the current user owns.                                                                                                                         |
| <a href="#">USER_SUBPART_KEY_COLUMNS</a>       | Displays subpartitioning key columns of the partitioned tables or in dexes that the current user owns.                                                                              |
| <a href="#">USER_SYNONYMS</a>                  | Displays synonyms that the current user owns.                                                                                                                                       |
| <a href="#">USER_SYS_PRIVS</a>                 | Displays system privileges granted to the current user.                                                                                                                             |
| <a href="#">USER_TABLES</a>                    | Displays all tables that the current user owns.                                                                                                                                     |
| <a href="#">USER_TABLESPACES</a>               | Synonym for DBA_TABLESPACES.                                                                                                                                                        |
| <a href="#">USER_TAB_COMMENTS</a>              | Displays comments on all tables and views that the current user owns.                                                                                                               |
| <a href="#">USER_TAB_HISTGRM_PENDING_STATS</a> | Displays histogram statistics of columns in pending state current user owns.                                                                                                        |
| <a href="#">USER_TAB_IDENTITY_COLS</a>         | Displays identity columns of all tables that the current user owns.                                                                                                                 |
| <a href="#">USER_TAB_PENDING_STATS</a>         | Displays pending statistics for the tables that the current user owns.                                                                                                              |
| <a href="#">USER_TBL_COLS</a>                  | Displays columns of all tables and views that the current user owns.                                                                                                                |
| <a href="#">USER_TBL_COLUMNS</a>               | Displays columns of all tables and views that the current user owns.                                                                                                                |
| <a href="#">USER_TBL_COL_STATISTICS</a>        | Displays statistics and a histogram of columns by extracting them from USER_TBL_COLUMNS.                                                                                            |
| <a href="#">USER_TBL_HISTOGRAMS</a>            | Displays histogram information of tables that the current user owns.                                                                                                                |
| <a href="#">USER_TBL_PARTITIONS</a>            | Displays partition information of all tables that the current user owns.                                                                                                            |
| <a href="#">USER_TBL_PRIVS</a>                 | Displays object privileges granted on all tables for which the current user is the object owner, grantor, or grantee.                                                               |
| <a href="#">USER_TBL_PRIVS_MADE</a>            | Displays object privileges granted on tables for which the current user is the object owner.                                                                                        |
| <a href="#">USER_TBL_PRIVS_REC'D</a>           | Displays object privileges granted on tables for which the current user is the object grantee.                                                                                      |
| <a href="#">USER_TRIGGERS</a>                  | Displays all triggers that the current user owns.                                                                                                                                   |
| <a href="#">USER_TRIGGER_COLS</a>              | Displays column information of all triggers that the current user owns and all triggers on tables that the current user owns The columns are the same as those in ALL_TRIGGER_COLS. |
| <a href="#">USER_TYPES</a>                     | Displays all types that the current user owns.                                                                                                                                      |
| <a href="#">USER_TYPE_ATTRS</a>                | Displays all the attributes of this type that the current user owns.                                                                                                                |
| <a href="#">USER_TYPE_METHODS</a>              | Displays all the methods of this type that the current user owns.                                                                                                                   |
| <a href="#">USER_UNUSED_COL_TABS</a>           | Displays all unused columns that the current user owns.                                                                                                                             |
| <a href="#">USER_UPDATABLE_COLUMNS</a>         | Displays all join view columns that the current user can update.                                                                                                                    |

| Static View                | Description                                    |
|----------------------------|------------------------------------------------|
| <a href="#">USER_USERS</a> | Displays a description on the current user.    |
| <a href="#">USER_VIEWS</a> | Displays all views that the current user owns. |

### 3.2.1. ALL\_ARGUMENTS

Displays all parameters of the procedure or function that the current user can call.

- Columns

| Column Name    | Data Type      | Description                                                 |
|----------------|----------------|-------------------------------------------------------------|
| OWNER          | VARCHAR(128)   | Owner of the procedure or function                          |
| OBJECT_NAME    | VARCHAR(128)   | Name of the procedure or function                           |
| PACKAGE_NAME   | VARCHAR(128)   | Name of the package                                         |
| OBJECT_ID      | NUMBER         | ID of the package which contains the procedure or function  |
| ARGUMENT_NAME  | VARCHAR(128)   | Name of the parameter                                       |
| OVERLOAD       | NUMBER         | Identifier of the overloaded function                       |
| POSITION       | NUMBER         | Position of the parameter (NULL for function return values) |
| DATA_LEVEL     | NUMBER         | Depth of the parameter for composite data type.             |
| DATA_TYPE      | VARCHAR(65532) | Data type of the parameter                                  |
| DATA_TYPE_NO   | NUMBER         | Data type number of the parameter                           |
| DEFAULTED      | VARCHAR(1)     | Default expression (Y/N)                                    |
| IN_OUT         | VARCHAR(9)     | Direction of the parameter<br>- IN<br>- OUT<br>- IN/OUT     |
| DATA_LENGTH    | NUMBER         | Length of the column                                        |
| DATA_PRECISION | NUMBER         | Precision of the column                                     |

| Column Name  | Data Type    | Description                                                               |
|--------------|--------------|---------------------------------------------------------------------------|
| DATA_SCALE   | NUMBER       | Scale of the column                                                       |
| NULLABLE     | VARCHAR(3)   | Whether NULL values are allowed or not                                    |
| TYPE_OWNER   | VARCHAR(128) | Owner of the parameter type                                               |
| TYPE_NAME    | VARCHAR(128) | Name of the parameter type (Package name for the package local type)      |
| TYPE_SUBNAME | VARCHAR(128) | Name of the type defined in the package (Only for the package local type) |
| TYPE_LINK    | VARCHAR(128) | Link of the type if it is remote. (Not supported)                         |

- See also

[USER\\_ARGUMENTS](#)

### 3.2.2. ALL\_CATALOG

Displays all tables, views, synonyms, and sequences that the current user can access.

- Columns

| Column Name | Data Type    | Description                                                        |
|-------------|--------------|--------------------------------------------------------------------|
| OWNER       | VARCHAR(128) | Owner of the object                                                |
| OBJECT_NAME | VARCHAR(128) | Name of the object                                                 |
| TABLE_NAME  | VARCHAR(128) | Name of the object                                                 |
| OBJECT_TYPE | VARCHAR(23)  | Type of the object<br>- TABLE<br>- VIEW<br>- SYNONYM<br>- SEQUENCE |

- See also

[DBA\\_CATALOG](#)

### 3.2.3. ALL\_COLL\_TYPES

Displays all the collection types that the current user can access.

- Columns

| Column Name        | Data Type      | Description                                           |
|--------------------|----------------|-------------------------------------------------------|
| OWNER              | VARCHAR(128)   | Owner of this type                                    |
| TYPE_NAME          | VARCHAR(128)   | Name of this type                                     |
| COLL_TYPE          | VARCHAR(13)    | Type of this collection type                          |
| UPPER_BOUND        | NUMBER         | Maximum array size (Only applicable to VARYING ARRAY) |
| ELEM_TYPE_MOD      | VARCHAR(7)     | Modifier of this element type (not supported)         |
| ELEM_TYPE_OWNER    | VARCHAR(128)   | Owner of this element type                            |
| ELEM_TYPE_NAME     | VARCHAR(65532) | Name of this element type                             |
| LENGTH             | NUMBER         | Length of this element type (not supported)           |
| PRECISION          | NUMBER         | Precision of this element type                        |
| SCALE              | NUMBER         | Scale of this element type                            |
| CHARACTER_SET_NAME | VARCHAR(128)   | Name of the Character set (not supported)             |
| ELEM_STORAGE       | NUMBER         | (not supported)                                       |
| NULLS_STORED       | VARCHAR(3)     | (not supported)                                       |

- See also

[DBA\\_COLL\\_TYPES](#)

[USER\\_COLL\\_TYPES](#)

### 3.2.4. ALL\_COL\_COMMENTS

Displays comments on columns of the tables or views that the current user can access.

- Columns

| Column Name | Data Type     | Description           |
|-------------|---------------|-----------------------|
| OWNER       | VARCHAR(128)  | Owner of the object   |
| TABLE_NAME  | VARCHAR(128)  | Name of the object    |
| COLUMN_NAME | VARCHAR(128)  | Name of the column    |
| COMMENTS    | VARCHAR(4000) | Comment of the column |

- See also

[DBA\\_COL\\_COMMENTS](#)

[USER\\_COL\\_COMMENTS](#)

### 3.2.5. ALL\_COL\_PENDING\_STATS

Displays statistics and a histogram of columns in pending state current user can access.

- Columns

| Column Name       | Data Type     | Description                     |
|-------------------|---------------|---------------------------------|
| OWNER             | VARCHAR(128)  | Owner of the table              |
| TABLE_NAME        | VARCHAR(128)  | Name of the table               |
| PARTITION_NAME    | VARCHAR(128)  | Name of the partition           |
| SUBPARTITION_NAME | VARCHAR(128)  | Name of the subpartition        |
| COLUMN_NAME       | VARCHAR(128)  | Name of the column              |
| NUM_DISTINCT      | NUMBER        | The number of distinct values   |
| LOW_VALUE         | VARCHAR(4000) | The lowest value in the column  |
| HIGH_VALUE        | VARCHAR(4000) | The highest value in the column |
| DENSITY           | NUMBER        | Density of the column           |

| Column Name   | Data Type | Description                                                     |
|---------------|-----------|-----------------------------------------------------------------|
| NUM_NULLS     | NUMBER    | The number of NULL values                                       |
| AVG_COL_LEN   | NUMBER    | Average length of the column (in bytes)                         |
| SAMPLE_SIZE   | NUMBER    | Size of the sample used for analyzing statistics for the column |
| LAST_ANALYZED | DATE      | Date of the most recent analysis for the column                 |

- See also

[DBA\\_COL\\_PENDING\\_STATS](#)

[USER\\_COL\\_PENDING\\_STATS](#)

### 3.2.6. ALL\_COL\_PRIVS

Displays columns for which the current user or public user is the object owner or grantor (user who granted the privilege).

- Columns

| Column Name | Data Type    | Description                                                             |
|-------------|--------------|-------------------------------------------------------------------------|
| GRANTEE     | VARCHAR(128) | Name of the user to whom the privilege was granted                      |
| OWNER       | VARCHAR(128) | Owner of the object                                                     |
| TABLE_NAME  | VARCHAR(128) | Name of the object                                                      |
| COLUMN_NAME | VARCHAR(128) | Name of the column                                                      |
| GRANTOR     | VARCHAR(128) | Name of the user who granted the privilege                              |
| PRIVILEGE   | VARCHAR(40)  | Name of the privilege to the column                                     |
| GRANTABLE   | VARCHAR(3)   | Whether privilege was granted with ADMIN OPTION or not<br>- YES<br>- NO |

- See also

[DBA\\_COL\\_PRIVS](#)

[USER\\_COL\\_PRIVS](#)

### 3.2.7. ALL\_COL\_PRIVS\_MADE

Displays columns for which the current user is the object owner or grantor.

- Columns

| Column Name | Data Type    | Description                                                                     |
|-------------|--------------|---------------------------------------------------------------------------------|
| GRANTEE     | VARCHAR(128) | Name of the user to whom the privilege was granted                              |
| OWNER       | VARCHAR(128) | Owner of the object                                                             |
| TABLE_NAME  | VARCHAR(128) | Name of the object                                                              |
| COLUMN_NAME | VARCHAR(128) | Name of the column                                                              |
| GRANTOR     | VARCHAR(128) | Name of the user who granted the privilege                                      |
| PRIVILEGE   | VARCHAR(40)  | Name of the privilege to the column                                             |
| GRANTABLE   | VARCHAR(3)   | Whether privilege was granted with ADMIN OPTION or not<br><br>- YES<br><br>- NO |

- See also

[USER\\_COL\\_PRIVS\\_MADE](#)

### 3.2.8. ALL\_COL\_PRIVS\_RECD

Displays columns for which the current user or public user is the grantee.

- Columns

| Column Name | Data Type    | Description                                        |
|-------------|--------------|----------------------------------------------------|
| GRANTEE     | VARCHAR(128) | Name of the user to whom the privilege was granted |
| OWNER       | VARCHAR(128) | Owner of the object                                |

| Column Name | Data Type    | Description                                                                     |
|-------------|--------------|---------------------------------------------------------------------------------|
| TABLE_NAME  | VARCHAR(128) | Name of the object                                                              |
| COLUMN_NAME | VARCHAR(128) | Name of the column                                                              |
| GRANTOR     | VARCHAR(128) | Name of the user who granted the privilege                                      |
| PRIVILEGE   | VARCHAR(40)  | Name of the privilege to the column                                             |
| GRANTABLE   | VARCHAR(3)   | Whether privilege was granted with ADMIN OPTION or not<br><br>- YES<br><br>- NO |

- See also

[USER\\_COL\\_PRIVS\\_RECD](#)

### 3.2.9. ALL\_CONSTRAINTS

Displays constraints defined in the table that the current user can access.

- Columns

| Column Name     | Data Type    | Description                                                                                                                                                                                                                                                                                                               |
|-----------------|--------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| OWNER           | VARCHAR(128) | Owner of the constraint                                                                                                                                                                                                                                                                                                   |
| CONSTRAINT_NAME | VARCHAR(128) | Name of the constraint                                                                                                                                                                                                                                                                                                    |
| CON_TYPE        | VARCHAR(22)  | Type of the constraint<br><br>- CHECK: Check constraint<br><br>- PRIMARY KEY: Primary key constraint<br><br>- UNIQUE: Unique key constraint<br><br>- REFERENTIAL: Referential constraint<br><br>- CHECK VIEW: Check view constraint<br><br>- READ ONLY, VIEW: Read only constraint<br><br>- NOT NULL: Not null constraint |

| Column Name       | Data Type      | Description                                                                                                                                                                                                                                           |
|-------------------|----------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                   |                | - ERR: Error constraint                                                                                                                                                                                                                               |
| CONSTRAINT_TYPE   | VARCHAR(1)     | Type of the constraint<br><br>- C: CHECK, NOT NULL constraint<br><br>- P: PRIMARY KEY constraint<br><br>- U: UNIQUE constraint<br><br>- R: REFERENTIAL constraint<br><br>- O: READ ONLY VIEW constraint<br><br>- V: VIEW WITH CHECK OPTION constraint |
| TABLE_NAME        | VARCHAR(128)   | Name of the table or view for which constraints are defined                                                                                                                                                                                           |
| SEARCH_CONDITION  | VARCHAR(65532) | Search condition for CHECK constraint                                                                                                                                                                                                                 |
| R_OWNER           | VARCHAR(128)   | Owner of the table that is referred to by referential constraint                                                                                                                                                                                      |
| R_CONSTRAINT_NAME | VARCHAR(128)   | Name of the primary key or unique key constraint defined in the table to be referred                                                                                                                                                                  |
| DELETE_RULE       | VARCHAR(9)     | Delete rule for the referential constraint<br><br>- CASCADE<br><br>- SET NULL<br><br>- NO ACTION                                                                                                                                                      |
| GENERATED         | VARCHAR(14)    | Whether the name of the constraint indicates the user or the generated system                                                                                                                                                                         |
| STATUS            | VARCHAR(8)     | Current status<br><br>- ENABLED<br><br>- DISABLED                                                                                                                                                                                                     |
| INDEX_OWNER       | VARCHAR(128)   | Owner of the index used by the constraint                                                                                                                                                                                                             |
| INDEX_NAME        | VARCHAR(128)   | Name of the index used by the constraint                                                                                                                                                                                                              |

- See also

[DBA\\_CONSTRAINTS](#)

[USER\\_CONSTRAINTS](#)

### 3.2.10. ALL\_CONS\_COLUMNS

Displays columns that the current user can access and that are specified in integrity constraint definitions.

- Columns

| Column Name     | Data Type    | Description                                           |
|-----------------|--------------|-------------------------------------------------------|
| OWNER           | VARCHAR(128) | Owner of the constraint                               |
| CONSTRAINT_NAME | VARCHAR(128) | Name of the constraint                                |
| TABLE_NAME      | VARCHAR(128) | Name of the table for which the constraint is defined |
| COLUMN_NAME     | VARCHAR(128) | Name of the column defined in constraint definition   |
| POSITION        | NUMBER       | Position of the column                                |

- See also

[DBA\\_CONS\\_COLUMNS](#)

[USER\\_CONS\\_COLUMNS](#)

### 3.2.11. ALL\_CONS\_COLUMNS\_INFO

Displays columns with constraint information that the current user can access and that are specified in integrity constraint definitions.

- Columns

| Column Name     | Data Type    | Description                                           |
|-----------------|--------------|-------------------------------------------------------|
| OWNER           | VARCHAR(128) | Owner of the constraint                               |
| CON_ID          | VARCHAR(128) | Constraint id                                         |
| CONSTRAINT_NAME | VARCHAR(128) | Name of the constraint                                |
| TABLE_NAME      | VARCHAR(128) | Name of the table for which the constraint is defined |

| Column Name | Data Type    | Description                                                                                                                                                                                                                                                                                                              |
|-------------|--------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| COLUMN_NAME | VARCHAR(128) | Name of the column defined in constraint definition                                                                                                                                                                                                                                                                      |
| POSITION    | NUMBER       | Position of the column                                                                                                                                                                                                                                                                                                   |
| CON_TYPE    | VARCHAR(22)  | Type of the constraint<br>- CHECK: Check constraint<br>- PRIMARY KEY: Primary key constraint<br>- UNIQUE: Unique key constraint<br>- REFERENTIAL: Referential constraint<br>- CHECK VIEW: Check view constraint<br>- READ ONLY, VIEW: Read only constraint<br>- NOT NULL: Not null constraint<br>- ERR: Error constraint |
| DELETE_RULE | VARCHAR(9)   | Delete rule for the referential constraint<br>- CASCADE<br>- SET NULL<br>- NO ACTION                                                                                                                                                                                                                                     |
| REF_CON_ID  | NUMBER       | Constraint id of the primary key or unique key constraint defined in the table to be referred to                                                                                                                                                                                                                         |

### 3.2.12. ALL\_DB\_LINKS

Displays information for the database links that the current user can use.

- Columns

| Column Name | Data Type    | Description                  |
|-------------|--------------|------------------------------|
| OWNER       | VARCHAR(128) | Owner of the database link   |
| DB_LINK     | VARCHAR(128) | Name of the database link    |
| USERNAME    | VARCHAR(128) | User ID of the database link |
| HOST        | VARCHAR(128) | Name of the remote database  |

| Column Name | Data Type | Description            |
|-------------|-----------|------------------------|
| CREATED     | DATE      | Database creation time |

- See also

[DBA\\_DB\\_LINKS](#)

[USER\\_DB\\_LINKS](#)

### 3.2.13. ALL\_DEPENDENCIES

Displays dependencies between procedures, functions, packages, package bodies, triggers, and views that the current user can access.

- Columns

| Column Name | Data Type    | Description                                                                                                                                                                                            |
|-------------|--------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| OWNER       | VARCHAR(128) | Owner of the object                                                                                                                                                                                    |
| NAME        | VARCHAR(128) | Name of the object                                                                                                                                                                                     |
| TYPE        | VARCHAR(12)  | Type of the object<br>- INDEX<br>- TABLE<br>- VIEW<br>- SYNONYM<br>- SEQUENCE<br>- PROCEDURE<br>- FUNCTION<br>- PACKAGE<br>- NON-EXISTENT<br>- PACKAGE BODY<br>- TRIGGER<br>- UNDEFINED<br>- DIRECTORY |

| Column Name      | Data Type    | Description                                                                                      |
|------------------|--------------|--------------------------------------------------------------------------------------------------|
|                  |              | - LIBRARY<br>- MATERIALIZED VIEW<br>- JAVA<br>- TYPE<br>- TYPE BODY<br>- OPERATOR<br>- INDEXTYPE |
| PARENT_OBJ_OWNER | VARCHAR(128) | Owner of the parent object                                                                       |
| PARENT_OBJ_NAME  | VARCHAR(128) | Name of the parent object                                                                        |
| PARENT_OBJ_TYPE  | VARCHAR(12)  | Type of the parent object                                                                        |

- See also

[DBA\\_DEPENDENCIES](#)

[USER\\_DEPENDENCIES](#)

### 3.2.14. ALL\_DIRECTORIES

Displays all directories that the current user can access.

- Columns

| Column Name | Data Type     | Description           |
|-------------|---------------|-----------------------|
| NAME        | VARCHAR(128)  | Name of the directory |
| PATH        | VARCHAR(4000) | Path to the directory |

- See also

[DBA\\_DIRECTORIES](#)

### 3.2.15. ALL\_ENCRYPTED\_COLUMNS

Displays information of all encrypted columns that the current user can access.

- Columns

| Column Name    | Data Type    | Description                                                                                                             |
|----------------|--------------|-------------------------------------------------------------------------------------------------------------------------|
| OWNER          | VARCHAR(128) | Owner of the table                                                                                                      |
| TABLE_NAME     | VARCHAR(128) | Name of the table                                                                                                       |
| COLUMN_NAME    | VARCHAR(128) | Name of the column                                                                                                      |
| ENCRYPTION_ALG | VARCHAR(29)  | Encryption algorithm used for the encrypted column                                                                      |
| SALT           | VARCHAR(3)   | Whether the SALT option was used for the encrypted column<br><br>- Y: Salt option used<br><br>- N: Salt option not used |

- See also

[DBA\\_ENCRYPTED\\_COLUMNS](#)

[USER\\_ENCRYPTED\\_COLUMNS](#)

### 3.2.16. ALL\_ERRORS

Displays the recent errors on every accessible view, procedure, function, package, package body, trigger, library, and java object.

- Columns

| Column Name | Data Type    | Description                                                                                                                |
|-------------|--------------|----------------------------------------------------------------------------------------------------------------------------|
| OWNER       | VARCHAR(128) | Owner of the object                                                                                                        |
| NAME        | VARCHAR(128) | Name of the object                                                                                                         |
| TYPE        | VARCHAR(12)  | Type of the object ('VIEW', 'PROCEDURE', 'FUNCTION', 'PACKAGE', 'PACKAGE BODY', 'TRIGGER', 'LIBRARY', 'JAVA', 'UNDEFINED') |
| SEQUENCE    | NUMBER       | Serial number of the error                                                                                                 |

| Column Name    | Data Type     | Description                                         |
|----------------|---------------|-----------------------------------------------------|
| LINE           | NUMBER        | Line number of the error                            |
| POSITION       | NUMBER        | Position in the line of the error                   |
| TEXT           | VARCHAR(4000) | Error description                                   |
| ATTRIBUTE      | VARCHAR(9)    | Type of the error ('ERROR', 'WARNING', 'UNDEFINED') |
| MESSAGE_NUMBER | NUMBER        | Error number                                        |

- See also

[DBA\\_ERRORS](#)

[USER\\_ERRORS](#)

### 3.2.17. ALL\_EXTERNAL\_LOCATIONS

Displays all external table locations that the current user can access.

- Columns

| Column Name    | Data Type     | Description                                          |
|----------------|---------------|------------------------------------------------------|
| OWNER          | VARCHAR(128)  | Owner of the external table location                 |
| TABLE_NAME     | VARCHAR(128)  | Name of the external table location                  |
| DIRECTORY_NAME | VARCHAR(30)   | Name of the directory of the external table location |
| LOCATION       | VARCHAR(4000) | External table location                              |

- See also

[DBA\\_EXTERNAL\\_LOCATIONS](#)

[USER\\_EXTERNAL\\_LOCATIONS](#)

### 3.2.18. ALL\_EXTERNAL\_TABLES

Displays all external tables that the current user can access.

- Columns

| Column Name            | Data Type    | Description                                         |
|------------------------|--------------|-----------------------------------------------------|
| OWNER                  | VARCHAR(128) | Owner of the external table                         |
| TABLE_NAME             | VARCHAR(128) | Name of the external table                          |
| DEFAULT_DIRECTORY_NAME | VARCHAR(30)  | Name of the default directory of the external table |
| REJECT_LIMIT           | VARCHAR      | Reject limit of the external table                  |
| ACCESS_PARAMETERS      | LONG         | Access parameter of the external table              |

- See also

[DBA\\_EXTERNAL\\_TABLES](#)

[USER\\_EXTERNAL\\_TABLES](#)

### 3.2.19. ALL\_IDX\_AND\_LOB\_COLUMNS\_INFO

Displays index columns (including LOB) with index information of all tables that the current user can access

- Columns

| Column Name     | Data Type    | Description            |
|-----------------|--------------|------------------------|
| TABLE_OWNER     | VARCHAR(128) | Owner of the table     |
| TABLE_NAME      | VARCHAR(128) | Name of the table      |
| INDEX_OWNER     | VARCHAR(128) | Owner of the index     |
| INDEX_NAME      | VARCHAR(128) | Name of the index      |
| INDEX_TYPE      | VARCHAR      | Index type (string)    |
| COLUMN_NAME     | VARCHAR(128) | Name of the column     |
| COLUMN_POSITION | NUMBER       | Position of the column |

### 3.2.20. ALL\_IDX\_COLUMNS

Displays index columns of all tables that the current user can access.

- Columns

| Column Name     | Data Type    | Description                                         |
|-----------------|--------------|-----------------------------------------------------|
| INDEX_OWNER     | VARCHAR(128) | Owner of the index                                  |
| INDEX_NAME      | VARCHAR(128) | Name of the index                                   |
| TABLE_OWNER     | VARCHAR(128) | Owner of the table                                  |
| TABLE_NAME      | VARCHAR(128) | Name of the table                                   |
| COLUMN_NAME     | VARCHAR(128) | Name of the column                                  |
| COLUMN_POSITION | NUMBER       | Position of the column within the index             |
| COLUMN_LENGTH   | NUMBER       | Length of the column within the index               |
| DESCEND         | VARCHAR(4)   | Sorting order in columns<br><br>- DESC<br><br>- ASC |

- See also

[DBA\\_IDX\\_COLUMNS](#)

[USER\\_IDX\\_COLUMNS](#)

### 3.2.21. ALL\_IDX\_COLUMNS\_INFO

Displays index columns with index information of all tables that the current user can access.

- Columns

| Column Name | Data Type    | Description                               |
|-------------|--------------|-------------------------------------------|
| INDEX_OWNER | VARCHAR(128) | Owner of the index                        |
| INDEX_NAME  | VARCHAR(128) | Name of the index                         |
| TABLE_OWNER | VARCHAR(128) | Owner of the table                        |
| TABLE_NAME  | VARCHAR(128) | Name of the table                         |
| UNIQUENESS  | VARCHAR(9)   | Whether or not the unique key is indexed. |

| Column Name     | Data Type    | Description                                 |
|-----------------|--------------|---------------------------------------------|
|                 |              | - NONUNIQUE<br>- UNIQUE<br>- UNDEFINED      |
| DISTINCT_KEYS   | NUMBER       | The number of distinct keys in the index    |
| LEAF_BLOCKS     | NUMBER       | The number of leaf blocks in the index      |
| COLUMN_NAME     | VARCHAR(128) | Name of the column                          |
| COLUMN_POSITION | NUMBER       | Position of the column within the index     |
| COLUMN_LENGTH   | NUMBER       | Length of the column within the index       |
| DESCEND         | VARCHAR(4)   | Sorting order in columns<br>- DESC<br>- ASC |

### 3.2.22. ALL\_IDX\_EXPRESSIONS

Displays expressions on function-based index columns of all tables that the current user can access.

- Columns

| Column Name       | Data Type    | Description                               |
|-------------------|--------------|-------------------------------------------|
| INDEX_OWNER       | VARCHAR(128) | Owner of the index                        |
| INDEX_NAME        | VARCHAR(128) | Name of the index                         |
| TABLE_OWNER       | VARCHAR(128) | Owner of the table                        |
| TABLE_NAME        | VARCHAR(128) | Name of the table                         |
| COLUMN_POSITION   | NUMBER       | Position of the column within the index   |
| COLUMN_EXPRESSION | LONG         | Expression of the column within the index |

- See also

[DBA\\_IDX\\_EXPRESSIONS](#)

[USER\\_IDX\\_EXPRESSIONS](#)

### 3.2.23. ALL\_IDX\_PARTITIONS

displays partitions of the indexes that the current user can access.

- Columns

| Column Name        | Data Type    | Description                                                                                |
|--------------------|--------------|--------------------------------------------------------------------------------------------|
| OWNER              | VARCHAR(128) | name of the user                                                                           |
| INDEX_NAME         | VARCHAR(128) | name of the index                                                                          |
| COMPOSITE          | VARCHAR(3)   | whether or not partition belongs to a composite-partitioned index<br><br>- yes<br><br>- no |
| PARTITION_NAME     | VARCHAR(128) | name of the partition                                                                      |
| SUBPARTITION_COUNT | NUMBER       | the number of the subpartitions that belong to the partition                               |
| BOUND              | LONG         | partition bound value                                                                      |
| PARTITION_NO       | NUMBER       | partition number within the index                                                          |
| BLEVEL             | NUMBER       | level of b-tree index (the level is 0 for a b-tree with only a root node)                  |
| LEAF_BLOCKS        | NUMBER       | The number of leaf blocks in the index                                                     |
| DISTINCT_KEYS      | NUMBER       | The number of distinct keys in the index                                                   |
| CLUSTERING_FACTOR  | NUMBER       | The clustering factor of the index                                                         |
| STATUS             | VARCHAR(8)   | Current status of the index<br><br>- USABLE<br><br>- UNUSABLE                              |

| Column Name     | Data Type    | Description                                                             |
|-----------------|--------------|-------------------------------------------------------------------------|
| NUM_ROWS        | NUMBER       | The number of rows in the index                                         |
| TABLESPACE_NAME | VARCHAR(128) | Name of the tablespace to which the partition belongs                   |
| PCT_FREE        | NUMBER       | Value of the PCTFREE parameter                                          |
| INI_TRANS       | NUMBER       | Value of the INITRANS parameter                                         |
| NEXT_EXTENT     | NUMBER       | Size of the next extent                                                 |
| MAX_EXTENTS     | NUMBER       | Value of the MAX EXTENTS parameter                                      |
| BUFFER_POOL     | VARCHAR(7)   | Type of the buffer pool<br><br>- DEFAULT<br><br>- KEEP<br><br>- RECYCLE |
| MAX_TRANS       | NUMBER       | Value of the MAX_TRANS parameter                                        |
| INITIAL_EXTENT  | NUMBER       | Value of the INITIAL_EXTENT parameter                                   |
| MIN_EXTENT      | NUMBER       | Value of the MIN_EXTENT parameter                                       |
| LOGGING         | VARCHAR(3)   | Whether logging is enabled or not<br><br>- YES<br><br>- NO              |
| LAST_ANALYZED   | DATE         | Date of the most recent analysis for the index partition                |

- See also

[DBA\\_IDX\\_PARTITIONS](#)

[USER\\_IDX\\_PARTITIONS](#)

### 3.2.24. ALL\_IDX\_SUBPARTITIONS

Displays subpartitions of the index that the current user can access.

- Columns

| Column Name       | Data Type    | Description                                                             |
|-------------------|--------------|-------------------------------------------------------------------------|
| OWNER             | VARCHAR(128) | Name of the user                                                        |
| INDEX_NAME        | VARCHAR(128) | Name of the index                                                       |
| PARTITION_NAME    | VARCHAR(128) | Name of the index partition                                             |
| SUBPARTITION_NAME | VARCHAR(128) | Name of the index subpartition                                          |
| BOUND             | LONG         | Subpartition bound value                                                |
| SUBPARTITION_NO   | NUMBER       | Subpartition number                                                     |
| STATUS            | VARCHAR(8)   | Current status of the index<br><br>- USABLE<br><br>- UNUSABLE           |
| TABLESPACE_NAME   | VARCHAR(128) | Name of the tablespace to which the subpartition belongs                |
| PCT_FREE          | NUMBER       | Value of the PCTFREE parameter                                          |
| INI_TRANS         | NUMBER       | Value of the INITRANS parameter                                         |
| NEXT_EXTENT       | NUMBER       | Size of the next extent                                                 |
| MAX_EXTENTS       | NUMBER       | Value of the MAX EXTENTS parameter                                      |
| BUFFER_POOL       | VARCHAR(7)   | Type of the buffer pool<br><br>- DEFAULT<br><br>- KEEP<br><br>- RECYCLE |
| LOGGING           | VARCHAR(3)   | Whether logging is enabled or not<br><br>- YES<br><br>- NO              |
| NUM_ROWS          | NUMBER       | The number of rows in the index subpartition                            |
| LAST_ANALYZED     | DATE         | Date of the most recent analysis for the subpartition                   |

- See also

[DBA\\_IDX\\_SUBPARTITIONS](#)

### 3.2.25. ALL\_INDEXES

Displays indexes of all tables that the current user can access.

- Columns

| Column Name | Data Type    | Description                                                                            |
|-------------|--------------|----------------------------------------------------------------------------------------|
| OWNER       | VARCHAR(128) | Owner of the index                                                                     |
| INDEX_NAME  | VARCHAR(128) | Name of the index                                                                      |
| INDEX_TYPE  | VARCHAR(25)  | Type of the index<br>- NORMAL<br>- FUNCTION-BASED<br>- LOB                             |
| TABLE_OWNER | VARCHAR(128) | Owner of the indexed object                                                            |
| TABLE_NAME  | VARCHAR(128) | The indexed object                                                                     |
| TABLE_TYPE  | CHAR(9)      | Type of the indexed object<br>- INDEX<br>- TABLE<br>- VIEW<br>- SYNONYM<br>- UNDEFINED |
| UNIQUENESS  | VARCHAR(9)   | Whether or not the unique key is indexed.<br>- NONUNIQUE<br>- UNIQUE<br>- UNDEFINED    |
| COMPRESSION | VARCHAR(8)   | Whether or not the index is compressed.<br>- DISABLED                                  |

| Column Name       | Data Type    | Description                                                               |
|-------------------|--------------|---------------------------------------------------------------------------|
|                   |              | - ENABLED                                                                 |
| PREFIX_LENGTH     | NUMBER       | The number of prefix column count if compressed.                          |
| TABLESPACE_NAME   | VARCHAR(128) | Name of the tablespace containing the index                               |
| INI_TRANS         | NUMBER       | Value of the INITRANS parameter                                           |
| PCT_FREE          | NUMBER       | Value of the PCTFREE parameter                                            |
| INITIAL_EXTENT    | NUMBER       | Size of the initial extent                                                |
| NEXT_EXTENT       | NUMBER       | Size of the next extent                                                   |
| LOGGING           | VARCHAR(3)   | Whether logging is enabled or not<br><br>- YES<br><br>- NO                |
| BLEVEL            | NUMBER       | Level of B-tree index (The level is 0 for a B-tree with only a root node) |
| LEAF_BLOCKS       | NUMBER       | The number of leaf blocks in the index                                    |
| DISTINCT_KEYS     | NUMBER       | The number of distinct keys in the index                                  |
| CLUSTERING_FACTOR | NUMBER       | The clustering factor of the index                                        |
| STATUS            | VARCHAR(8)   | Indicates whether a nonpartitioned index is VALID or UNUSABLE             |
| NUM_ROWS          | NUMBER       | The number of rows in the index                                           |
| LAST_ANALYZED     | DATE         | Date of the most recent analysis for the table                            |
| PARTITIONED       | VARCHAR(3)   | Whether or not the index is partitioned<br><br>- YES<br><br>- NO          |
| BUFFER_POOL       | VARCHAR(7)   | Type of the buffer pool<br><br>- DEFAULT<br><br>- KEEP                    |

| Column Name         | Data Type    | Description                                                                                       |
|---------------------|--------------|---------------------------------------------------------------------------------------------------|
|                     |              | - RECYCLE                                                                                         |
| GENERATED_BY_SYSTEM | VARCHAR(1)   | Whether or not the name of the index was generated in the system<br><br>- Y<br><br>- N            |
| REFERENTIAL         | VARCHAR(3)   | Whether the index was automatically generated for corresponding referential integrity constraints |
| MAX_EXTENTS         | NUMBER       | Maximum number of extents that the index can use                                                  |
| INCLUDE_COLUMN      | NUMBER       | ID of the last column to be included in the primary key index of the index-organized table        |
| MIN_EXTENTS         | NUMBER       | Minimum number of extents that the index can use                                                  |
| PCT_THRESHOLD       | NUMBER       | Percentage of the block space allowed per index entry.                                            |
| VISIBILITY          | VARCHAR(9)   | Visibility property of an index.                                                                  |
| PARAMETERS          | VARCHAR(30)  | (Not supported)                                                                                   |
| DOMIDX_STATUS       | VARCHAR(12)  | (Not supported)                                                                                   |
| DOMIDX_OPSTATUS     | VARCHAR(6)   | (Not supported)                                                                                   |
| DOMIDX_MANAGEMENT   | VARCHAR2(14) | (Not supported)                                                                                   |

- See also

[DBA\\_INDEXES](#)

[USER\\_INDEXES](#)

### 3.2.26. ALL\_IND\_PENDING\_STATS

Displays indexes of all tables that the current user can access.

- Columns

| Column Name          | Data Type    | Description                                    |
|----------------------|--------------|------------------------------------------------|
| OWNER                | VARCHAR(128) | Owner of the index                             |
| INDEX_NAME           | VARCHAR(128) | Name of the index                              |
| TABLE_OWNER          | VARCHAR(128) | Owner of the indexed object                    |
| TABLE_NAME           | VARCHAR(128) | The indexed object                             |
| PARTITION_NAME       | VARCHAR(128) | Name of the partition                          |
| SUBPARTITION_NAME    | VARCHAR(128) | Name of the subpartition                       |
| BLEVEL               | NUMBER       | Level of B-tree index                          |
| LEAF_BLOCKS          | NUMBER       | The number of leaf blocks in the index         |
| DISTINCT_KEYS        | NUMBER       | The number of distinct keys in the index       |
| AVG_LEAF_BLK_PER_KEY | NUMBER       | Average number of leaf blocks per key          |
| AVG_DATA_BLK_PER_KEY | NUMBER       | Average number of data blocks per key          |
| CLUSTERING_FACTOR    | NUMBER       | The clustering factor of the index             |
| NUM_ROWS             | NUMBER       | The number of rows in the index                |
| LAST_ANALYZED        | DATE         | Date of the most recent analysis for the table |

- See also

[DBA\\_IND\\_PENDING\\_STATS](#)

[USER\\_IND\\_PENDING\\_STATS](#)

### 3.2.27. ALL\_JAVA\_ARGUMENTS

Displays all parameters that the current user can access.

- Columns

| Column Name | Data Type    | Description        |
|-------------|--------------|--------------------|
| OWNER       | VARCHAR(128) | Owner of the class |

| Column Name       | Data Type     | Description                                             |
|-------------------|---------------|---------------------------------------------------------|
| NAME              | VARCHAR(4000) | Name of the class                                       |
| METHOD_INDEX      | NUMBER        | Index of the method                                     |
| METHOD_NAME       | VARCHAR(4000) | Name of the method                                      |
| ARGUMENT_POSITION | NUMBER        | Position of the parameter (start position 1)            |
| ARRAY_DEPTH       | NUMBER        | Array depth of the parameter type                       |
| ARGUMENT_TYPE     | VARCHAR(7)    | Name of the parameter type                              |
| ARGUMENT_CLASS    | VARCHAR(4000) | Actual class name for the parameter whose type is class |

- See also

[DBA\\_JAVA\\_ARGUMENTS](#)

[USER\\_JAVA\\_ARGUMENTS](#)

### 3.2.28. ALL\_JAVA\_CLASSES

Displays all classes that the current user can access.

- Columns

| Column Name   | Data Type     | Description                                 |
|---------------|---------------|---------------------------------------------|
| OWNER         | VARCHAR(128)  | Owner of the class                          |
| NAME          | VARCHAR(4000) | Name of the class                           |
| MAJOR         | NUMBER        | Major version of the class defined in JVM   |
| MINOR         | NUMBER        | Minor version of the class defined in JVM   |
| KIND          | VARCHAR2(9)   | Type of the class<br>- Interface<br>- Class |
| ACCESSIBILITY | VARCHAR(9)    | Accessibility of the class                  |
| IS_INNER      | VARCHAR(1)    | Whether or not the class is an INNER class  |

| Column Name | Data Type     | Description                                            |
|-------------|---------------|--------------------------------------------------------|
| IS_ABSTRACT | VARCHAR(1)    | Whether or not the class is an abstract class          |
| IS_FINAL    | VARCHAR(1)    | Whether or not the class is a final class              |
| STATUS      | VARCHAR(7)    | Status of the library<br><br>- VALID<br><br>- INVALID  |
| SOURCE      | VARCHAR(4000) | Source of the class                                    |
| SUPER       | VARCHAR(4000) | Name of the SUPER class                                |
| OUTER       | VARCHAR(4000) | Name of the OUTER class if the class is an INNER class |

- See also

[DBA\\_JAVA\\_CLASSES](#)

[USER\\_JAVA\\_CLASSES](#)

## 3.2.29. ALL\_JAVA\_FIELDS

Displays all fields that the current user can access.

- Columns

| Column Name   | Data Type     | Description                                  |
|---------------|---------------|----------------------------------------------|
| OWNER         | VARCHAR(128)  | Owner of the field                           |
| NAME          | VARCHAR(4000) | Name of the class                            |
| FIELD_INDEX   | NUMBER        | Index of the field                           |
| FIELD_NAME    | VARCHAR(4000) | Name of the field                            |
| ACCESSIBILITY | VARCHAR(9)    | Accessibility of the field                   |
| IS_STATIC     | VARCHAR(1)    | Whether or not the field is a STATIC field   |
| IS_FINAL      | VARCHAR(1)    | Whether or not the field is a FINAL field    |
| IS_VOLATILE   | VARCHAR(1)    | Whether or not the field is a VOLATILE field |

| Column Name  | Data Type     | Description                                   |
|--------------|---------------|-----------------------------------------------|
| IS_TRANSIENT | VARCHAR(1)    | Whether or not the field is a TRANSIENT field |
| ARRAY_DEPTH  | NUMBER        | Array depth of the field type                 |
| FIELD_TYPE   | VARCHAR(7)    | Name of the field type                        |
| FIELD_CLASS  | VARCHAR(4000) | Actual class name if the field type is class  |

- See also

[DBA\\_JAVA\\_FIELDS](#)

[USER\\_JAVA\\_FIELDS](#)

### 3.2.30. ALL\_JAVA\_METHODS

Displays all methods that the current user can access.

- Columns

| Column Name     | Data Type     | Description                                        |
|-----------------|---------------|----------------------------------------------------|
| OWNER           | VARCHAR(128)  | Owner of the method                                |
| NAME            | VARCHAR(4000) | Name of the class                                  |
| METHOD_INDEX    | NUMBER        | Index of the method                                |
| METHOD_NAME     | VARCHAR(4000) | Name of the method                                 |
| ACCESSIBILITY   | VARCHAR(9)    | Accessibility of the method                        |
| IS_STATIC       | VARCHAR(1)    | Whether or not the method is a STATIC method       |
| IS_FINAL        | VARCHAR(1)    | Whether or not the method is a FINAL method        |
| IS_SYNCHRONIZED | VARCHAR(1)    | Whether or not the method is a SYNCHRONIZED method |
| IS_NATIVE       | VARCHAR(1)    | Whether or not the method is a NATIVE method       |
| IS_STRICT       | VARCHAR(1)    | Whether or not the method is a STRICT method       |
| ARGUMENTS       | NUMBER        | The number of parameters of the method             |

| Column Name  | Data Type     | Description                                   |
|--------------|---------------|-----------------------------------------------|
| THROWS       | NUMBER        | The number of exceptions sent by the method   |
| ARRAY_DEPTH  | NUMBER        | Array depth of the return type                |
| RETURN_TYPE  | VARCHAR(7)    | Name of the return type                       |
| RETURN_CLASS | VARCHAR(4000) | Actual class name if the return type is class |

- See also

[DBA\\_JAVA\\_METHODS](#)

[USER\\_JAVA\\_METHODS](#)

### 3.2.31. ALL\_JOBS

Displays all jobs that the current user can access.

- Columns

| Column Name | Data Type     | Description                                                                                 |
|-------------|---------------|---------------------------------------------------------------------------------------------|
| JOB         | NUMBER        | ID of the job                                                                               |
| SCHEMA_USER | NUMBER        | Owner ID of the job                                                                         |
| NEXT_DATE   | DATE          | Next job execution date                                                                     |
| INTERVAL    | VARCHAR(4000) | Interval used to evaluate the value to be set in NEXT_DATE column after the job is executed |
| BROKEN      | VARCHAR(1)    | Status of the job<br><br>- Y: Invalid<br><br>- N: Valid                                     |
| FAILURES    | NUMBER        | The number of successive failures in job executions                                         |
| WHAT        | VARCHAR(4000) | The contents that this job will execute                                                     |
| INSTANCE    | NUMBER        | Instance number of RDBMS which runs the job                                                 |
| THIS_DATE   | DATE          | Date that this job started                                                                  |

| Column Name | Data Type | Description                              |
|-------------|-----------|------------------------------------------|
| LAST_DATE   | DATE      | Date that this job finished successfully |

- See also

[DBA\\_JOBS](#)

[USER\\_JOBS](#)

### 3.2.32. ALL\_JOBS\_WITH\_NAME

Displays all jobs with job name that the current user can access.

- Columns

| Column Name | Data Type     | Description                                                                                 |
|-------------|---------------|---------------------------------------------------------------------------------------------|
| JOB         | NUMBER        | ID of the job                                                                               |
| NAME        | VARCHAR(100)  | NAME of the job                                                                             |
| SCHEMA_USER | NUMBER        | Owner ID of the job                                                                         |
| NEXT_DATE   | DATE          | Next job execution date                                                                     |
| INTERVAL    | VARCHAR(4000) | Interval used to evaluate the value to be set in NEXT_DATE column after the job is executed |
| BROKEN      | VARCHAR(1)    | Status of the job<br>- Y: Invalid<br>- N: Valid                                             |
| FAILURES    | NUMBER        | The number of successive failures in job executions                                         |
| WHAT        | VARCHAR(4000) | The contents that this job will execute                                                     |
| INSTANCE    | NUMBER        | Instance number of RDBMS which runs the job                                                 |
| THIS_DATE   | DATE          | Date that this job started                                                                  |
| LAST_DATE   | DATE          | Date that this job finished successfully                                                    |

- See also

[DBA\\_JOBS\\_WITH\\_NAME](#)

[USER\\_JOBS\\_WITH\\_NAME](#)

### 3.2.33. ALL\_LIBRARIES

Displays all libraries that the current user can access.

- Columns

| Column Name  | Data Type     | Description                                           |
|--------------|---------------|-------------------------------------------------------|
| OWNER        | VARCHAR(128)  | Owner of the library                                  |
| LIBRARY_NAME | VARCHAR(128)  | Name of the library                                   |
| FILE_SPEC    | VARCHAR(4000) | Path to the library                                   |
| STATUS       | VARCHAR(7)    | Status of the library<br><br>- VALID<br><br>- INVALID |

- See also

[DBA\\_LIBRARIES](#)

[USER\\_LIBRARIES](#)

### 3.2.34. ALL\_LOBS

Displays large objects (LOB) which belong to all tables that the current user can access.

- Columns

| Column Name  | Data Type    | Description                               |
|--------------|--------------|-------------------------------------------|
| OWNER        | VARCHAR(128) | Owner of the object that contains the LOB |
| TABLE_NAME   | VARCHAR(128) | Name of the table that contains the LOB   |
| COLUMN_NAME  | VARCHAR(128) | Name of the LOB column                    |
| SEGMENT_NAME | VARCHAR(128) | Name of the LOB segment                   |

| Column Name   | Data Type    | Description                                                                                                 |
|---------------|--------------|-------------------------------------------------------------------------------------------------------------|
| TABSPACE_NAME | VARCHAR(128) | Name of the tablespace to which the LOB belongs                                                             |
| INDEX_NAME    | VARCHAR(128) | Name of the LOB index                                                                                       |
| IN_ROW        | VARCHAR(3)   | Whether STORAGE IN ROW is enabled or not<br><br>- YES<br><br>- NO                                           |
| ENCRYPT       | VARCHAR(3)   | Whether ENCRYPTION is enabled or not<br><br>- YES<br><br>- NO                                               |
| COMPRESSION   | VARCHAR(10)  | Whether COMPRESS is low, medium, high or not<br><br>- LOW<br><br>- MEDIUM<br><br>- HIGH<br><br>- NOCOMPRESS |
| DEDUPLICATION | VARCHAR(3)   | Whether DEDUPLICATE is enabled or not<br><br>- LOB<br><br>- NO                                              |
| BUFFER_POOL   | VARCHAR(7)   | Type of the buffer pool<br><br>- DEFAULT<br><br>- KEEP<br><br>- RECYCLE                                     |
| MAX_EXTENTS   | NUMBER       | Value of the MAX EXTENTS parameter                                                                          |
| PARTITIONED   | VARCHAR(3)   | Whether or not the LOB column is in a partitioned table<br><br>- YES<br><br>- NO                            |

- See also

[DBA\\_LOBS](#)

[USER\\_LOBS](#)

### 3.2.35. ALL\_LOB\_PARTITIONS

Displays all LOB partitions that the current user can access.

- Columns

| Column Name        | Data Type    | Description                                                                    |
|--------------------|--------------|--------------------------------------------------------------------------------|
| OWNER              | VARCHAR(128) | Name of the user                                                               |
| TABLE_NAME         | VARCHAR(128) | Name of the table                                                              |
| COLUMN_NAME        | VARCHAR(128) | Name of the LOB column                                                         |
| PARTITION_NAME     | VARCHAR(128) | Name of the table partition                                                    |
| LOB_PARTITION_NAME | VARCHAR(128) | Name of the LOB partition                                                      |
| LOB_INDPART_NAME   | VARCHAR(128) | Name of the LOB index partition                                                |
| PARTITION_NO       | NUMBER       | Partition number within the table                                              |
| COMPOSITE          | VARCHAR(3)   | Whether or not the partition is a composite partition<br><br>- LOB<br><br>- NO |
| IN_ROW             | VARCHAR(3)   | Whether STORAGE IN ROW is enabled or not<br><br>- YES<br><br>- NO              |
| ENCRYPT            | VARCHAR(3)   | Whether ENCRYPTION is enabled or not<br><br>- YES<br><br>- NO                  |
| COMPRESSION        | VARCHAR(10)  | Whether COMPRESS is low, medium, high or not<br><br>- LOW<br><br>- MEDIUM      |

| Column Name     | Data Type    | Description                                                     |
|-----------------|--------------|-----------------------------------------------------------------|
|                 |              | - HIGH<br>- NOCOMPRESS                                          |
| DEDUPLICATION   | VARCHAR(3)   | Whether DEDUPLICATE is enabled or not<br><br>- YES<br>- NO      |
| TABLESPACE_NAME | VARCHAR(128) | Name of the tablespace to which the LOB partition belongs       |
| NEXT_EXTENT     | NUMBER       | Size of the next extent                                         |
| MAX_EXTENTS     | NUMBER       | Value of the MAX EXTENTS parameter                              |
| BUFFER_POOL     | VARCHAR(7)   | Type of the buffer pool<br><br>- DEFAULT<br>- KEEP<br>- RECYCLE |

- See also

[DBA\\_LOB\\_PARTITIONS](#)

[USER\\_LOB\\_PARTITIONS](#)

### 3.2.36. ALL\_LOB\_SUBPARTITIONS

Displays LOB subpartitions that the current user can access.

- Columns

| Column Name    | Data Type    | Description                 |
|----------------|--------------|-----------------------------|
| OWNER          | VARCHAR(128) | Name of the user            |
| TABLE_NAME     | VARCHAR(128) | Name of the table           |
| COLUMN_NAME    | VARCHAR(128) | Name of the LOB column      |
| PARTITION_NAME | VARCHAR(128) | Name of the table partition |

| Column Name           | Data Type    | Description                                                                                                 |
|-----------------------|--------------|-------------------------------------------------------------------------------------------------------------|
| LOB_NAME              | VARCHAR(128) | Name of the LOB                                                                                             |
| LOB_PARTITION_NAME    | VARCHAR(128) | Name of the LOB partition                                                                                   |
| SUBPARTITION_NAME     | VARCHAR(128) | Name of the table subpartition                                                                              |
| LOB_SUBPARTITION_NAME | VARCHAR(128) | Name of the LOB subpartition                                                                                |
| LOB_INDSUBPART_NAME   | VARCHAR(128) | Name of the LOB index subpartition                                                                          |
| SUBPARTITION_NO       | NUMBER       | Subpartition number within the table                                                                        |
| IN_ROW                | VARCHAR(3)   | Whether STORAGE IN ROW is enabled or not<br><br>- YES<br><br>- NO                                           |
| ENCRYPT               | VARCHAR(3)   | Whether ENCRYPTION is enabled or not<br><br>- YES<br><br>- NO                                               |
| COMPRESSION           | VARCHAR(10)  | Whether COMPRESS is low, medium, high or not<br><br>- LOW<br><br>- MEDIUM<br><br>- HIGH<br><br>- NOCOMPRESS |
| DEDUPLICATION         | VARCHAR(3)   | Whether DEDUPLICATE is enabled or not<br><br>- YES<br><br>- NO                                              |
| TABLESPACE_NAME       | VARCHAR(128) | Name of the tablespace to which the LOB partition belongs                                                   |
| NEXT_EXTENT           | NUMBER       | Size of the next extent                                                                                     |
| MAX_EXTENTS           | NUMBER       | Value of the MAX EXTENTS parameter                                                                          |

| Column Name | Data Type  | Description                                                             |
|-------------|------------|-------------------------------------------------------------------------|
| BUFFER_POOL | VARCHAR(7) | Type of the buffer pool<br><br>- DEFAULT<br><br>- KEEP<br><br>- RECYCLE |

- See also

[DBA\\_LOB\\_SUBPARTITIONS](#)

[USER\\_LOB\\_SUBPARTITIONS](#)

### 3.2.37. ALL\_LOG\_GROUPS

Displays the log group definitions on the tables accessible to the current user.

- Columns

| Column Name    | Data Type    | Description                                                                                                                                                      |
|----------------|--------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| OWNER          | VARCHAR(128) | Owner of the log group definition                                                                                                                                |
| LOG_GROUP_NAME | VARCHAR(128) | Name of the log group definition                                                                                                                                 |
| TABLE_NAME     | VARCHAR(128) | Name of the table where the log group is defined                                                                                                                 |
| LOG_GROUP_TYPE | VARCHAR(19)  | Type of the log group:<br><br>- PRIMARY KEY LOGGING<br><br>- UNIQUE KEY LOGGING<br><br>- FOREIGN KEY LOGGING<br><br>- ALL COLUMN LOGGING<br><br>- USER LOG GROUP |
| ALWAYS         | VARCHAR(11)  | Y Indicates that the log group is logged whenever a row is updated; N Indicates that the log group is logged whenever a member column is updated.                |
| GENERATED      | VARCHAR(14)  | Whether the name of the supplemental log group indicates the system generated (GENERATED NAME) or not (USER NAME).                                               |

- See also

[DBA\\_LOG\\_GROUPS](#)

[USER\\_LOG\\_GROUPS](#)

### 3.2.38. ALL\_LOG\_GROUP\_COLUMNS

Displays columns that are accessible to the current user and that are specified in log groups.

- Columns

| Column Name      | Data Type    | Description                                                                                     |
|------------------|--------------|-------------------------------------------------------------------------------------------------|
| OWNER            | VARCHAR(128) | Owner of the log group definition                                                               |
| LOG_GROUP_NAME   | VARCHAR(128) | Name of the log group definition                                                                |
| TABLE_NAME       | VARCHAR(128) | Name of the table where the log group is defined                                                |
| COLUMN_NAME      | VARCHAR(128) | Name of the column or attribute of the object type column specified in the log group definition |
| POSITION         | NUMBER       | Original position of the column or attribute in the definition of the object                    |
| LOGGING_PROPERTY | VARCHAR(6)   | Indicates whether the column or attribute would be supplementally logged (LOG) or not (NO LOG)  |

- See also

[DBA\\_LOG\\_GROUP\\_COLUMNS](#)

[USER\\_LOG\\_GROUP\\_COLUMNS](#)

### 3.2.39. ALL\_METHOD\_PARAMS

Displays all the method parameters of the type that the current user can access.

- Columns

| Column Name | Data Type    | Description        |
|-------------|--------------|--------------------|
| OWNER       | VARCHAR(128) | Owner of this type |
| TYPE_NAME   | VARCHAR(128) | Name of this type  |

| Column Name        | Data Type      | Description                                     |
|--------------------|----------------|-------------------------------------------------|
| METHOD_NAME        | VARCHAR(128)   | Name of this method                             |
| METHOD_NO          | NUMBER         | Number of this method                           |
| PARAM_NAME         | VARCHAR(128)   | Name of this parameter                          |
| PARAM_NO           | NUMBER         | Location of this parameter                      |
| PARAM_MODE         | VARCHAR(9)     | Mode of this parameter                          |
| PARAM_TYPE_MOD     | VARCHAR(128)   | Modifier of this parameter type (not supported) |
| PARAM_TYPE_OWNER   | VARCHAR(128)   | Owner of this parameter type                    |
| PARAM_TYPE_NAME    | VARCHAR(65532) | Name of this parameter type                     |
| CHARACTER_SET_NAME | VARCHAR(128)   | Name of the Character set (not supported)       |

- See also

[DBA\\_METHOD\\_PARAMS](#)

[USER\\_METHOD\\_PARAMS](#)

### 3.2.40. ALL\_METHOD\_RESULTS

Displays the result of the method of this type that the current user can access.

- Columns

| Column Name      | Data Type    | Description                                  |
|------------------|--------------|----------------------------------------------|
| OWNER            | VARCHAR(128) | Owner of this type                           |
| TYPE_NAME        | VARCHAR(128) | Name of this type                            |
| METHOD_NAME      | VARCHAR(128) | Name of this method                          |
| METHOD_NO        | NUMBER       | Method number of this method                 |
| PARAM_TYPE_MOD   | VARCHAR(128) | Modifier of this result type (not supported) |
| PARAM_TYPE_OWNER | VARCHAR(128) | Owner of this result type                    |

| Column Name        | Data Type      | Description                           |
|--------------------|----------------|---------------------------------------|
| PARAM_TYPE_NAME    | VARCHAR(65532) | Name of this result type              |
| CHARACTER_SET_NAME | VARCHAR(128)   | Name of Character set (not supported) |

- See also

[DBA\\_METHOD\\_RESULTS](#)

[USER\\_METHOD\\_RESULTS](#)

### 3.2.41. ALL\_MVIEWS

Displays all materialized views that the current user can access.

- Columns

| Column Name         | Data Type    | Description                                                                                |
|---------------------|--------------|--------------------------------------------------------------------------------------------|
| OWNER               | VARCHAR(128) | Owner of the materialized view                                                             |
| MVIEW_NAME          | VARCHAR(128) | Name of the materialized view                                                              |
| CONTAINER_NAME      | VARCHAR(128) | Container table name of the materialized view                                              |
| QUERY               | LONG         | Defining query of the materialized view.                                                   |
| QUERY_LEN           | NUMBER(38)   | Length of the defining query in bytes                                                      |
| UPDATABLE           | CHAR(1)      | Whether or not the materialized view is updatable<br>- Y: Updatable<br>- N: Not updatable  |
| UPDATE_LOG          | CHAR(0)      | Update log for an updatable materialized view                                              |
| MASTER_ROLLBACK_SEG | CHAR(0)      | Rollback segment of the master site                                                        |
| MASTER_LINK         | CHAR(0)      | Database link to the master site                                                           |
| REWRITE_ENABLED     | VARCHAR(1)   | Whether or not the rewrite of a materialized view query is enabled<br>- Y: rewrite enabled |

| Column Name        | Data Type  | Description                                                                                           |
|--------------------|------------|-------------------------------------------------------------------------------------------------------|
|                    |            | - N: rewrite disabled                                                                                 |
| REWRITE_CAPABILITY | CHAR(7)    | Rewrite capability of the materialized view                                                           |
| REFRESH_MODE       | VARCHAR(6) | Refresh mode of the materialized view<br>- Commit<br>- Demand                                         |
| REFRESH_METHOD     | VARCHAR(1) | Refresh method of the materialized view<br>- C (Complete)<br>- F (Fast)<br>- ? (Force)<br>- N (Never) |
| BUILD_MODE         | VARCHAR(9) | Build mode of the materialized view<br>- Immediate<br>- Deferred<br>- Prebuilt                        |
| FAST_REFRESHABLE   | CHAR(2)    | Whether or not the materialized view is eligible for fast refresh                                     |
| LAST_REFRESH_TYPE  | CHAR(2)    | The most recent refresh type                                                                          |
| LAST_REFRESH_DATE  | DATE       | The most recent refresh date                                                                          |
| STALENESS          | VARCHAR(9) | Staleness of the materialized view                                                                    |
| AFTER_FAST_REFRESH | CHAR(2)    | Staleness of the materialized view after the expected fast refresh                                    |
| UNKNOWN_PREBUILT   | CHAR(1)    | Whether or not the materialized view is of unknown status as it is prebuilt (Y/N)                     |
| UNKNOWN_PLSQL_FUNC | CHAR(1)    | Whether or not the materialized view is of unknown status as it contains psm function (Y/N)           |

| Column Name            | Data Type     | Description                                                                                                                                            |
|------------------------|---------------|--------------------------------------------------------------------------------------------------------------------------------------------------------|
| UNKNOWN_EXTERNAL_TABLE | CHAR(1)       | Whether or not the materialized view is of unknown status as it contains external tables (Y/N)                                                         |
| UNKNOWN_CONSIDER_FRESH | CHAR(1)       | Whether or not the materialized view is of unknown status as it is marked consider fresh (Y/N)                                                         |
| UNKNOWN_IMPORT         | CHAR(1)       | Whether or not the materialized view is of unknown status as it is imported (Y/N)                                                                      |
| UNKNOWN_TRUSTED_FD     | CHAR(1)       | Whether or not the materialized view is of unknown status as it uses trusted constraints (Y/N)                                                         |
| COMPILE_STATE          | CHAR(5)       | Compilation status of the materialized view                                                                                                            |
| USE_NO_INDEX           | CHAR(1)       | Whether or not the index is used in the materialized view creation<br><br>- Y: using no index<br><br>- N: using the default index                      |
| STALE_SINCE            | VARCHAR(0)    | Time passed from when the materialized view became stale                                                                                               |
| INTERVAL               | VARCHAR(2000) | Date function used to compute the next refresh time                                                                                                    |
| REDUCED_PRECISION      | VARCHAR(1)    | Whether or not the reduced precision is used for prebuilt materialized view (Y/N)                                                                      |
| REFRESH_KEY            | VARCHAR(11)   | Represents the key type which is used to refresh a materialized view.<br><br>- ROWID: rowid<br><br>- PRIMARY KEY: primary key column of the base table |

- See also

[DBA\\_MVIEWS](#)

[USER\\_MVIEWS](#)

### 3.2.42. ALL\_MVIEW\_COMMENTS

Displays comments on the materialized views that the current user can access.

- Columns

| Column Name | Data Type     | Description                      |
|-------------|---------------|----------------------------------|
| OWNER       | VARCHAR(128)  | Owner of the materialized view   |
| MVIEW_NAME  | VARCHAR(128)  | Name of the materialized view    |
| COMMENTS    | VARCHAR(4000) | Comment on the materialized view |

- See also

[DBA\\_MVIEW\\_COMMENTS](#)

[USER\\_MVIEW\\_COMMENTS](#)

### 3.2.43. ALL\_MVIEW\_DETAIL\_RELATIONS

Describes all such detail relations defined for materialized views in the database that the current user can access.

- Columns

| Column Name     | Data Type    | Description                                                           |
|-----------------|--------------|-----------------------------------------------------------------------|
| OWNER           | VARCHAR(128) | Owner of the materialized view                                        |
| MVIEW_NAME      | VARCHAR(128) | Name of the materialized view                                         |
| DETAILOBJ_OWNER | VARCHAR(128) | Owner of the related object                                           |
| DETAILOBJ_NAME  | VARCHAR(128) | Name of the related object                                            |
| DETAILOBJ_TYPE  | VARCHAR(23)  | Type of the related object                                            |
| DETAILOBJ_ALIAS | VARCHAR(23)  | Implicit or explicit alias for detail relation. Not supported column. |

- See also

[DBA\\_MVIEW\\_DETAIL\\_RELATIONS](#)

[USER\\_MVIEW\\_DETAIL\\_RELATIONS](#)

### 3.2.44. ALL\_MVIEW\_LOGS

Displays all materialized view logs that the current user can access.

- Columns

| Column Name        | Data Type     | Description                                                                                                     |
|--------------------|---------------|-----------------------------------------------------------------------------------------------------------------|
| LOG_OWNER          | VARCHAR2(128) | Owner of the materialized view log                                                                              |
| MASTER             | VARCHAR2(128) | Name of the master table whose changes are logged                                                               |
| LOG_TABLE          | VARCHAR2(128) | Name of the log table                                                                                           |
| LOG_TRIGGER        | VARCHAR2(128) | after-row trigger on the master table which inserts rows into the log                                           |
| ROWIDS             | VARCHAR2(3)   | Whether to record the rowid information                                                                         |
| PRIMARY_KEY        | VARCHAR2(3)   | Whether to record the primary key information                                                                   |
| FILTER_COLUMNS     | VARCHAR2(3)   | Whether to record the filter column information                                                                 |
| SEQUENCE           | VARCHAR2(3)   | Whether to record the sequence information                                                                      |
| INCLUDE_NEW_VALUES | VARCHAR2(3)   | Whether to record both old and new values<br>- Y: Keeps both old and new values.<br>- N: Keeps old values only. |

- See also

[DBA\\_MVIEW\\_LOGS](#)

[USER\\_MVIEW\\_LOGS](#)

### 3.2.45. ALL\_MVIEW\_REFRESH\_TIMES

Describes refresh times of all materialized views accessible to the current user.

- Columns

| Column Name | Data Type    | Description                    |
|-------------|--------------|--------------------------------|
| OWNER       | VARCHAR(128) | Owner of the materialized view |
| NAME        | VARCHAR(128) | Name of the materialized view  |

| Column Name  | Data Type    | Description                                                  |
|--------------|--------------|--------------------------------------------------------------|
| MASTER_OWNER | VARCHAR(128) | Owner of the master table                                    |
| MASTER       | VARCHAR(128) | Name of the master table                                     |
| LAST_REFRESH | DATE         | SYSDATE from the master site at the time of the last refresh |

- See also

[DBA\\_MVIEW\\_REFRESH\\_TIMES](#)

[USER\\_MVIEW\\_REFRESH\\_TIMES](#)

## 3.2.46. ALL\_OBJECTS

Displays all objects that the current user can access.

- Columns

| Column Name    | Data Type    | Description                                                                                                                            |
|----------------|--------------|----------------------------------------------------------------------------------------------------------------------------------------|
| OWNER          | VARCHAR(128) | Owner of the object                                                                                                                    |
| OBJECT_NAME    | VARCHAR(128) | Name of the object                                                                                                                     |
| SUBOBJECT_NAME | VARCHAR(128) | Name of the subobject (For example, partition name)                                                                                    |
| OBJECT_ID      | NUMBER       | ID of the object                                                                                                                       |
| DATA_OBJECT_ID | NUMBER       | ID of the segment                                                                                                                      |
| OBJECT_TYPE    | VARCHAR(23)  | Type of the object<br><br>- INDEX<br><br>- TABLE<br><br>- VIEW<br><br>- SYNONYM<br><br>- SEQUENCE<br><br>- PROCEDURE<br><br>- FUNCTION |

| Column Name    | Data Type   | Description                                                                                                                                                                                                                                                                                                                                            |
|----------------|-------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                |             | <ul style="list-style-type: none"> <li>- PACKAGE</li> <li>- PACKAGE BODY</li> <li>- TRIGGER</li> <li>- TABLE PARTITION</li> <li>- INDEX PARTITION</li> <li>- LOB</li> <li>- LOB PARTITION</li> <li>- TABLE SUBPARTITION</li> <li>- INDEX SUBPARTITION</li> <li>- LOB SUBPARTITION</li> <li>- DIRECTORY</li> <li>- UNDEFINED</li> <li>- TYPE</li> </ul> |
| OBJECT_TYPE_NO | NUMBER      | Type number of the object                                                                                                                                                                                                                                                                                                                              |
| CREATED        | DATE        | Date that the object was created                                                                                                                                                                                                                                                                                                                       |
| LAST_DDL_TIME  | DATE        | Date that the modification took effect on the object through DDL statement                                                                                                                                                                                                                                                                             |
| TIMESTAMP      | VARCHAR(19) | Timestamp for the latest changes on object structure                                                                                                                                                                                                                                                                                                   |
| STATUS         | VARCHAR(7)  | Status of the object <ul style="list-style-type: none"> <li>- VALID</li> <li>- INVALID</li> </ul>                                                                                                                                                                                                                                                      |
| TEMPORARY      | VARCHAR(1)  | Whether or not the object is a temporary object <ul style="list-style-type: none"> <li>- Y</li> <li>- N</li> </ul>                                                                                                                                                                                                                                     |

- See also

[DBA\\_OBJECTS](#)

### 3.2.47. ALL\_PACKAGE\_COLL\_TYPES

Displays collection types of the package that the current user can access,

- Columns

| Column Name       | Data Type    | Description                                                    |
|-------------------|--------------|----------------------------------------------------------------|
| OWNER             | VARCHAR(128) | Owner of this type                                             |
| PACKAGE_NAME      | VARCHAR(128) | Package name of this type                                      |
| TYPE_NAME         | VARCHAR(128) | Name of this type                                              |
| UPPER_BOUND       | NUMBER       | Maximum of a varray or length constraint of an index by table. |
| ELEM_TYPE_MOD     | VARCHAR(7)   | Type modifier of the field                                     |
| ELEM_TYPE_OWNER   | VARCHAR(128) | Owner of the field                                             |
| ELEM_TYPE_NAME    | VARCHAR(128) | Type name of the field                                         |
| ELEM_TYPE_PACKAGE | VARCHAR(128) | Name of the package containing the attribute type.             |
| PRECISION         | NUMBER       | Precision of this type                                         |
| SCALE             | NUMBER       | Scale of this type                                             |
| INDEX_TYPE        | VARCHAR(128) | Type name of the index by type (VARCHAR2 or BINARY_INTEGER)    |

- See also

[DBA\\_PACKAGE\\_COLL\\_TYPES](#)

[USER\\_PACKAGE\\_COLL\\_TYPES](#)

### 3.2.48. ALL\_PACKAGE\_MEMBER\_TYPES

Displays attributes of the package type that the current user can access.

- Columns

| Column Name              | Data Type    | Description                                                        |
|--------------------------|--------------|--------------------------------------------------------------------|
| OWNER                    | VARCHAR(128) | Owner of this type                                                 |
| PACKAGE_NAME             | VARCHAR(128) | Package name of this type                                          |
| TYPE_NAME                | VARCHAR(128) | Name of this type                                                  |
| TYPE_NO                  | NUMBER       | Type number of this type                                           |
| PRECISION                | NUMBER       | Precision of this type                                             |
| SCALE                    | NUMBER       | Scale of this type                                                 |
| TYPE_OBJ_ID              | NUMBER       | Object id of this type                                             |
| MEMBER_NO                | NUMBER       | Member number of this type                                         |
| KIND                     | NUMBER       | Member kind of this type                                           |
| LIMIT                    | NUMBER       | Limit of this type                                                 |
| COUNT                    | NUMBER       | Count of this type                                                 |
| POS_NO                   | NUMBER       | Order number of the attribute                                      |
| ATTR_NAME                | VARCHAR(128) | Name of the record field                                           |
| ATTR_TYPE_MOD            | VARCHAR(7)   | Type modifier of the record field                                  |
| ATTR_TYPE_OWNER          | VARCHAR(128) | Owner of the record field                                          |
| ATTR_TYPE_NAME           | VARCHAR(128) | Type name of the record field (Same as name of the TYPE_NO column) |
| ATTR_TYPE_PACKAGE<br>AGE | VARCHAR(128) | Name of the package containing the attribute type.                 |

- See also

[DBA\\_PACKAGE\\_MEMBER\\_TYPES](#)

[USER\\_PACKAGE\\_MEMBER\\_TYPES](#)

### 3.2.49. ALL\_PACKAGE\_TYPES

Displays member types of the package that the current user can access

- Columns

| Column Name  | Data Type    | Description                               |
|--------------|--------------|-------------------------------------------|
| OWNER        | VARCHAR(128) | Owner of this type                        |
| PACKAGE_NAME | VARCHAR(128) | Package name of this type                 |
| TYPE_NAME    | VARCHAR(128) | Name of this type                         |
| KIND         | NUMBER       | Member kind of this type                  |
| KIND_NAME    | VARCHAR      | Member kind of this type in string        |
| LIMIT        | NUMBER       | Limit of this type                        |
| COUNT        | NUMBER       | Count of this type                        |
| MEMBER_NO    | NUMBER       | Member number of the type in the package. |

- See also

[DBA\\_PACKAGE\\_TYPES](#)

[USER\\_PACKAGE\\_TYPES](#)

### 3.2.50. ALL\_PART\_INDEXES

Displays partition information for the partitioned indexes that the current user can access.

- Columns

| Column Name       | Data Type    | Description                      |
|-------------------|--------------|----------------------------------|
| OWNER             | VARCHAR(128) | Owner of the partitioned index   |
| INDEX_NAME        | VARCHAR(128) | Name of the partitioned index    |
| TABLE_NAME        | VARCHAR(128) | Name of the partitioned table    |
| PARTITIONING_TYPE | VARCHAR(5)   | Type of the partition<br>- RANGE |

| Column Name                   | Data Type    | Description                                                                                                                                                        |
|-------------------------------|--------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                               |              | <ul style="list-style-type: none"> <li>- HASH</li> <li>- LIST</li> </ul>                                                                                           |
| SUBPARTITION<br>ING_TYPE      | VARCHAR(5)   | Type of the composite partition <ul style="list-style-type: none"> <li>- RANGE</li> <li>- HASH</li> <li>- LIST</li> <li>- NONE</li> </ul>                          |
| PARTITION_COUNT               | NUMBER       | The number of partitions                                                                                                                                           |
| DEF_SUBPARTI<br>TION_COUNT    | NUMBER       | In the case of composite partition, the default number for the number of subpartitions (when only it is defined)                                                   |
| PARTITION<br>ING_KEY_COUNT    | NUMBER       | The number of columns used as partitioning key                                                                                                                     |
| SUBPARTITION<br>ING_KEY_COUNT | NUMBER       | For a composite partition, the number of columns used as subpartitioning key                                                                                       |
| LOCALITY                      | VARCHAR(6)   | Whether the index is a local index or global index <ul style="list-style-type: none"> <li>- LOCAL: Local index</li> <li>- GLOBAL: Global index</li> </ul>          |
| ALIGNMENT                     | VARCHAR(12)  | Whether or not the partitioned key column is prefixed <ul style="list-style-type: none"> <li>- PREFIXED: prefixed</li> <li>- NON_PREFIXED: not prefixed</li> </ul> |
| DEF_TA<br>BLESPEACE_NAME      | VARCHAR(128) | Name of the default tablespace used to add or split a partition                                                                                                    |
| DEF_PCT_FREE                  | NUMBER       | Default value of PCTFREE used to add a partition                                                                                                                   |
| DEF_INI_TRANS                 | NUMBER       | Default value of INITRANS used to add a partition.                                                                                                                 |
| DEF_NEXT_EXTENT               | NUMBER       | Default value of NEXT_EXTENT used to add a partition                                                                                                               |
| DEF_MAX_EXTENTS               | NUMBER       | Default value of MAX_EXTENTS used to add a partition                                                                                                               |
| DEF_BUFFER_POOL               | VARCHAR(7)   | Default value of buffer pool type used to add a partition                                                                                                          |

| Column Name | Data Type | Description                      |
|-------------|-----------|----------------------------------|
|             |           | - DEFAULT<br>- KEEP<br>- RECYCLE |

- See also

[DBA\\_PART\\_INDEXES](#)

[USER\\_PART\\_INDEXES](#)

### 3.2.51. ALL\_PART\_KEY\_COLUMNS

Displays partitioning key columns of the partitioned tables or indexes that the current user can access.

- Columns

| Column Name     | Data Type    | Description                                        |
|-----------------|--------------|----------------------------------------------------|
| OWNER           | VARCHAR(128) | Owner of the partitioned table or index            |
| NAME            | VARCHAR(128) | Name of the partitioned table or index             |
| OBJECT_TYPE     | CHAR(5)      | Type of the partition<br>- TABLE<br>- INDEX        |
| COLUMN_NAME     | VARCHAR(128) | Name of the column                                 |
| COLUMN_POSITION | NUMBER       | Position of the column within the partitioning key |

- See also

[DBA\\_PART\\_KEY\\_COLUMNS](#)

[USER\\_PART\\_KEY\\_COLUMNS](#)

### 3.2.52. ALL\_PART\_LOBS

Displays LOBs which belong to all partitioned tables that the current user can access.

- Columns

| Column Name         | Data Type    | Description                                                                                                   |
|---------------------|--------------|---------------------------------------------------------------------------------------------------------------|
| OWNER               | VARCHAR(128) | Owner of the partitioned table                                                                                |
| TABLE_NAME          | VARCHAR(128) | Name of the partitioned table                                                                                 |
| COLUMN_NAME         | VARCHAR(128) | Name of the LOB column                                                                                        |
| LOB_NAME            | VARCHAR(128) | Name of the LOB                                                                                               |
| LOB_INDEX_NAME      | VARCHAR(128) | Name of the LOB index                                                                                         |
| DEF_IN_ROW          | VARCHAR(3)   | Whether STORAGE IN ROW used by default to add or split a partition is enabled or not<br><br>- YES<br><br>- NO |
| DEF_TABLESPACE_NAME | VARCHAR(128) | Name of the default tablespace used to add or split a partition                                               |
| DEF_NEXT_EXTENT     | NUMBER       | Default value of NEXT_EXTENT used to add a partition                                                          |
| DEF_MAX_EXTENTS     | NUMBER       | Default value of MAX_EXTENTS used to add a partition                                                          |
| DEF_BUFFER_POOL     | VARCHAR(7)   | Default value of buffer pool type used to add a partition<br><br>- DEFAULT<br><br>- KEEP<br><br>- RECYCLE     |

- See also

[DBA\\_PART\\_LOBS](#)

[USER\\_PART\\_LOBS](#)

### 3.2.53. ALL\_PART\_TABLES

Displays partitioning information of all partitioned tables that the current user can access.

- Columns

| Column Name               | Data Type    | Description                                                                      |
|---------------------------|--------------|----------------------------------------------------------------------------------|
| OWNER                     | VARCHAR(128) | Owner of the partitioned table                                                   |
| TABLE_NAME                | VARCHAR(128) | Name of the partitioned table                                                    |
| PARTITIONING_TYPE         | VARCHAR(5)   | Type of the partitioning method<br>- RANGE<br>- HASH<br>- LIST                   |
| SUBPARTITIONING_TYPE      | VARCHAR(5)   | Type of the composite partition<br>- RANGE<br>- HASH<br>- LIST<br>- NONE         |
| PARTITION_COUNT           | NUMBER       | The number of partitions                                                         |
| DEF_SUBPARTITION_COUNT    | NUMBER       | For a composite partition, the number of subpartitions (only when it is defined) |
| PARTITIONING_KEY_COUNT    | NUMBER       | The number of columns participating in the partitioning key                      |
| SUBPARTITIONING_KEY_COUNT | NUMBER       | For a composite partition, the number of columns used as subpartitioning key     |
| DEF_TABLESPACE_NAME       | VARCHAR(128) | Name of default tablespace used to add or split a partition                      |
| DEF_PCT_FREE              | NUMBER       | Default value of PCTFREE used to add a partition                                 |
| DEF_INI_TRANS             | NUMBER       | Default value of INITRANS used to add a partition                                |
| DEF_NEXT_EXTENT           | NUMBER       | Default value of NEXT_EXTENT used to add a partition                             |
| DEF_MAX_EXTENTS           | NUMBER       | Default value of MAX_EXTENTS used to add a partition                             |
| DEF_LOGGING               | VARCHAR(3)   | Default value of LOGGING used to add a partition                                 |
| DEF_COMPRESSION           | VARCHAR(3)   | Default value of COMPRESS used to add a partition                                |

| Column Name      | Data Type    | Description                                                                                               |
|------------------|--------------|-----------------------------------------------------------------------------------------------------------|
| DEF_COMPRESS_FOR | VARCHAR(12)  | Default value of COMPRESS_FOR used to add a partition                                                     |
| DEF_BUFFER_POOL  | VARCHAR(7)   | Default value of buffer pool type used to add a partition<br><br>- DEFAULT<br><br>- KEEP<br><br>- RECYCLE |
| MAX_TRANS        | NUMBER       | Value of the MAX_TRANS parameter                                                                          |
| INITIAL_EXTENT   | NUMBER       | Value of the INITIAL_EXTENT parameter                                                                     |
| MIN_EXTENT       | NUMBER       | Value of the MIN_EXTENT parameter                                                                         |
| LOGGING          | VARCHAR(3)   | Whether logging is enabled or not<br><br>- YES<br><br>- NO                                                |
| INTERVAL         | VARCHAR(128) | String of the interval values                                                                             |

- See also

[DBA\\_PART\\_TABLES](#)

[USER\\_PART\\_TABLES](#)

### 3.2.54. ALL\_PROCEEDURES

Displays all procedures defined in all procedures and tables that the current user owns.

- Columns

| Column Name    | Data Type    | Description                                |
|----------------|--------------|--------------------------------------------|
| OWNER          | VARCHAR(128) | Owner of the procedure                     |
| OBJECT_NAME    | VARCHAR(128) | Name of the object                         |
| PROCEDURE_NAME | VARCHAR(128) | Name of the procedure                      |
| AGGREGATE      | CHAR(2)      | Whether or not it is an AGGREGATE function |

| Column Name   | Data Type  | Description                                    |
|---------------|------------|------------------------------------------------|
| PIPELINED     | CHAR(2)    | Whether or not the table function is pipelined |
| IMPLTYPEOWNER | CHAR(0)    | Owner of the IMPLEMENTATION type               |
| IMPLTYPENAME  | CHAR(0)    | Name of the IMPLEMENTATION type                |
| PARALLEL      | CHAR(2)    | Whether PARALLEL is enabled or not             |
| INTERFACE     | CHAR(2)    | Whether or not it is an INTERFACE function     |
| DETERMINISTIC | CHAR(2)    | Whether or not it is a DETERMINISTIC function  |
| FUNCTIONABLE  | VARCHAR(3) | Whether or not it is a function                |
| WRAPPED       | VARCHAR(3) | Whether or not it is encrypted (deprecated)    |
| AUTHID        | VARCHAR(7) | Name of AUTHID                                 |
| MEMBER_NO     | NUMBER     | Index of its object                            |

- See also

[DBA\\_PROCEduRES](#)

[USER\\_PROCEduRES](#)

### 3.2.55. ALL\_QUEUES

Describes all queues on which the current user has enqueue or dequeue privileges.

- Columns

| Column Name | Data Type    | Description                            |
|-------------|--------------|----------------------------------------|
| OWNER       | VARCHAR(128) | Owner of the queue                     |
| NAME        | VARCHAR(128) | Name of the queue                      |
| QUEUE_TABLE | VARCHAR(128) | Name of the queue table                |
| QID         | NUMBER       | Object number of the queue             |
| QUEUE_TYPE  | VARCHAR(20)  | Type of the queue<br>- EXCEPTION_QUEUE |

| Column Name     | Data Type    | Description                                                                         |
|-----------------|--------------|-------------------------------------------------------------------------------------|
|                 |              | - NON_PERSISTENT_QUEUE<br>- NORMAL_QUEUE                                            |
| MAX_RETRIES     | NUMBER       | Maximum number of retries allowed when dequeuing from the queue                     |
| RETRY_DELAY     | NUMBER       | Time interval between retries                                                       |
| ENQUEUE_ENABLED | VARCHAR(3)   | Indicates whether the queue is enabled for enqueue (YES) or not (NO)                |
| DEQUEUE_ENABLED | VARCHAR(3)   | Indicates whether the queue is enabled for dequeue (YES) or not (NO)                |
| RETENTION       | VARCHAR(256) | Time interval (in seconds) processed messages are retained in the queue, or FOREVER |
| USER_COMMENT    | VARCHAR(100) | User specified comment                                                              |

- See also

[DBA\\_QUEUES](#)

[USER\\_QUEUES](#)

### 3.2.56. ALL\_QUEUE\_SUBSCRIBERS

Shows the list of subscribers that the current user has privilege to dequeue from.

- Columns

| Column Name   | Data Type     | Description                                           |
|---------------|---------------|-------------------------------------------------------|
| OWNER         | VARCHAR(128)  | Owner of the queue                                    |
| QUEUE_NAME    | VARCHAR(128)  | Name of the queue                                     |
| QUEUE_TABLE   | VARCHAR(128)  | Name of the queue table on which the queue is defined |
| CONSUMER_NAME | VARCHAR(128)  | Name of the subscriber                                |
| ADDRESS       | VARCHAR(1024) | Address of the subscriber                             |
| PROTOCOL      | NUMBER        | Protocol of the subscriber                            |

| Column Name                      | Data Type  | Description                                                                           |
|----------------------------------|------------|---------------------------------------------------------------------------------------|
| TRANSFORMATION                   | VARCHAR(0) | Transformation for the subscriber                                                     |
| RULE                             | VARCHAR(0) | Rule condition for the subscriber                                                     |
| DELIVERY_MODE                    | CHAR(10)   | Message delivery mode for the subscriber                                              |
| IF_NON<br>DURABLE_SUB<br>SCRIBER | CHAR(2)    | Indicates whether the subscriber is a non-durable subscriber (YES) or not (NO)        |
| QUEUE_TO_QUEUE                   | CHAR(5)    | Indicates whether the subscriber is a queue-to-queue subscriber (TRUE) or not (FALSE) |

- See also

[DBA\\_QUEUE\\_SUBSCRIBERS](#)

[USER\\_QUEUE\\_SUBSCRIBERS](#)

## 3.2.57. ALL\_QUEUE\_TABLES

Describes the queues in the queue tables accessible to the current user.

- Columns

| Column Name          | Data Type    | Description                                                                |
|----------------------|--------------|----------------------------------------------------------------------------|
| OWNER                | VARCHAR(128) | Owner of the queue table                                                   |
| QUEUE_TABLE          | VARCHAR(128) | Name of the queue table                                                    |
| TYPE                 | VARCHAR(6)   | Type of user data<br>- RAW: Raw type<br>- OBJECT: User-defined object type |
| OBJECT_TYPE          | VARCHAR(0)   | Object type of the payload when TYPE is OBJECT                             |
| SORT_ORDER           | VARCHAR(22)  | User-specified sort order                                                  |
| RECIPIENTS           | VARCHAR(8)   | SINGLE or MULTIPLE recipients                                              |
| MESSAGE_GROUP<br>ING | VARCHAR(13)  | NONE or TRANSACTIONAL                                                      |

| Column Name        | Data Type    | Description                                                                                                                                                                                                                                  |
|--------------------|--------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| PRIMARY_INSTANCE   | NUMBER       | Indicates the instance number of the instance which is the primary owner of the queue table. A value of 0 indicates that there is no primary owner                                                                                           |
| SECONDARY_INSTANCE | NUMBER       | Indicates the instance number of the instance which is the secondary owner of the queue table. This instance becomes the owner of the queue table if the primary owner is not alive. A value of 0 indicates that there is no secondary owner |
| OWNER_INSTANCE     | NUMBER       | Instance number of the instance which currently owns the queue table                                                                                                                                                                         |
| USER_COMMENT       | VARCHAR(100) | User specified comment                                                                                                                                                                                                                       |

- See also

[DBA\\_QUEUE\\_TABLES](#)

[USER\\_QUEUE\\_TABLES](#)

### 3.2.58. ALL\_RECYCLEBIN

Displays information of all temporarily deleted objects that the current user can access.

- Columns

| Column Name   | Data Type    | Description                                           |
|---------------|--------------|-------------------------------------------------------|
| OBJECT_NAME   | VARCHAR(128) | Current name of the object                            |
| ORIGINAL_NAME | VARCHAR(128) | Original name of the object                           |
| TYPE          | VARCHAR(7)   | Type of the object<br>- Table<br>- Index<br>- Trigger |
| TS_NAME       | VARCHAR(128) | Name of the tablespace containing the object          |
| CREATETIME    | VARCHAR(19)  | Timestamp for the object creation                     |

| Column Name | Data Type    | Description                                     |
|-------------|--------------|-------------------------------------------------|
| DROPTIME    | VARCHAR(19)  | Timestamp for the object deletion               |
| DROPTSN     | NUMBER       | TSN of the transaction which dropped the object |
| BASE_OBJECT | VARCHAR(256) | Object ID of the base object                    |
| SPACE       | VARCHAR(0)   | Number of blocks used by the object             |

- See also

[DBA\\_RECYCLEBIN](#)

[USER\\_RECYCLEBIN](#)

### 3.2.59. ALL\_SCHEDULER\_CHAINS

Displays all scheduler chains that the current user can access.

- Columns

| Column Name         | Data Type     | Description                               |
|---------------------|---------------|-------------------------------------------|
| OWNER_ID            | NUMBER        | ID of the owner                           |
| CHAIN_NAME          | VARCHAR2(30)  | Name of the chain                         |
| RULE_SET_NAME       | VARCHAR2(30)  | This column is for future use.            |
| EVALUATION_INTERVAL | INTERVAL      | This column is for future use.            |
| ENABLED             | VARCHAR(5)    | Status of the chain (enabled or disabled) |
| COMMENTS            | VARCHAR2(240) | Comments on the chain                     |

- See also

[DBA\\_SCHEDULER\\_CHAINS](#)

[USER\\_SCHEDULER\\_CHAINS](#)

### 3.2.60. ALL\_SCHEDULER\_JOBS

Displays all scheduler jobs that the current user can access.

- Columns

| Column Name         | Data Type      | Description                                        |
|---------------------|----------------|----------------------------------------------------|
| OWNER_ID            | NUMBER         | ID of the owner                                    |
| OWNER               | VARCHAR2(4000) | Name of the owner                                  |
| JOB_NAME            | VARCHAR2(30)   | Name of the job                                    |
| JOB_SUBNAME         | VARCHAR2(30)   | Subname of the job (for a job in chain)            |
| JOB_STYLE           | VARCHAR2(11)   | This column is for future use.                     |
| JOB_CREATOR         | VARCHAR2(30)   | This column is for future use.                     |
| CLIENT_ID           | VARCHAR2(64)   | This column is for future use.                     |
| GLOBAL_UID          | VARCHAR2(32)   | This column is for future use.                     |
| PROGRAM_OWNER       | VARCHAR2(128)  | This column is for future use.                     |
| PROGRAM_NAME        | VARCHAR2(30)   | This column is for future use.                     |
| JOB_TYPE            | VARCHAR2(16)   | Job action type:<br><br>- PSM_BLOCK<br><br>- CHAIN |
| JOB_ACTION          | VARCHAR2(4000) | Job action                                         |
| NUMBER_OF_ARGUMENTS | NUMBER         | This column is for future use.                     |
| SCHEDULE_OWNER      | VARCHAR2(4000) | This column is for future use.                     |
| SCHEDULE_NAME       | VARCHAR2(4000) | This column is for future use.                     |
| SCHEDULE_TYPE       | VARCHAR2(12)   | This column is for future use.                     |
| START_DATE          | TIMESTAMP(6)   | Start date of the job                              |
| REPEAT_INTERVAL     | VARCHAR2(4000) | Cron expression                                    |

| Column Name        | Data Type      | Description                                                |
|--------------------|----------------|------------------------------------------------------------|
| EVENT_QUEUE_OWNER  | VARCHAR2(30)   | This column is for future use.                             |
| EVENT_QUEUE_NAME   | VARCHAR2(30)   | This column is for future use.                             |
| EVENT_QUEUE_AGENT  | VARCHAR2(256)  | This column is for future use.                             |
| EVENT_CONDITION    | VARCHAR2(4000) | This column is for future use.                             |
| EVENT_RULE         | VARCHAR2(65)   | This column is for future use.                             |
| FILE_WATCHER_OWNER | VARCHAR2(65)   | This column is for future use.                             |
| FILE_WATCHER_NAME  | VARCHAR2(65)   | This column is for future use.                             |
| END_DATE           | TIMESTAMP(6)   | Date after which the job will no longer run                |
| JOB_CLASS          | VARCHAR2(30)   | This column is for future use.                             |
| ENABLED            | VARCHAR2(5)    | Indicates whether the job is enabled (TRUE) or not (FALSE) |
| AUTO_DROP          | VARCHAR2(5)    | This column is for future use.                             |
| RESTARTABLE        | VARCHAR2(5)    | This column is for future use.                             |
| STATE              | VARCHAR2(15)   | This column is for future use.                             |
| JOB_PRIORITY       | NUMBER         | This column is for future use.                             |
| RUN_COUNT          | NUMBER         | Number of times the job has run                            |
| MAX_RUNS           | NUMBER         | This column is for future use.                             |
| FAILURE_COUNT      | NUMBER         | Number of times the job has failed to run                  |
| MAX_FAILURES       | NUMBER         | This column is for future use.                             |
| RETRY_COUNT        | NUMBER         | This column is for future use.                             |
| LAST_START_DATE    | TIMESTAMP(6)   | This column is for future use.                             |
| LAST_RUN_DURATION  | INTERVAL       | This column is for future use.                             |
| NEXT_RUN_DATE      | TIMESTAMP(6)   | This column is for future use.                             |

| Column Name                   | Data Type      | Description                      |
|-------------------------------|----------------|----------------------------------|
| SCHEDULE_LIMIT                | INTERVAL       | This column is for future use.   |
| MAX_RUN_DURATION              | INTERVAL       | This column is for future use.   |
| LOGGING_LEVEL                 | VARCHAR2(30)   | This column is for future use.   |
| STOP_ON_WINDOW_CLOSE          | VARCHAR2(5)    | This column is for future use.   |
| INSTANCE_STICKINESS           | VARCHAR2(5)    | This column is for future use.   |
| RAISE_EVENTS                  | VARCHAR2(4000) | This column is for future use.   |
| SYSTEM                        | VARCHAR2(5)    | This column is for future use.   |
| JOB_WEIGHT                    | NUMBER         | This column is for future use.   |
| NLS_ENV                       | VARCHAR2(4000) | This column is for future use.   |
| SOURCE                        | VARCHAR2(128)  | This column is for future use.   |
| NUMBER_OF_DESTINATIONS        | NUMBER         | This column is for future use.   |
| DESTINATION_OWNER             | VARCHAR2(128)  | This column is for future use.   |
| DESTINATION                   | VARCHAR2(128)  | This column is for future use.   |
| CREDENTIAL_OWNER              | VARCHAR2(30)   | This column is for future use.   |
| CREDENTIAL_NAME               | VARCHAR2(30)   | This column is for future use.   |
| INSTANCE_ID                   | NUMBER         | ID of the instance (for TAC)     |
| DEFERRED_DROP                 | VARCHAR2(5)    | This column is for future use.   |
| ALLOW_RUNS_IN_RESTRICTED_MODE | VARCHAR2(5)    | This column is for future use.   |
| COMMENTS                      | VARCHAR2(240)  | Comments on the job              |
| FLAGS                         | NUMBER         | This column is for internal use. |
| JOB_ID                        | NUMBER         | This column is for internal use. |

| Column Name | Data Type    | Description                      |
|-------------|--------------|----------------------------------|
| CHAIN_NAME  | VARCHAR2(30) | This column is for internal use. |
| STEP_NAME   | VARCHAR2(30) | This column is for internal use. |
| CTX_ID      | NUMBER       | This column is for internal use. |

- See also

[DBA\\_SCHEDULER\\_JOBS](#)

[USER\\_SCHEDULER\\_JOBS](#)

### 3.2.61. ALL\_SCHEDULER\_JOB\_LOG

Displays all scheduler job log that the current user can access.

- Columns

| Column Name | Data Type      | Description                     |
|-------------|----------------|---------------------------------|
| LOG_ID      | NUMBER         | ID of the log                   |
| LOG_DATE    | TIMESTAMP(6)   | Date of the log entry           |
| OWNER_ID    | NUMBER         | ID of the owner for this log    |
| OWNER       | VARCHAR2(4000) | Owner of the job for this log   |
| JOB_NAME    | VARCHAR2(128)  | Name of the job for this log    |
| JOB_SUBNAME | VARCHAR2(128)  | Subname of the job for this log |
| JOB_CLASS   | VARCHAR2(30)   | Class of the job for this log   |
| OPERATION   | VARCHAR2(30)   | The job operation               |
| STATUS      | VARCHAR2(30)   | The status of job               |
| USER_NAME   | VARCHAR2(65)   | The name of job                 |
| CLIENT_ID   | VARCHAR2(65)   | The id of client                |
| GLOBAL_UID  | VARCHAR2(32)   | The global id of client         |

| Column Name       | Data Type     | Description                         |
|-------------------|---------------|-------------------------------------|
| CREDENTIAL_OWNER  | VARCHAR2(32)  | The credential owner                |
| CREDENTIAL_NAME   | VARCHAR2(32)  | The credential name                 |
| DESTINATION_OWNER | VARCHAR2(128) | The destination owner               |
| DESTINATION       | VARCHAR2(128) | The destination                     |
| ADDITIONAL_INFO   | CLOB          | additional information of this log. |

- See also

[DBA\\_SCHEDULER\\_JOB\\_LOG](#)

[USER\\_SCHEDULER\\_JOB\\_LOG](#)

### 3.2.62. ALL\_SCHEDULER\_JOB\_RUN\_DETAILS

Displays all scheduler job run details that the current user can access.

- Columns

| Column Name       | Data Type      | Description                          |
|-------------------|----------------|--------------------------------------|
| LOG_ID            | NUMBER         | ID of the log                        |
| LOG_DATE          | TIMESTAMP(6)   | Date of the log entry                |
| OWNER_ID          | NUMBER         | ID of the owner for this log         |
| OWNER             | VARCHAR2(4000) | Owner of the job for this log        |
| JOB_NAME          | VARCHAR2(128)  | Name of the job for this log         |
| JOB_SUBNAME       | VARCHAR2(128)  | Subname of the job for this log      |
| STATUS            | VARCHAR2(30)   | The status of job                    |
| ERROR#            | NUMBER         | The error number                     |
| REQ_START_DATE    | TIMESTAMP(6)   | The requested time to start the job  |
| ACTUAL_START_DATE | TIMESTAMP(6)   | The actual start time the job ran on |

| Column Name       | Data Type     | Description                         |
|-------------------|---------------|-------------------------------------|
| INSTANCE_ID       | NUMBER        | The id of instance                  |
| SESSION_ID        | VARCHAR2(30)  | The session id of job               |
| SLAVE_PID         | VARCHAR2(30)  | The process id of the slave session |
| CPU_USED          | NUMBER        | The elapsed time spent using CPU    |
| CREDENTIAL_OWNER  | VARCHAR2(32)  | The credential owner                |
| CREDENTIAL_NAME   | VARCHAR2(32)  | The credential name                 |
| DESTINATION_OWNER | VARCHAR2(128) | The destination owner               |
| DESTINATION       | VARCHAR2(128) | The destination                     |
| ADDITIONAL_INFO   | CLOB          | additional information of this log. |

- See also

[DBA\\_SCHEDULER\\_JOB\\_RUN\\_DETAILS](#)

[USER\\_SCHEDULER\\_JOB\\_RUN\\_DETAILS](#)

### 3.2.63. ALL\_SCHEDULER\_PROGRAMS

Displays all scheduler programs that the current user can access.

- Columns

| Column Name    | Data Type      | Description                        |
|----------------|----------------|------------------------------------|
| OWNER_ID       | NUMBER         | ID of the owner                    |
| OWNER          | VARCHAR2(4000) | Name of the owner                  |
| PROGRAM_NAME   | VARCHAR2(128)  | Name of the program                |
| PROGRAM_TYPE   | VARCHAR2(16)   | Type of the program<br>- PSM_BLOCK |
| PROGRAM_ACTION | VARCHAR2(4000) | Program action                     |

| Column Name         | Data Type      | Description                                                    |
|---------------------|----------------|----------------------------------------------------------------|
| NUMBER_OF_ARGUMENTS | NUMBER         | This column is for future use.                                 |
| ENABLED             | VARCHAR2(5)    | Indicates whether the program is enabled (TRUE) or not (FALSE) |
| DETACHED            | VARCHAR2(5)    | This column is for future use.                                 |
| SCHEDULE_LIMIT      | INTERVAL       | This column is for future use.                                 |
| PRIORITY            | NUMBER         | This column is for future use.                                 |
| WEIGHT              | NUMBER         | This column is for future use.                                 |
| MAX_RUNS            | NUMBER         | This column is for future use.                                 |
| MAX_FAILURES        | NUMBER         | This column is for future use.                                 |
| MAX_RUN_DURATION    | INTERVAL       | This column is for future use.                                 |
| NLS_ENV             | VARCHAR2(4000) | This column is for future use.                                 |
| COMMENTS            | VARCHAR2(240)  | Comments on the program                                        |

- See also

[DBA\\_SCHEDULER\\_PROGRAMS](#)

[USER\\_SCHEDULER\\_PROGRAMS](#)

### 3.2.64. ALL\_SCHEDULER\_RULES

Displays all scheduler rules that the current user can access.

- Columns

| Column Name | Data Type      | Description           |
|-------------|----------------|-----------------------|
| OWNER_ID    | NUMBER         | ID of the owner       |
| CHAIN_NAME  | VARCHAR2(30)   | Name of the chain     |
| CONDITION   | VARCHAR2(4000) | Condition of the rule |
| ACTION      | VARCHAR2(4000) | Action of the rule    |

| Column Name | Data Type     | Description          |
|-------------|---------------|----------------------|
| RULE_NAME   | VARCHAR2(30)  | Name of the rule     |
| COMMENTS    | VARCHAR2(240) | Comments on the rule |

- See also

[DBA\\_SCHEDULER\\_RULES](#)

[USER\\_SCHEDULER\\_RULES](#)

### 3.2.65. ALL\_SCHEDULER\_RULE\_CTX

Displays all scheduler rule ctx that the current user can access.

- Columns

| Column Name | Data Type      | Description                          |
|-------------|----------------|--------------------------------------|
| OWNER_ID    | NUMBER         | ID of the owner                      |
| CHAIN_NAME  | VARCHAR2(30)   | Name of the chain                    |
| CONDITION   | VARCHAR2(4000) | Condition of the rule                |
| ACTION      | VARCHAR2(4000) | Action of the rule                   |
| RULE_NAME   | VARCHAR2(30)   | Name of the rule                     |
| CTX_ID      | NUMBER         | ID of the context (for internal use) |

- See also

[DBA\\_SCHEDULER\\_RULE\\_CTX](#)

[USER\\_SCHEDULER\\_RULE\\_CTX](#)

### 3.2.66. ALL\_SCHEDULER\_RUNNING\_JOBS

Displays all scheduler running jobs that the current user can access.

- Columns

| Column Name         | Data Type      | Description                                        |
|---------------------|----------------|----------------------------------------------------|
| SESS_ID             | NUMBER         | Session ID                                         |
| SESS_SERIAL_NO      | NUMBER         | A serial number of a session                       |
| OWNER_ID            | NUMBER         | ID of the owner                                    |
| OWNER               | VARCHAR2(4000) | Name of the owner                                  |
| JOB_NAME            | VARCHAR2(30)   | Name of the job                                    |
| JOB_SUBNAME         | VARCHAR2(30)   | Subname of the job (for a job in chain)            |
| JOB_STYLE           | VARCHAR2(11)   | This column is for future use.                     |
| JOB_CREATOR         | VARCHAR2(30)   | This column is for future use.                     |
| CLIENT_ID           | VARCHAR2(64)   | This column is for future use.                     |
| GLOBAL_UID          | VARCHAR2(32)   | This column is for future use.                     |
| PROGRAM_OWNER       | VARCHAR2(128)  | This column is for future use.                     |
| PROGRAM_NAME        | VARCHAR2(30)   | This column is for future use.                     |
| JOB_TYPE            | VARCHAR2(16)   | Job action type:<br><br>- PSM_BLOCK<br><br>- CHAIN |
| JOB_ACTION          | VARCHAR2(4000) | Job action                                         |
| NUMBER_OF_ARGUMENTS | NUMBER         | This column is for future use.                     |
| SCHEDULE_OWNER      | VARCHAR2(4000) | This column is for future use.                     |
| SCHEDULE_NAME       | VARCHAR2(4000) | This column is for future use.                     |
| SCHEDULE_TYPE       | VARCHAR2(12)   | This column is for future use.                     |
| START_DATE          | TIMESTAMP(6)   | Start date of the job                              |
| REPEAT_INTERVAL     | VARCHAR2(4000) | Cron expression                                    |
| EVENT_QUEUE_OWNER   | VARCHAR2(30)   | This column is for future use.                     |

| Column Name        | Data Type      | Description                                                |
|--------------------|----------------|------------------------------------------------------------|
| EVENT_QUEUE_NAME   | VARCHAR2(30)   | This column is for future use.                             |
| EVENT_QUEUE_AGENT  | VARCHAR2(256)  | This column is for future use.                             |
| EVENT_CONDITION    | VARCHAR2(4000) | This column is for future use.                             |
| EVENT_RULE         | VARCHAR2(65)   | This column is for future use.                             |
| FILE_WATCHER_OWNER | VARCHAR2(65)   | This column is for future use.                             |
| FILE_WATCHER_NAME  | VARCHAR2(65)   | This column is for future use.                             |
| END_DATE           | TIMESTAMP(6)   | Date after which the job will no longer run                |
| JOB_CLASS          | VARCHAR2(30)   | This column is for future use.                             |
| ENABLED            | VARCHAR2(5)    | Indicates whether the job is enabled (TRUE) or not (FALSE) |
| AUTO_DROP          | VARCHAR2(5)    | This column is for future use.                             |
| RESTARTABLE        | VARCHAR2(5)    | This column is for future use.                             |
| STATE              | VARCHAR2(15)   | This column is for future use.                             |
| JOB_PRIORITY       | NUMBER         | This column is for future use.                             |
| RUN_COUNT          | NUMBER         | This column is for future use.                             |
| MAX_RUNS           | NUMBER         | This column is for future use.                             |
| FAILURE_COUNT      | NUMBER         | This column is for future use.                             |
| MAX_FAILURES       | NUMBER         | This column is for future use.                             |
| RETRY_COUNT        | NUMBER         | This column is for future use.                             |
| LAST_START_DATE    | TIMESTAMP(6)   | This column is for future use.                             |
| LAST_RUN_DURATION  | INTERVAL       | This column is for future use.                             |
| NEXT_RUN_DATE      | TIMESTAMP(6)   | This column is for future use.                             |
| SCHEDULE_LIMIT     | INTERVAL       | This column is for future use.                             |

| Column Name                   | Data Type      | Description                      |
|-------------------------------|----------------|----------------------------------|
| MAX_RUN_DURATION              | INTERVAL       | This column is for future use.   |
| LOGGING_LEVEL                 | VARCHAR2(30)   | This column is for future use.   |
| STOP_ON_WINDOW_CLOSE          | VARCHAR2(5)    | This column is for future use.   |
| INSTANCE_STICKINESS           | VARCHAR2(5)    | This column is for future use.   |
| RAISE_EVENTS                  | VARCHAR2(4000) | This column is for future use.   |
| SYSTEM                        | VARCHAR2(5)    | This column is for future use.   |
| JOB_WEIGHT                    | NUMBER         | This column is for future use.   |
| NLS_ENV                       | VARCHAR2(4000) | This column is for future use.   |
| SOURCE                        | VARCHAR2(128)  | This column is for future use.   |
| NUMBER_OF_DESTINATIONS        | NUMBER         | This column is for future use.   |
| DESTINATION_OWNER             | VARCHAR2(128)  | This column is for future use.   |
| DESTINATION                   | VARCHAR2(128)  | This column is for future use.   |
| CREDENTIAL_OWNER              | VARCHAR2(30)   | This column is for future use.   |
| CREDENTIAL_NAME               | VARCHAR2(30)   | This column is for future use.   |
| INSTANCE_ID                   | NUMBER         | ID of the instance (for TAC)     |
| DEFERRED_DROP                 | VARCHAR2(5)    | This column is for future use.   |
| ALLOW_RUNS_IN_RESTRICTED_MODE | VARCHAR2(5)    | This column is for future use.   |
| COMMENTS                      | VARCHAR2(240)  | Comments on the job              |
| FLAGS                         | NUMBER         | This column is for internal use. |
| JOB_ID                        | NUMBER         | This column is for internal use. |
| CHAIN_NAME                    | VARCHAR2(30)   | This column is for internal use. |

| Column Name | Data Type    | Description                      |
|-------------|--------------|----------------------------------|
| STEP_NAME   | VARCHAR2(30) | This column is for internal use. |
| CTX_ID      | NUMBER       | This column is for internal use. |

- See also

[DBA\\_SCHEDULER\\_RUNNING\\_JOBS](#)

[USER\\_SCHEDULER\\_RUNNING\\_JOBS](#)

### 3.2.67. ALL\_SCHEDULER\_SCHEDULES

Displays all scheduler schedules that the current user can access.

- Columns

| Column Name        | Data Type      | Description                                 |
|--------------------|----------------|---------------------------------------------|
| OWNER_ID           | NUMBER         | ID of the owner                             |
| OWNER              | VARCHAR2(4000) | Name of the owner                           |
| SCHEDULE_NAME      | VARCHAR2(128)  | Name of the schedule                        |
| SCHEDULE_TYPE      | VARCHAR2(16)   | Type of the schedule                        |
| START_DATE         | TIMESTAMP(6)   | Start date of the job                       |
| REPEAT_INTERVAL    | VARCHAR2(4000) | Cron expression                             |
| EVENT_QUEUE_OWNER  | VARCHAR2(30)   | This column is for future use.              |
| EVENT_QUEUE_NAME   | VARCHAR2(30)   | This column is for future use.              |
| EVENT_QUEUE_AGENT  | VARCHAR2(256)  | This column is for future use.              |
| EVENT_CONDITION    | VARCHAR2(4000) | This column is for future use.              |
| FILE_WATCHER_OWNER | VARCHAR2(65)   | This column is for future use.              |
| FILE_WATCHER_NAME  | VARCHAR2(65)   | This column is for future use.              |
| END_DATE           | TIMESTAMP(6)   | Date after which the job will no longer run |

| Column Name | Data Type     | Description             |
|-------------|---------------|-------------------------|
| COMMENTS    | VARCHAR2(240) | Comments on the program |

- See also

[DBA\\_SCHEDULER\\_SCHEDULES](#)

[USER\\_SCHEDULER\\_SCHEDULES](#)

### 3.2.68. ALL\_SCHEDULER\_STEPS

Displays all scheduler steps that the current user can access.

- Columns

| Column Name  | Data Type     | Description          |
|--------------|---------------|----------------------|
| OWNER_ID     | NUMBER        | ID of the owner      |
| CHAIN_NAME   | VARCHAR2(30)  | Name of the chain    |
| STEP_NAME    | VARCHAR2(30)  | Name of the step     |
| PROGRAM_NAME | VARCHAR2(30)  | Name of the program  |
| COMMENTS     | VARCHAR2(240) | Comments on the step |

- See also

[DBA\\_SCHEDULER\\_STEPS](#)

[USER\\_SCHEDULER\\_STEPS](#)

### 3.2.69. ALL\_SCHEDULER\_STEP\_CTX

Displays all scheduler step context the current user can access.

- Columns

| Column Name | Data Type    | Description       |
|-------------|--------------|-------------------|
| OWNER_ID    | NUMBER       | ID of the owner   |
| CHAIN_NAME  | VARCHAR2(30) | Name of the chain |

| Column Name | Data Type    | Description                                  |
|-------------|--------------|----------------------------------------------|
| STEP_NAME   | VARCHAR2(30) | Name of the step                             |
| CTX_ID      | NUMBER       | This column is for internal use.             |
| JOB_ID      | NUMBER       | ID of the job                                |
| INSTANCE_ID | NUMBER       | ID of the instance (for TAC)                 |
| COMPLETED   | VARCHAR2(5)  | State of the step context (completed or not) |
| STATE       | VARCHAR2(16) | State of the step context                    |
| START_DATE  | TIMESTAMP(6) | This column is for internal use.             |
| END_DATE    | TIMESTAMP(6) | This column is for internal use.             |
| ERROR_CODE  | NUMBER       | This column is for internal use.             |
| DURATION    | INTERVAL     | This column is for future use.               |

- See also

[DBA\\_SCHEDULER\\_STEP\\_CTX](#)

[USER\\_SCHEDULER\\_STEP\\_CTX](#)

## 3.2.70. ALL\_SEGMENTS

Displays all segments that the current user can access.

- Columns

| Column Name    | Data Type    | Description                                                                 |
|----------------|--------------|-----------------------------------------------------------------------------|
| SEGMENT_NAME   | VARCHAR(128) | Name of the segment                                                         |
| PARTITION_NAME | VARCHAR(128) | Partition number of the object (Set to NULL for non-partitioned objects)    |
| SEGMENT_TYPE   | VARCHAR(18)  | Type of the segment<br>- TABLE<br>- TABLE PARTITION<br>- TABLE SUBPARTITION |

| Column Name     | Data Type    | Description                                                                                                                                                                                                           |
|-----------------|--------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                 |              | <ul style="list-style-type: none"> <li>- LOB</li> <li>- INDEX</li> <li>- INDEX PARTITION</li> <li>- INDEX SUBPARTITION</li> <li>- LOB</li> <li>- LOB PARTITION</li> <li>- LOB SUBPARTITION</li> <li>- UNDO</li> </ul> |
| TABLESPACE_NAME | VARCHAR(128) | 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                                                                                                                                                                         |
| BLOCKS          | NUMBER       | Size of the segment in blocks                                                                                                                                                                                         |
| EXTENTS         | NUMBER       | Size of the segment in extents                                                                                                                                                                                        |
| RELATIVE_FNO    | NUMBER       | Relative file number of the segment header                                                                                                                                                                            |
| BUFFER_POOL     | VARCHAR(7)   | Type of the buffer pool <ul style="list-style-type: none"> <li>- DEFAULT</li> <li>- KEEP</li> <li>- RECYCLE</li> </ul>                                                                                                |
| MAX_EXTENTS     | NUMBER       | Maximum number of extents that the segment can use                                                                                                                                                                    |
| BYTES           | NUMBER       | Size of segment in bytes                                                                                                                                                                                              |
| INITIAL_EXTENT  | NUMBER       | Size in bytes requested for the initial extent of the segment                                                                                                                                                         |
| NEXT_EXTENT     | NUMBER       | Size in bytes of the next extent to be allocated to the segment                                                                                                                                                       |

- See also

[DBA\\_SEGMENTS](#)

[USER\\_SEGMENTS](#)

### 3.2.71. ALL\_SEQUENCES

Displays all sequences that the current user can access.

- Columns

| Column Name    | Data Type    | Description                                                                                       |
|----------------|--------------|---------------------------------------------------------------------------------------------------|
| SEQUENCE_OWNER | VARCHAR(128) | Owner of the sequence                                                                             |
| SEQUENCE_NAME  | VARCHAR(128) | Name of the sequence                                                                              |
| MIN_VALUE      | NUMBER       | Minimum value of the sequence                                                                     |
| MAX_VALUE      | NUMBER       | Maximum value of the sequence                                                                     |
| INCREMENT_BY   | NUMBER       | Increment value of the sequence                                                                   |
| CYCLE_FLAG     | VARCHAR(1)   | Whether the sequence is reused when the maximum value is reached<br><br>- Y<br><br>- N<br><br>- E |
| ORDER_FLAG     | CHAR(1)      | Whether sequences are generated in order<br><br>- Y                                               |
| IF_AVAIL       | VARCHAR(1)   | Whether the last value is readable<br><br>- Y<br><br>- N<br><br>- E                               |
| CACHE_SIZE     | NUMBER       | The number of cached sequences                                                                    |
| LAST_NUMBER    | NUMBER       | Last sequence number to be written to disk                                                        |

- See also

[DBA\\_SEQUENCES](#)

[USER\\_SEQUENCES](#)

### 3.2.72. ALL\_SOURCE

Displays source texts of the stored objects that the current user can access such as procedures, functions, package bodies, or triggers.

- Columns

| Column Name | Data Type     | Description                                                                                                                                                         |
|-------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| OWNER       | VARCHAR(128)  | Owner of the object                                                                                                                                                 |
| NAME        | VARCHAR(128)  | Name of the object                                                                                                                                                  |
| TYPE        | VARCHAR(12)   | Type of the object<br><br>- PROCEDURE<br><br>- FUNCTION<br><br>- PACKAGE<br><br>- PACKAGE BODY<br><br>- TRIGGER<br><br>- TYPE<br><br>- TYPE BODY<br><br>- UNDEFINED |
| LINE        | NUMBER        | Line number of the source                                                                                                                                           |
| TEXT        | VARCHAR(4000) | Source text                                                                                                                                                         |

- See also

[DBA\\_SOURCE](#)

[USER\\_SOURCE](#)

### 3.2.73. ALL\_STAT\_EXTENSIONS

Displays information of all optimizer statistics extensions that the current user can access.

- Columns

| Column Name    | Data Type    | Description                                      |
|----------------|--------------|--------------------------------------------------|
| OWNER          | VARCHAR(128) | Owner of the extension                           |
| TABLE_NAME     | VARCHAR(128) | Name of the table to which the extension belongs |
| EXTENSION_NAME | VARCHAR(128) | Name of the statistics extension                 |
| EXTENSION      | LONG         | Extension (the expression)                       |

- See also

[DBA\\_STAT\\_EXTENSIONS](#)

[USER\\_STAT\\_EXTENSIONS](#)

### 3.2.74. ALL\_SUBPARTITION\_TEMPLATES

displays all subpartition templates that the current user can access.

- Columns

| Column Name           | Data Type   | Description                             |
|-----------------------|-------------|-----------------------------------------|
| USER_NAME             | VARCHAR(30) | name of the user                        |
| TABLE_NAME            | VARCHAR(30) | name of the table                       |
| SUBPARTITION_NAME     | VARCHAR(34) | name of subpartition                    |
| SUBPARTITION_POSITION | NUMBER      | position of the subpartition            |
| TABLESPACE_NAME       | VARCHAR(30) | tablespace name of the subpartition     |
| HIGH_BOUND            | LONG        | literal list values of the subpartition |

- See also

[DBA\\_SUBPARTITION\\_TEMPLATES](#)

### 3.2.75. ALL\_SUBPART\_KEY\_COLUMNS

Displays subpartitioning key columns of the partitioned tables or indexes that the current user can access.

- Columns

| Column Name     | Data Type    | Description                                           |
|-----------------|--------------|-------------------------------------------------------|
| OWNER           | VARCHAR(128) | Owner of the partitioned table or index               |
| NAME            | VARCHAR(128) | Name of the partitioned table or index                |
| OBJECT_TYPE     | VARCHAR(5)   | Type of the object<br>- TABLE<br>- INDEX              |
| COLUMN_NAME     | VARCHAR(128) | Name of the column                                    |
| COLUMN_POSITION | NUMBER       | Position of the column within the subpartitioning key |

- See also

[DBA\\_SUBPART\\_KEY\\_COLUMNS](#)

[USER\\_SUBPART\\_KEY\\_COLUMNS](#)

### 3.2.76. ALL\_SYNONYMS

Displays all synonyms that the current user can access.

- Columns

| Column Name      | Data Type    | Description                                     |
|------------------|--------------|-------------------------------------------------|
| OWNER            | VARCHAR(128) | Owner of the synonym                            |
| SYNONYM_NAME     | VARCHAR(128) | Name of the synonym                             |
| ORG_OBJECT_OWNER | VARCHAR(128) | Owner of the object to which the synonym refers |
| ORG_OBJECT_NAME  | VARCHAR(257) | Name of the object to which the synonym refers  |

- See also

[DBA\\_SYNONYMS](#)

[USER\\_SYNONYMS](#)

## 3.2.77. ALL\_TABLES

Displays all tables that the current user can access.

- Columns

| Column Name     | Data Type    | Description                                                      |
|-----------------|--------------|------------------------------------------------------------------|
| OWNER           | VARCHAR(128) | Owner of the table                                               |
| TABLE_NAME      | VARCHAR(128) | Name of the table                                                |
| TABLESPACE_NAME | VARCHAR(128) | Name of the tablespace that contains the table                   |
| PCT_FREE        | NUMBER       | Value of the PCTFREE parameter                                   |
| INI_TRANS       | NUMBER       | Value of the INITRANS parameter                                  |
| LOGGING         | VARCHAR(3)   | Whether logging is enabled or not<br>- YES<br>- NO               |
| NUM_ROWS        | NUMBER       | The number of rows within the table                              |
| BLOCKS          | NUMBER       | The number of data blocks containing information about the table |
| AVG_ROW_LEN     | NUMBER       | Average length of rows within the table (in bytes)               |
| DEGREE          | NUMBER       | The number of threads used for table scanning                    |
| SAMPLE_SIZE     | NUMBER       | Size of the sample used for analyzing statistics for the table   |
| LAST_ANALYZED   | DATE         | Date of the most recent analysis for the table                   |
| PARTITIONED     | VARCHAR(3)   | Whether partitioning is used or not<br>- YES                     |

| Column Name  | Data Type   | Description                                                                                                                                           |
|--------------|-------------|-------------------------------------------------------------------------------------------------------------------------------------------------------|
|              |             | - NO                                                                                                                                                  |
| COMPRESS_FOR | VARCHAR(12) | Compression level of the partition<br>- QUERY LOW<br>- QUERY HIGH<br>- ARCHIVE LOW<br>- ARCHIVE HIGH<br>- ADVANCED<br>- BASIC<br>- NULL               |
| BUFFER_POOL  | VARCHAR(7)  | Type of the buffer pool<br>- DEFAULT<br>- KEEP<br>- RECYCLE                                                                                           |
| ROW_MOVEMENT | VARCHAR(8)  | Whether or not partitioned row movement is enabled<br>- ENABLED<br>- DISABLED                                                                         |
| DURATION     | VARCHAR(11) | Retention period of the temporary table<br>- SESSION: Rows are preserved during the session period<br>- TRANSACTION: Rows are deleted at every commit |
| COMPRESSION  | VARCHAR(3)  | Whether or not the table has been compressed<br>- YES<br>- NO                                                                                         |
| COMPRESS_FOR | VARCHAR(12) | Compression level of the table<br>- QUERY LOW<br>- QUERY HIGH<br>- ARCHIVE LOW                                                                        |

| Column Name    | Data Type   | Description                                                                                                                                                                                                                                       |
|----------------|-------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                |             | <ul style="list-style-type: none"> <li>- ARCHIVE HIGH</li> <li>- ADVANCED</li> <li>- NULL</li> </ul>                                                                                                                                              |
| DROPPED        | VARCHAR(3)  | Whether or not the table has been dropped <ul style="list-style-type: none"> <li>- YES</li> <li>- NO</li> </ul>                                                                                                                                   |
| READ_ONLY      | VARCHAR(3)  | Whether or not the table is in read-only mode <ul style="list-style-type: none"> <li>- YES</li> <li>- NO</li> </ul>                                                                                                                               |
| TEMPORARY      | VARCHAR(3)  | Whether or not the table is temporary <ul style="list-style-type: none"> <li>- YES</li> <li>- NO</li> </ul>                                                                                                                                       |
| IOT_TYPE       | VARCHAR(12) | If the table is an index-organized table, the IOT_TYPE is IOT or IOT_OVERFLOW. If the table is not an index-organized table, the IOT_TYPE is NULL <ul style="list-style-type: none"> <li>- IOT</li> <li>- IOT_OVERFLOW</li> <li>- NULL</li> </ul> |
| MAX_EXTENTS    | NUMBER      | Maximum number of extents that the table can use                                                                                                                                                                                                  |
| INITIAL_EXTENT | NUMBER      | Size in bytes of the initial extent                                                                                                                                                                                                               |
| NEXT_EXTENT    | NUMBER      | Size in bytes of the next extent to be allocated to the table                                                                                                                                                                                     |
| MIN_EXTENTS    | NUMBER      | Minimum number of extents that the table can use                                                                                                                                                                                                  |
| IS_VIRTUAL     | VARCHAR(1)  | Whether or not the table is virtual                                                                                                                                                                                                               |
| INMEMORY       | VARCHAR(8)  | Whether or not the table is inmemory <ul style="list-style-type: none"> <li>- ENABLED</li> <li>- DISABLED</li> </ul>                                                                                                                              |

| Column Name          | Data Type   | Description                                                                                                                        |
|----------------------|-------------|------------------------------------------------------------------------------------------------------------------------------------|
|                      |             | - NULL                                                                                                                             |
| INMEMORY_PRIORITY    | VARCHAR(8)  | Inmemory Priority level of the table<br>- NONE : DEFAULT<br>- LOW<br>- MEDIUM<br>- HIGH<br>- CRITICAL<br>- NULL                    |
| INMEMORY_DISTRIBUTE  | VARCHAR(15) | Inmemory Distribution method of the table<br>- AUTO : DEFAULT<br>- BY ROWID RANGE<br>- BY PARTITION<br>- BY SUBPARTITION<br>- NULL |
| INMEMORY_COMPRESSION | VARCHAR(14) | Inmemory Compression level of the table<br>- NO MEMCOMPRESS : DEFAULT<br>- FOR QUERY LOW<br>- FOR QUERY HIGH<br>- NULL             |
| INMEMORY_DUPLICATE   | VARCHAR(13) | Inmemory Duplicate method of the table<br>- NO DUPLICATE : DEFAULT<br>- DUPLICATE<br>- DUPLICATE ALL<br>- NULL                     |

- See also

[DBA\\_TABLES](#)

### 3.2.78. ALL\_TAB\_COMMENTS

Displays comments on the tables that the current user can access.

- Columns

| Column Name | Data Type     | Description           |
|-------------|---------------|-----------------------|
| OWNER       | VARCHAR(128)  | Owner of the object   |
| TABLE_NAME  | VARCHAR(128)  | Name of the object    |
| TABLE_TYPE  | VARCHAR(9)    | Type of the object    |
| COMMENTS    | VARCHAR(4000) | Comment on the object |

- See also

[DBA\\_TAB\\_COMMENTS](#)

### 3.2.79. ALL\_TAB\_HISTGRM\_PENDING\_STATS

Displays histogram statistics of columns in pending state current user can access.

- Columns

| Column Name       | Data Type    | Description                                                    |
|-------------------|--------------|----------------------------------------------------------------|
| OWNER             | VARCHAR(128) | Owner of the table                                             |
| TABLE_NAME        | VARCHAR(128) | Name of the table                                              |
| PARTITION_NAME    | VARCHAR(128) | Name of the partition                                          |
| SUBPARTITION_NAME | VARCHAR(128) | Name of the subpartition                                       |
| COLUMN_NAME       | VARCHAR(128) | Name of the column                                             |
| ENDPOINT_NUMBER   | NUMBER       | The position of the endpoint in the histogram                  |
| ENDPOINT_VALUE    | NUMBER       | The normalized endpoint value for this bucket of the histogram |

| Column Name           | Data Type      | Description                                                |
|-----------------------|----------------|------------------------------------------------------------|
| ENDPOINT_ACTUAL_VALUE | VARCHAR2(4000) | The actual endpoint value for this bucket of the histogram |

- See also

[DBA\\_TAB\\_HISTGRM\\_PENDING\\_STATS](#)

[USER\\_TAB\\_HISTGRM\\_PENDING\\_STATS](#)

### 3.2.80. ALL\_TAB\_IDENTITY\_COLS

Displays identity columns of all tables that the current user can access.

- Columns

| Column Name      | Data Type    | Description                                              |
|------------------|--------------|----------------------------------------------------------|
| OWNER            | VARCHAR(128) | Owner of the table                                       |
| TABLE_NAME       | VARCHAR(128) | Name of the table                                        |
| COLUMN_NAME      | VARCHAR(128) | Name of the identity column                              |
| GENERATION_TYPE  | VARCHAR(10)  | Generation type of the identity column                   |
| SEQUENCE_NAME    | VARCHAR(128) | Name of the sequence associated with the identity column |
| IDENTITY_OPTIONS | VARCHAR(298) | Options for the identity column sequence generator       |

- See also

[DBA\\_TAB\\_IDENTITY\\_COLS](#)

[USER\\_TAB\\_IDENTITY\\_COLS](#)

### 3.2.81. ALL\_TAB\_PENDING\_STATS

Displays pending statistics for the tables that the current user can access.

- Columns

| Column Name       | Data Type    | Description                                                       |
|-------------------|--------------|-------------------------------------------------------------------|
| OWNER             | VARCHAR(128) | Owner of the object                                               |
| TABLE_NAME        | VARCHAR(128) | Name of the table                                                 |
| PARTITION_NAME    | VARCHAR(128) | Name of the table partition                                       |
| SUBPARTITION_NAME | VARCHAR(128) | Name of the table subpartition                                    |
| NUM_ROWS          | NUMBER       | The number of rows in the object                                  |
| BLOCKS            | NUMBER       | The number of data blocks containing information about the object |
| AVG_ROW_LEN       | NUMBER       | Average length of a row in the table                              |
| SAMPLE_SIZE       | NUMBER       | The number of rows sampled for statistics gathering               |
| LAST_ANALYZED     | DATE         | Date of the most recent analysis for the object                   |

- See also

[DBA\\_TAB\\_PENDING\\_STATS](#)

[USER\\_TAB\\_PENDING\\_STATS](#)

### 3.2.82. ALL\_TBL\_COLS

Displays column information of all tables and views that the current user can access.

- Columns

| Column Name    | Data Type    | Description                         |
|----------------|--------------|-------------------------------------|
| OWNER          | VARCHAR(128) | Owner of the table or view          |
| TABLE_NAME     | VARCHAR(128) | Name of the table or view           |
| COLUMN_NAME    | VARCHAR(128) | Name of the column                  |
| DATA_TYPE      | VARCHAR(128) | Data type of the column             |
| DATA_LENGTH    | NUMBER       | Column length (in bytes)            |
| DATA_PRECISION | NUMBER       | Precision level of the column value |

| Column Name          | Data Type     | Description                                                                                                                                                    |
|----------------------|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------|
| DATA_SCALE           | NUMBER        | Scale of the column value                                                                                                                                      |
| NULLABLE             | VARCHAR(1)    | Whether NULL values are allowed or not<br><br>- Y<br><br>- N (It is set to "N" if NOT NULL constraints are defined or the column is a part of the primary key) |
| COLUMN_ID            | NUMBER        | ID of the column                                                                                                                                               |
| DATA_DEFAULT         | LONG          | Default value of the column                                                                                                                                    |
| DEFAULT_LENGTH       | NUMBER        | Default column length (in bytes)                                                                                                                               |
| NUM_NULLS            | NUMBER        | (Not supported)                                                                                                                                                |
| CHAR_COL_DECL_LENGTH | NUMBER        | Maximum number of characters that can fit in the column                                                                                                        |
| CHAR_LENGTH          | NUMBER        | Column length in characters                                                                                                                                    |
| CHAR_USED            | VARCHAR(1)    | Length semantics of the column<br><br>- C<br><br>- B (It is set to "C" if the length was specified in characters, "B" if in bytes)                             |
| VIRTUAL_COLUMN       | VARCHAR(1)    | Whether the column is a virtual column                                                                                                                         |
| HIDDEN_COLUMN        | VARCHAR(1)    | (Not supported)                                                                                                                                                |
| SEGMENT_COLUMN_ID    | NUMBER        | (Not supported)                                                                                                                                                |
| INTERNAL_COLUMN_ID   | NUMBER        | (Not supported)                                                                                                                                                |
| QUALIFIED_COL_NAME   | VARCHAR(4000) | (Not supported)                                                                                                                                                |
| OBJ_ID               | NUMBER        | ID of the object                                                                                                                                               |

- See also

[DBA\\_TBL\\_COLS](#)

[USER\\_TBL\\_COLS](#)

### 3.2.83. ALL\_TBL\_COLUMNS

Displays column information of all tables and views that the current user can access.

- Columns

| Column Name              | Data Type    | Description                                                                                                                                                    |
|--------------------------|--------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------|
| OWNER                    | VARCHAR(128) | Owner of the table or view                                                                                                                                     |
| TABLE_NAME               | VARCHAR(128) | Name of the table or view                                                                                                                                      |
| COLUMN_NAME              | VARCHAR(128) | Name of the column                                                                                                                                             |
| DATA_TYPE                | VARCHAR(128) | Data type of the column                                                                                                                                        |
| DATA_TYPE_OWNER          | VARCHAR(128) | Owner of the datatype of the column                                                                                                                            |
| DATA_LENGTH              | NUMBER       | Column length (in bytes)                                                                                                                                       |
| DATA_PRECISION           | NUMBER       | Precision level of the column value                                                                                                                            |
| DATA_SCALE               | NUMBER       | Scale of the column value                                                                                                                                      |
| NULLABLE                 | VARCHAR(1)   | Whether NULL values are allowed or not<br><br>- Y<br><br>- N (It is set to "N" if NOT NULL constraints are defined or the column is a part of the primary key) |
| COLUMN_ID                | NUMBER       | ID of the column                                                                                                                                               |
| DATA_DEFAULT             | LONG         | Default value of the column                                                                                                                                    |
| DEFAULT_LENGTH           | NUMBER       | Default column length (in bytes)                                                                                                                               |
| CHAR_COL_DE<br>CL_LENGTH | NUMBER       | Maximum number of characters that can fit in the column                                                                                                        |
| CHAR_LENGTH              | NUMBER       | Column length in characters                                                                                                                                    |
| CHAR_USED                | VARCHAR(1)   | Length semantics of the column<br><br>- C<br><br>- B (It is set to "C" if the length was specified in characters, "B" if in bytes)                             |
| VIRTUAL_COLUMN           | VARCHAR(1)   | Whether the column is a virtual column                                                                                                                         |

| Column Name | Data Type | Description      |
|-------------|-----------|------------------|
| OBJ_ID      | NUMBER    | ID of the object |

- See also

[ALL\\_TBL\\_COLUMNS\\_WITH\\_DNO](#)

[DBA\\_TBL\\_COLUMNS](#)

[USER\\_TBL\\_COLUMNS](#)

### 3.2.84. ALL\_TBL\_COLUMNS\_WITH\_DNO

Displays column information of all tables and views, which the current user can access, with data type number of the column.

- Columns

| Column Name     | Data Type    | Description                                                                                                                                                    |
|-----------------|--------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------|
| OWNER           | VARCHAR(128) | Owner of the table or view                                                                                                                                     |
| TABLE_NAME      | VARCHAR(128) | Name of the table or view                                                                                                                                      |
| COLUMN_NAME     | VARCHAR(128) | Name of the column                                                                                                                                             |
| DATA_TYPE       | VARCHAR(128) | Data type of the column                                                                                                                                        |
| DATA_TYPE_NO    | NUMBER       | Data type number of the column                                                                                                                                 |
| DATA_TYPE_OWNER | VARCHAR(128) | Owner of the datatype of the column                                                                                                                            |
| DATA_LENGTH     | NUMBER       | Column length (in bytes)                                                                                                                                       |
| DATA_PRECISION  | NUMBER       | Precision level of the column value                                                                                                                            |
| DATA_SCALE      | NUMBER       | Scale of the column value                                                                                                                                      |
| NULLABLE        | VARCHAR(1)   | Whether NULL values are allowed or not<br><br>- Y<br><br>- N (It is set to "N" if NOT NULL constraints are defined or the column is a part of the primary key) |
| COLUMN_ID       | NUMBER       | ID of the column                                                                                                                                               |

| Column Name              | Data Type  | Description                                                                                                                        |
|--------------------------|------------|------------------------------------------------------------------------------------------------------------------------------------|
| DATA_DEFAULT             | LONG       | Default value of the column                                                                                                        |
| DEFAULT_LENGTH           | NUMBER     | Default column length (in bytes)                                                                                                   |
| CHAR_COL_DE<br>CL_LENGTH | NUMBER     | Maximum number of characters that can fit in the column                                                                            |
| CHAR_LENGTH              | NUMBER     | Column length in characters                                                                                                        |
| CHAR_USED                | VARCHAR(1) | Length semantics of the column<br><br>- C<br><br>- B (It is set to "C" if the length was specified in characters, "B" if in bytes) |
| VIRTUAL_COLUMN           | VARCHAR(1) | Whether the column is a virtual column                                                                                             |
| OBJ_ID                   | NUMBER     | ID of the object                                                                                                                   |

- See also

[ALL\\_TBL\\_COLUMNS](#)

[DBA\\_TBL\\_COLUMNS](#)

[USER\\_TBL\\_COLUMNS](#)

### 3.2.85. ALL\_TBL\_COL\_STATISTICS

Displays statistics and a histogram of columns by extracting them from ALL\_TBL\_COLUMNS.

- Columns

| Column Name  | Data Type     | Description                    |
|--------------|---------------|--------------------------------|
| TABLE_OWNER  | VARCHAR(128)  | Owner of the table             |
| TABLE_NAME   | VARCHAR(128)  | Name of the table              |
| COLUMN_NAME  | VARCHAR(128)  | Name of the column             |
| NUM_DISTINCT | NUMBER        | The number of distinct values  |
| LOW_VALUE    | VARCHAR(4000) | The lowest value in the column |

| Column Name   | Data Type     | Description                                                     |
|---------------|---------------|-----------------------------------------------------------------|
| HIGH_VALUE    | VARCHAR(4000) | The highest value in the column                                 |
| DENSITY       | NUMBER        | Density of the column                                           |
| NUM_NULLS     | NUMBER        | The number of NULL values                                       |
| NUM_BUCKETS   | NUMBER        | The number of buckets for the histogram                         |
| LAST_ANALYZED | DATE          | Date of the most recent analysis for the column                 |
| SAMPLE_SIZE   | NUMBER        | Size of the sample used for analyzing statistics for the column |
| AVG_COL_LEN   | NUMBER        | Average length of the column (in bytes)                         |
| HISTOGRAM     | VARCHAR(15)   | Indicates the type of histogram                                 |

- See also

[DBA\\_TBL\\_COL\\_STATISTICS](#)

[USER\\_TBL\\_COL\\_STATISTICS](#)

### 3.2.86. ALL\_TBL\_HISTOGRAMS

Displays histogram information of tables that the current user can access.

- Columns

| Column Name           | Data Type     | Description                                                      |
|-----------------------|---------------|------------------------------------------------------------------|
| OWNER                 | VARCHAR(128)  | Owner of the table                                               |
| TABLE_NAME            | VARCHAR(128)  | Name of the table                                                |
| COLUMN_NAME           | VARCHAR(128)  | Name of the column                                               |
| ENDPOINT_NUMBER       | NUMBER        | History bucket number                                            |
| ENDPOINT_VALUE        | NUMBER        | End point value of the current bucket                            |
| ENDPOINT_ACTUAL_VALUE | VARCHAR(4000) | Actual string value of the end point value of the current bucket |

- See also

[DBA\\_TBL\\_HISTOGRAMS](#)

[USER\\_TBL\\_HISTOGRAMS](#)

### 3.2.87. ALL\_TBL\_PARTITIONS

Displays partition information of all tables that the current user can access.

- Columns

| Column Name        | Data Type    | Description                                                                                    |
|--------------------|--------------|------------------------------------------------------------------------------------------------|
| OWNER              | VARCHAR(128) | Name of the user                                                                               |
| TABLE_NAME         | VARCHAR(128) | Name of the table                                                                              |
| COMPOSITE          | VARCHAR(3)   | Whether or not the partition belongs to a composite-partitioned table<br><br>- YES<br><br>- NO |
| PARTITION_NAME     | VARCHAR(128) | Name of the table partition                                                                    |
| SUBPARTITION_COUNT | NUMBER       | The number of subpartitions which belong to the partition                                      |
| BOUND              | LONG         | Partition bound value                                                                          |
| PARTITION_NO       | NUMBER       | Partition number within the table                                                              |
| TABLESPACE_NAME    | VARCHAR(128) | Name of the tablespace to which the partition belongs                                          |
| PCT_FREE           | NUMBER       | Value of the PCTFREE parameter                                                                 |
| INI_TRANS          | NUMBER       | Value of the INITRANS parameter                                                                |
| NEXT_EXTENT        | NUMBER       | Size of the next extent                                                                        |
| MAX_EXTENTS        | NUMBER       | Value of the MAX EXTENTS parameter                                                             |
| COMPRESSION        | VARCHAR(3)   | Whether or not the table partition has been compressed<br><br>- YES<br><br>- NO                |

| Column Name       | Data Type   | Description                                                                                                                                       |
|-------------------|-------------|---------------------------------------------------------------------------------------------------------------------------------------------------|
| COMPRESS_FOR      | VARCHAR(12) | Compression level of the partition<br><br>- QUERY LOW<br><br>- QUERY HIGH<br><br>- ARCHIVE LOW<br><br>- ARCHIVE HIGH<br><br>- NULL                |
| BUFFER_POOL       | VARCHAR(7)  | Type of the buffer pool<br><br>- DEFAULT<br><br>- KEEP<br><br>- RECYCLE                                                                           |
| NUM_ROWS          | NUMBER      | The number of rows within the table partition                                                                                                     |
| BLOCKS            | NUMBER      | The number of data blocks containing information about the table partition                                                                        |
| LAST_ANALYZED     | DATE        | Date of the most recent analysis for the table partition                                                                                          |
| LOGGING           | VARCHAR(3)  | Whether logging is enabled or not<br><br>- YES<br><br>- NO                                                                                        |
| INMEMORY          | VARCHAR(8)  | Whether or not the table partition is inmemory<br><br>- ENABLED<br><br>- DISABLED                                                                 |
| INMEMORY_PRIORITY | VARCHAR(8)  | Inmemory Priority level of the table partition<br><br>- NONE : DEFAULT<br><br>- LOW<br><br>- MEDIUM<br><br>- HIGH<br><br>- CRITICAL<br><br>- NULL |

| Column Name          | Data Type   | Description                                                                                                                                  |
|----------------------|-------------|----------------------------------------------------------------------------------------------------------------------------------------------|
| INMEMORY_DISTRIBUTE  | VARCHAR(15) | Inmemory Distribution method of the table partition<br>- AUTO : DEFAULT<br>- BY ROWID RANGE<br>- BY PARTITION<br>- BY SUBPARTITION<br>- NULL |
| INMEMORY_COMPRESSION | VARCHAR(14) | Inmemory Compression level of the table partition<br>- NO MEMCOMPRESS : DEFAULT<br>- FOR QUERY LOW<br>- FOR QUERY HIGH<br>- NULL             |
| INMEMORY_DUPLICATE   | VARCHAR(13) | Inmemory Duplicate method of the table partition<br>- NO DUPLICATE : DEFAULT<br>- DUPLICATE<br>- DUPLICATE ALL<br>- NULL                     |

- See also

[DBA\\_TBL\\_PARTITIONS](#)

[USER\\_TBL\\_PARTITIONS](#)

### 3.2.88. ALL\_TBL\_PRIVS

Displays tables for which the current user or public user is granted object privileges.

- Columns

| Column Name | Data Type    | Description                                        |
|-------------|--------------|----------------------------------------------------|
| GRANTEE     | VARCHAR(128) | Name of the user to whom the privilege was granted |
| OWNER       | VARCHAR(128) | Owner of the object                                |

| Column Name | Data Type    | Description                                                                         |
|-------------|--------------|-------------------------------------------------------------------------------------|
| TABLE_NAME  | VARCHAR(128) | Name of the object                                                                  |
| GRANTOR     | VARCHAR(128) | Name of the user who granted the privilege                                          |
| PRIVILEGE   | VARCHAR(40)  | Name of the privilege to the table                                                  |
| GRANTABLE   | VARCHAR(3)   | Whether or not the privilege was granted with GRANT OPTION<br><br>- YES<br><br>- NO |
| TYPE        | VARCHAR(23)  | Type of the object                                                                  |

- See also

[DBA\\_TBL\\_PRIVS](#)

[USER\\_TBL\\_PRIVS](#)

### 3.2.89. ALL\_TBL\_PRIVS\_MADE

Displays object privileges granted on tables for which the current user or public user is the object owner or grantor.

- Columns

| Column Name | Data Type    | Description                                                                         |
|-------------|--------------|-------------------------------------------------------------------------------------|
| GRANTEE     | VARCHAR(128) | Name of the user to whom the privilege was granted                                  |
| OWNER       | VARCHAR(128) | Name of the object owner                                                            |
| TABLE_NAME  | VARCHAR(128) | Name of the object                                                                  |
| GRANTOR     | VARCHAR(128) | Name of the user who granted the privilege                                          |
| PRIVILEGE   | VARCHAR(40)  | Name of the privilege to the table                                                  |
| GRANTABLE   | VARCHAR(3)   | Whether or not the privilege was granted with ADMIN OPTION<br><br>- YES<br><br>- NO |

| Column Name | Data Type   | Description        |
|-------------|-------------|--------------------|
| TYPE        | VARCHAR(23) | Type of the object |

- See also

[USER\\_TBL\\_PRIVS\\_MADE](#)

### 3.2.90. ALL\_TBL\_PRIVS\_REC'D

Displays object privileges granted on tables for which the current user or public user is the object grantee.

- Columns

| Column Name | Data Type    | Description                                                                         |
|-------------|--------------|-------------------------------------------------------------------------------------|
| GRANTEE     | VARCHAR(128) | Name of the user to whom the privilege was granted                                  |
| OWNER       | VARCHAR(128) | Name of the object owner                                                            |
| TABLE_NAME  | VARCHAR(128) | Name of the object                                                                  |
| GRANTOR     | VARCHAR(128) | Name of the user who granted the privilege                                          |
| PRIVILEGE   | VARCHAR(40)  | Name of the privilege to the table                                                  |
| GRANTABLE   | VARCHAR(3)   | Whether or not the privilege was granted with ADMIN OPTION<br><br>- YES<br><br>- NO |
| TYPE        | VARCHAR(23)  | Type of the object                                                                  |

### 3.2.91. ALL\_TBL\_SUBPARTITIONS

Displays subpartitions of all tables that the current user can access.

- Columns

| Column Name | Data Type    | Description      |
|-------------|--------------|------------------|
| OWNER       | VARCHAR(128) | Name of the user |

| Column Name       | Data Type    | Description                                                                                                                           |
|-------------------|--------------|---------------------------------------------------------------------------------------------------------------------------------------|
| TABLE_NAME        | VARCHAR(128) | Name of the table                                                                                                                     |
| PARTITION_NAME    | VARCHAR(128) | Name of the table partition                                                                                                           |
| SUBPARTITION_NAME | VARCHAR(128) | Name of the table subpartition                                                                                                        |
| BOUND             | LONG         | Subpartition bound value                                                                                                              |
| SUBPARTITION_NO   | NUMBER       | Subpartition number within the partition                                                                                              |
| TABLESPACE_NAME   | VARCHAR(128) | Name of the tablespace to which the subpartition belongs                                                                              |
| PCT_FREE          | NUMBER       | Value of the PCTFREE parameter                                                                                                        |
| INI_TRANS         | NUMBER       | Value of the INITRANS parameter                                                                                                       |
| NEXT_EXTENT       | NUMBER       | Size of the next extent                                                                                                               |
| MAX_EXTENTS       | NUMBER       | Value of the MAX EXTENTS parameter                                                                                                    |
| COMPRESSION       | VARCHAR(3)   | Whether or not the table partition has been compressed<br>- YES<br>- NO                                                               |
| COMPRESS_FOR      | VARCHAR(12)  | Compression level of the table subpartition<br>- QUERY LOW<br>- QUERY HIGH<br>- ARCHIVE LOW<br>- ARCHIVE HIGH<br>- ADVANCED<br>- NULL |
| BUFFER_POOL       | VARCHAR(7)   | Type of the buffer pool<br>- DEFAULT<br>- KEEP<br>- RECYCLE                                                                           |

| Column Name          | Data Type   | Description                                                                                                                                                         |
|----------------------|-------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| MAX_TRANS            | NUMBER      | Maximum number of transactions allowed                                                                                                                              |
| INITIAL_EXTENT       | NUMBER      | Size of the initial extent                                                                                                                                          |
| MIN_EXTENT           | NUMBER      | Minimum number of extents allowed in the segment                                                                                                                    |
| LOGGING              | VARCHAR(3)  | Whether logging is enabled or not<br><br>- YES<br><br>- NO                                                                                                          |
| NUM_ROWS             | NUMBER      | The number of rows within the table subpartition                                                                                                                    |
| LAST_ANALYZED        | DATE        | Date of the most recent analysis for the table subpartition                                                                                                         |
| INMEMORY             | VARCHAR(8)  | Whether or not the table subpartition is inmemory<br><br>- ENABLED<br><br>- DISABLED                                                                                |
| INMEMORY_PRIORITY    | VARCHAR(8)  | Inmemory Priority level of the table subpartition<br><br>- NONE : DEFAULT<br><br>- LOW<br><br>- MEDIUM<br><br>- HIGH<br><br>- CRITICAL<br><br>- NULL                |
| INMEMORY_DISTRIBUTE  | VARCHAR(15) | Inmemory Distribution method of the table subpartition<br><br>- AUTO : DEFAULT<br><br>- BY ROWID RANGE<br><br>- BY PARTITION<br><br>- BY SUBPARTITION<br><br>- NULL |
| INMEMORY_COMPRESSION | VARCHAR(14) | Inmemory Compression level of the table subpartition<br><br>- NO MEMCOMPRESS : DEFAULT                                                                              |

| Column Name        | Data Type   | Description                                                                                                                 |
|--------------------|-------------|-----------------------------------------------------------------------------------------------------------------------------|
|                    |             | - FOR QUERY LOW<br>- FOR QUERY HIGH<br>- NULL                                                                               |
| INMEMORY_DUPLICATE | VARCHAR(13) | Inmemory Duplicate method of the table subpartition<br>- NO DUPLICATE : DEFAULT<br>- DUPLICATE<br>- DUPLICATE ALL<br>- NULL |

- See also

[DBA\\_TBL\\_SUBPARTITIONS](#)

## 3.2.92. ALL\_TRIGGERS

Displays all triggers that the current user can access.

- Columns

| Column Name      | Data Type    | Description                                                                                                     |
|------------------|--------------|-----------------------------------------------------------------------------------------------------------------|
| OWNER            | VARCHAR(128) | Owner of the trigger                                                                                            |
| TRIGGER_NAME     | VARCHAR(128) | Name of the trigger                                                                                             |
| TRIGGER_TYPE     | VARCHAR(16)  | When the trigger is enabled<br>- BEFORE STATEMENT<br>- BEFORE EACH ROW<br>- AFTER STATEMENT<br>- AFTER EACH ROW |
| TRIGGERING_EVENT | VARCHAR(52)  | Event that activates the trigger                                                                                |
| TABLE_OWNER      | VARCHAR(128) | Owner of the table where the trigger is defined                                                                 |
| TABLE_NAME       | VARCHAR(128) | Name of the table where the trigger is defined                                                                  |

| Column Name           | Data Type    | Description                                                        |
|-----------------------|--------------|--------------------------------------------------------------------|
| REFERENC<br>ING_NAMES | VARCHAR(422) | Name to which the trigger refers to read OLD and NEW column values |
| WHEN_CLAUSE           | LONG         | Conditions that should be evaluated before the trigger is enabled  |
| STATUS                | VARCHAR(8)   | Whether or not the trigger is enabled                              |

- See also

[DBA\\_TRIGGERS](#)

[USER\\_TRIGGERS](#)

### 3.2.93. ALL\_TRIGGER\_COLS

Displays column information of all triggers that the current user can access and all triggers on tables that the current user can access.

- Columns

| Column Name   | Data Type    | Description                                                    |
|---------------|--------------|----------------------------------------------------------------|
| TRIGGER_OWNER | VARCHAR(128) | Owner of the trigger                                           |
| TRIGGER_NAME  | VARCHAR(128) | Name of the trigger                                            |
| TABLE_OWNER   | VARCHAR(128) | Owner of the table                                             |
| TABLE_NAME    | VARCHAR(128) | Name of the table                                              |
| COLUMN_NAME   | VARCHAR(128) | Name of the column                                             |
| COLUMN_LIST   | VARCHAR(3)   | Whether or not the column is used in UPDATE OF clause (Yes/No) |
| COLUMN_USAGE  | VARCHAR(17)  | Column usage in the trigger                                    |

- See also

[DBA\\_TRIGGER\\_COLS](#)

[USER\\_TRIGGER\\_COLS](#)

## 3.2.94. ALL\_TYPES

Displays all types that the current user can access.

- Columns

| Column Name      | Data Type    | Description                                               |
|------------------|--------------|-----------------------------------------------------------|
| OWNER            | VARCHAR(128) | Owner of this type                                        |
| TYPE_NAME        | VARCHAR(128) | Name of this type                                         |
| TYPE_OID         | RAW(16)      | Object identifier (OID) of this type                      |
| TYPECODE         | VARCHAR(10)  | Typecode of this type                                     |
| ATTRIBUTES       | NUMBER       | The number of attributes for this type                    |
| METHODS          | NUMBER       | The number of methods for this type                       |
| PREDEFINED       | VARCHAR(3)   | Indicates whether this type is predefined (Not supported) |
| INCOMPLETE       | VARCHAR(3)   | Indicates whether this type is incomplete (Not supported) |
| FINAL            | VARCHAR(3)   | Indicates whether this type is final                      |
| INSTANTIABLE     | VARCHAR(3)   | Indicates whether this type is instantiable               |
| SUPERTYPE_OWNER  | VARCHAR(128) | Owner of this supertype                                   |
| SUPERTYPE_NAME   | VARCHAR(128) | Name of this supertype                                    |
| LOCAL_ATTRIBUTES | VARCHAR(256) | The number of local attributes                            |
| LOCAL_METHODS    | VARCHAR(256) | The number of the local methods                           |
| TYPEID           | RAW(8)       | TYPEID of this type                                       |

- See also

[DBA\\_TYPES](#)

[USER\\_TYPES](#)

### 3.2.95. ALL\_TYPE\_ATTRS

Displays all the attributes that the current user can access.

- Columns

| Column Name        | Data Type    | Description                                     |
|--------------------|--------------|-------------------------------------------------|
| OWNER              | VARCHAR(128) | Owner of this type                              |
| TYPE_NAME          | VARCHAR(128) | Name of this type                               |
| ATTR_NAME          | VARCHAR(128) | Name of this attribute                          |
| ATTR_TYPE_MOD      | VARCHAR(7)   | Modifier of this attribute type (not supported) |
| ATTR_TYPE_OWNER    | VARCHAR(128) | Owner of this attribute type                    |
| ATTR_TYPE_NAME     | VARCHAR(128) | Name of this attribute type                     |
| LENGTH             | NUMBER       | Length of this type (not supported)             |
| PRECISION          | NUMBER       | Precision of this type                          |
| SCALE              | NUMBER       | Scale of this type                              |
| CHARACTER_SET_NAME | VARCHAR(128) | Name of the Character set (not supported)       |
| ATTR_NO            | NUMBER       | Attribute number                                |
| INHERITED          | VARCHAR(3)   | Indicates whether this is inherited             |

- See also

[DBA\\_TYPE\\_ATTRS](#)

[USER\\_TYPE\\_ATTRS](#)

### 3.2.96. ALL\_TYPE\_METHODS

Displays all the methods of this type that the current user can access.

- Columns

| Column Name | Data Type    | Description        |
|-------------|--------------|--------------------|
| OWNER       | VARCHAR(128) | Owner of this type |

| Column Name  | Data Type    | Description                                                                     |
|--------------|--------------|---------------------------------------------------------------------------------|
| TYPE_NAME    | VARCHAR(128) | Name of this type                                                               |
| METHOD_NAME  | VARCHAR(128) | Name of this method                                                             |
| METHOD_NO    | NUMBER       | Method number for this method (not supported)                                   |
| METHOD_TYPE  | VARCHAR(11)  | Type of this method                                                             |
| PARAMETERS   | NUMBER       | The number of parameters                                                        |
| RESULTS      | VARCHAR(3)   | Indicates whether this method has results to return.                            |
| FINAL        | VARCHAR(3)   | Indicates whether this method is final. (not supported)                         |
| INSTANTIABLE | VARCHAR(3)   | Indicates whether this method is instantiable. (not supported)                  |
| OVERRIDING   | VARCHAR(3)   | Indicates whether this method is overriding a supertype method. (not supported) |
| INHERITED    | VARCHAR(3)   | Indicates whether this is inherited from a supertype method. (not supported)    |
| STATIC       | VARCHAR(3)   | Indicates whether this method is static.                                        |
| CONSTRUCTOR  | VARCHAR(3)   | Indicates whether this method is constructor.                                   |

- See also

[DBA\\_TYPE\\_METHODS](#)

[USER\\_TYPE\\_METHODS](#)

### 3.2.97. ALL\_UNUSED\_COL\_TABS

Displays all unused columns that the current user can access.

- Columns

| Column Name | Data Type     | Description                           |
|-------------|---------------|---------------------------------------|
| OWNER       | VARCHAR2(128) | Owner of the table of unused columns. |
| TABLE_NAME  | VARCHAR2(128) | Name of the table                     |

| Column Name | Data Type | Description              |
|-------------|-----------|--------------------------|
| COUNT       | NUMBER    | Number of unused columns |

- See also

[DBA\\_UNUSED\\_COL\\_TABS](#)

[USER\\_UNUSED\\_COL\\_TABS](#)

### 3.2.98. ALL\_UPDATABLE\_COLUMNS

Displays all join view columns that the current user can update.

- Columns

| Column Name | Data Type    | Description                                              |
|-------------|--------------|----------------------------------------------------------|
| OWNER       | VARCHAR(128) | Owner of the table                                       |
| TABLE_NAME  | VARCHAR(128) | Name of the table                                        |
| COLUMN_NAME | VARCHAR(128) | Name of the column                                       |
| UPDATABLE   | VARCHAR(3)   | Whether or not the column is updatable<br>- YES<br>- NO  |
| INSERTABLE  | VARCHAR(3)   | Whether or not the column is insertable<br>- YES<br>- NO |
| DELETABLE   | VARCHAR(3)   | Whether or not column is deletable<br>- YES<br>- NO      |

- See also

[DBA\\_UPDATABLE\\_COLUMNS](#)

[USER\\_UPDATABLE\\_COLUMNS](#)

### 3.2.99. ALL\_USERS

Displays all users that the current user can view.

- Columns

| Column Name | Data Type    | Description                    |
|-------------|--------------|--------------------------------|
| USERNAME    | VARCHAR(128) | Name of the user               |
| USER_ID     | NUMBER       | ID of the user                 |
| CREATED     | DATE         | Date that the user was created |

- See also

[DBA\\_USERS](#)

[USER\\_USERS](#)

### 3.2.100. ALL\_VIEWS

Displays all views that the current user can access.

- Columns

| Column Name | Data Type    | Description            |
|-------------|--------------|------------------------|
| OWNER       | VARCHAR(128) | Owner of the view      |
| VIEW_NAME   | VARCHAR(128) | Name of the view       |
| TEXT        | LONG         | Definition of the view |

- See also

[ALL\\_VIEWS\\_INTERNAL](#)

[DBA\\_VIEWS](#)

[USER\\_VIEWS](#)

### 3.2.101. ALL\_VIEWS\_INTERNAL

Displays all views that the current user can access. (including views that can be accessed by the roles)

- Columns

| Column Name | Data Type    | Description       |
|-------------|--------------|-------------------|
| OWNER       | VARCHAR(128) | Owner of the view |
| VIEW_NAME   | VARCHAR(128) | Name of the view  |
| OBJ_ID      | NUMBER       | ID of the view    |

- See also

[ALL\\_VIEWS](#)

[DBA\\_VIEWS](#)

[USER\\_VIEWS](#)

### 3.2.102. CTX\_PREFERENCES

Displays preferences created by Tiberio users, as well as all the system-defined preferences.

- Columns

| Column Name | Data Type    | Description                  |
|-------------|--------------|------------------------------|
| PRE_OWNER   | VARCHAR(128) | Username of preference owner |
| PRE_NAME    | VARCHAR(30)  | Preference name              |
| PRE_CLASS   | VARCHAR(30)  | Preference class             |
| PRE_OBJECT  | VARCHAR(30)  | Object used                  |

### 3.2.103. CTX\_PREFERENCE\_VALUES

Displays the values assigned to all the preferences in the Text data dictionary.

- Columns

| Column Name    | Data Type    | Description                  |
|----------------|--------------|------------------------------|
| PRV_OWNER      | VARCHAR(128) | Username of preference owner |
| PRV_PREFERENCE | VARCHAR(30)  | Preference name              |

| Column Name   | Data Type    | Description     |
|---------------|--------------|-----------------|
| PRV_ATTRIBUTE | VARCHAR(30)  | Attribute name  |
| PRV_VALUE     | VARCHAR(500) | Attribute value |

### 3.2.104. CTX\_STOPLISTS

Displays stoplists

- Columns

| Column Name | Data Type    | Description         |
|-------------|--------------|---------------------|
| SPL_OWNER   | VARCHAR(128) | Owner of stoplist   |
| SPL_NAME    | VARCHAR(30)  | Name of stoplist    |
| SPL_COUNT   | NUMBER       | Number of stopwords |
| SPL_TYPE    | VARCHAR(30)  | Type of stoplist    |

### 3.2.105. DATABASE\_PROPERTIES

Displays database properties.

- Columns

| Column Name | Data Type     | Description                 |
|-------------|---------------|-----------------------------|
| NAME        | VARCHAR(128)  | Name of the property        |
| VALUE       | VARCHAR(80)   | Value of the property       |
| COMMENT_STR | VARCHAR(4000) | Description of the property |

### 3.2.106. DBA\_2PC\_NEIGHBORS

Displays connection information for pending transactions.

- Columns

| Column Name   | Data Type      | Description                                                                                                        |
|---------------|----------------|--------------------------------------------------------------------------------------------------------------------|
| LOCAL_TRAN_ID | VARCHAR(65532) | ID of the transaction                                                                                              |
| IN_OUT        | NUMBER         | Connection status<br><br>- IN: Incoming connection<br><br>- OUT: Outgoing connection                               |
| DATABASE      | VARCHAR(128)   | Name of the database<br><br>- IN: Client database name<br><br>- OUT: Database link name                            |
| DBUSER_OWNER  | VARCHAR(128)   | Name of the owner<br><br>- IN: Name of the user who accesses the database<br><br>- OUT: Owner of the database link |
| INTERFACE     | VARCHAR(1)     | Type of the interface<br><br>- C: Commit point site<br><br>- N: Non-commit point site                              |

### 3.2.107. DBA\_2PC\_PENDING

Displays pending transaction information.

- Columns

| Column Name   | Data Type      | Description                  |
|---------------|----------------|------------------------------|
| LOCAL_TRAN_ID | VARCHAR(65532) | ID of the transaction        |
| XID           | VARCHAR(550)   | XID                          |
| STATUS        | VARCHAR(15)    | Status of the XA transaction |

### 3.2.108. DBA\_AUDIT\_POLICIES

Displays all fine-grained auditing policies in the database.

- Columns

| Column Name   | Data Type     | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|---------------|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| OBJECT_SCHEMA | VARCHAR(128)  | Owner of the table or view                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| OBJECT_NAME   | VARCHAR(128)  | Name of the table or view                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| POLICY_OWNER  | VARCHAR(128)  | Owner of the policy                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| POLICY_NAME   | VARCHAR(128)  | Name of the policy                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| POLICY_TEXT   | VARCHAR(4000) | Audit condition                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| POLICY_COLUMN | VARCHAR(128)  | Relevant column                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| PF_SCHEMA     | VARCHAR(128)  | Owner of the audit handler function                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| PF_PACKAGE    | VARCHAR(128)  | Name of the package containing the audit handler function                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| PF_FUNCTION   | VARCHAR(128)  | Name of the audit handler function                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| ENABLED       | VARCHAR(3)    | Indicates whether the policy is enabled (YES) or disabled (NO)                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| SEL           | VARCHAR(3)    | Indicates whether the policy is applied to queries on the object (YES) or not (NO)                                                                                                                                                                                                                                                                                                                                                                                                                 |
| INS           | VARCHAR(3)    | Indicates whether the policy is applied to INSERT statements on the object (YES) or not (NO)                                                                                                                                                                                                                                                                                                                                                                                                       |
| UPD           | VARCHAR(3)    | Indicates whether the policy is applied to UPDATE statements on the object (YES) or not (NO)                                                                                                                                                                                                                                                                                                                                                                                                       |
| DEL           | VARCHAR(3)    | Indicates whether the policy is applied to DELETE statements on the object (YES) or not (NO)                                                                                                                                                                                                                                                                                                                                                                                                       |
| AUDIT_TRAIL   | VARCHAR(12)   | Indicates the audit trail to which the audit records generated by this audit policy will be written: 1. DB - Audit records are written to DBA_FGA_AUDIT_TRAIL 2. DB+EXTENDED - Audit records are written to DBA_FGA_AUDIT_TRAIL and the SQL_TEXT and SQL_BIND columns are populated for this policy 3. XML - Audit records are written to V\$XML_AUDIT_TRAIL 4. XML+EXTENDED - Audit records are written to V\$XML_AUDIT_TRAIL and the SQL_TEXT and SQL_BIND columns are populated for this policy |

| Column Name           | Data Type   | Description                                                                                                                                                                                     |
|-----------------------|-------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| POLICY_COLUMN_OPTIONS | VARCHAR(11) | Indicates whether all columns in the AUDIT_COLUMN parameter (ALL_COLUMNS) or any of the columns in the AUDIT_COLUMN parameter (ANY_COLUMNS) are considered for triggering fine-grained auditing |

### 3.2.109. DBA\_AUDIT\_TRAIL

Displays audit trails within the database.

- Columns

| Column Name     | Data Type      | Description                                                  |
|-----------------|----------------|--------------------------------------------------------------|
| OS_USERNAME     | VARCHAR(128)   | Operating system user name who was set to be audited         |
| USERNAME        | VARCHAR(128)   | Name of the user who was set to be audited                   |
| USERHOST        | VARCHAR(21)    | Name of the host machine from which the client has connected |
| TIMESTAMP       | VARCHAR(19)    | Audit trail-created time                                     |
| OWNER           | VARCHAR(128)   | Owner of the object associated with the audited action       |
| OBJ_NAME        | VARCHAR(128)   | Name of the object associated with the audited action        |
| SESSIONID       | NUMBER         | Session ID                                                   |
| SERIALNO        | NUMBER         | Sequence number of the session                               |
| ENTRYID         | NUMBER         | Sequence number for each audit trail in the session          |
| STATEMENTID     | NUMBER         | ID of the executed statement                                 |
| PRIV_USED       | VARCHAR(80)    | Name of the privilege used to execute the audited action     |
| CLIENT_ID       | VARCHAR(30)    | Client identifier                                            |
| INSTANCE_NUMBER | NUMBER         | Instance number of RDBMS which performed the action          |
| OS_PROCESS      | VARCHAR(16)    | Process ID of RDBMS                                          |
| TRANSACTIONID   | VARCHAR(65532) | Transaction ID that participated in the audited action       |

| Column Name | Data Type | Description               |
|-------------|-----------|---------------------------|
| TSN         | NUMBER    | TSN of the audited action |
| SQL_TEXT    | LONG      | SQL of the audited action |

- See also

[USER\\_AUDIT\\_TRAIL](#)

### 3.2.110. DBA\_CATALOG

Displays all tables, views, synonyms, and sequences in the database.

- See also

[ALL\\_CATALOG](#)

[USER\\_CATALOG](#)

### 3.2.111. DBA\_COLL\_TYPES

Displays all the collection types.

- See also

[ALL\\_COLL\\_TYPES](#)

[USER\\_COLL\\_TYPES](#)

### 3.2.112. DBA\_COL\_COMMENTS

Displays comments on all columns in the database.

- See also

[ALL\\_COL\\_COMMENTS](#)

[USER\\_COL\\_COMMENTS](#)

### 3.2.113. DBA\_COL\_PENDING\_STATS

Displays statistics and a histogram of columns in pending state.

- See also

[ALL\\_COL\\_PENDING\\_STATS](#)

[USER\\_COL\\_PENDING\\_STATS](#)

### 3.2.114. DBA\_COL\_PRIVS

Displays object privileges of all columns of the database.

- See also

[ALL\\_COL\\_PRIVS](#)

[USER\\_COL\\_PRIVS](#)

### 3.2.115. DBA\_CONSTRAINTS

Displays integrity constraints defined in the database table.

- See also

[ALL\\_CONSTRAINTS](#)

[USER\\_CONSTRAINTS](#)

### 3.2.116. DBA\_CONS\_COLUMNS

Displays the columns for which the database integrity constraint is specified.

- See also

[ALL\\_CONS\\_COLUMNS](#)

[USER\\_CONS\\_COLUMNS](#)

### 3.2.117. DBA\_DATA\_FILES

Displays data file information.

- Columns

| Column Name | Data Type    | Description           |
|-------------|--------------|-----------------------|
| FILE_NAME   | VARCHAR(256) | Name of the data file |

| Column Name     | Data Type    | Description                                                                                                                                                                                  |
|-----------------|--------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| FILE_ID         | NUMBER       | ID of the data file                                                                                                                                                                          |
| TABLESPACE_NAME | VARCHAR(128) | Name of the tablespace where the data file is contained.                                                                                                                                     |
| BYTES           | NUMBER       | Size of the data file in bytes                                                                                                                                                               |
| BLOCKS          | NUMBER       | Size of the data file in blocks                                                                                                                                                              |
| STATUS          | CHAR(9)      | File status: AVAILABLE or INVALID                                                                                                                                                            |
| RELATIVE_FNO    | NUMBER       | Relative data file number                                                                                                                                                                    |
| AUTOEXTENSIBLE  | VARCHAR(3)   | Auto extensible indicator<br><br>- YES<br><br>- NO                                                                                                                                           |
| MAXBYTES        | NUMBER       | Maximum data file size in bytes                                                                                                                                                              |
| MAXBLOCKS       | NUMBER       | Maximum data file size in blocks                                                                                                                                                             |
| INCREMENT_BY    | NUMBER       | Auto increment value                                                                                                                                                                         |
| ONLINE_STATUS   | VARCHAR(7)   | Online Status of the data file<br><br>- SYSOFF : system offline file<br><br>- SYSTEM : system online file<br><br>- OFFLINE<br><br>- ONLINE<br><br>- RECOVER : Datafile needs to be recovered |

### 3.2.118. DBA\_DB\_LINKS

Displays information for all database links.

- See also

[ALL\\_DB\\_LINKS](#)

[USER\\_DB\\_LINKS](#)

### 3.2.119. DBA\_DEPENDENCIES

Displays dependencies of all objects.

- See also

[ALL\\_DEPENDENCIES](#)

[USER\\_DEPENDENCIES](#)

### 3.2.120. DBA\_DIRECTORIES

Displays all directories in the database.

- See also

[ALL\\_DIRECTORIES](#)

### 3.2.121. DBA\_ENCRYPTED\_COLUMNS

Displays information of all encrypted columns in the database. The columns are the same as those in ALL\_ENCRYPTED\_COLUMNS.

- See also

[ALL\\_ENCRYPTED\\_COLUMNS](#)

[USER\\_ENCRYPTED\\_COLUMNS](#)

### 3.2.122. DBA\_ERRORS

Displays the recent errors on every view, procedure, function, package, package body, trigger, library, and java object in the database. The columns are the same as those in ALL\_ERRORS.

- See also

[ALL\\_ERRORS](#)

[USER\\_ERRORS](#)

### 3.2.123. DBA\_EXTENTS

Displays information of the extents contained in the segments of all the tablespaces in the database.

- Columns

| Column Name     | Data Type    | Description                                                           |
|-----------------|--------------|-----------------------------------------------------------------------|
| OWNER           | VARCHAR(128) | Owner of the segment containing the extent                            |
| SEGMENT_NAME    | VARCHAR(128) | Name of the segment containing the extent                             |
| PARTITION_NAME  | VARCHAR(128) | Partition name of the object (Null value for non-partitioned objects) |
| SEGMENT_TYPE    | VARCHAR(18)  | Type of the segment                                                   |
| TABLESPACE_NAME | VARCHAR(30)  | Name of the tablespace containing the extent                          |
| EXTENT_ID       | NUMBER       | ID of the extent                                                      |
| FILE_ID         | NUMBER       | ID of the file containing the extent                                  |
| BLOCK_ID        | NUMBER       | Starting block number of the extent                                   |
| BYTES           | NUMBER       | Size of the extent in bytes                                           |
| BLOCKS          | NUMBER       | Size of the extent in blocks                                          |
| RELATIVE_FNO    | NUMBER       | Relative file number of the extent                                    |

- See also

[USER\\_EXTENTS](#)

### 3.2.124. DBA\_EXTERNAL\_LOCATIONS

Displays all external table locations in the database.

- See also

[ALL\\_EXTERNAL\\_LOCATIONS](#)

[USER\\_EXTERNAL\\_LOCATIONS](#)

### 3.2.125. DBA\_EXTERNAL\_TABLES

Displays all external tables in the database.

- See also

[ALL\\_EXTERNAL\\_TABLES](#)

[USER\\_EXTERNAL\\_TABLES](#)

### 3.2.126. DBA\_FGA\_AUDIT\_TRAIL

Displays TODO. TODO

### 3.2.127. DBA\_FREE\_SPACE

Displays information of the extents that all tablespaces can use.

- Columns

| Column Name   | Data Type    | Description                                  |
|---------------|--------------|----------------------------------------------|
| TABSPACE_NAME | VARCHAR(128) | Name of the tablespace containing the extent |
| FILE_ID       | NUMBER       | ID of the file containing the extent         |
| BLOCK_ID      | NUMBER       | Starting block number of the extent          |
| BYTES         | NUMBER       | Size of the extent in bytes                  |
| BLOCKS        | NUMBER       | Size of the extent in blocks                 |
| RELATIVE_FNO  | NUMBER       | Relative file number of the extent           |

- See also

[USER\\_FREE\\_SPACE](#)

### 3.2.128. DBA\_FREE\_TABLESPACE

Displays information of the total free space that all tablespaces can use.

- Columns

| Column Name   | Data Type    | Description                                  |
|---------------|--------------|----------------------------------------------|
| TABSPACE_NAME | VARCHAR(128) | Name of the tablespace containing the extent |

| Column Name           | Data Type | Description                      |
|-----------------------|-----------|----------------------------------|
| TABLESPACE_TOTAL_SIZE | NUMBER    | Total size of tablespace         |
| BYTES                 | NUMBER    | Size of the free space in bytes  |
| BLOCKS                | NUMBER    | Size of the free space in blocks |

### 3.2.129. DBA\_HIST\_ACTIVE\_SESS\_HISTORY

Displays history information on active sessions.

- Columns

| Column Name      | Data Type   | Description                                               |
|------------------|-------------|-----------------------------------------------------------|
| SNAP_ID          | NUMBER      | Snap ID                                                   |
| SAMPLE_ID        | NUMBER      | Sample ID                                                 |
| THREAD#          | NUMBER      | Thread#                                                   |
| SAMPLE_TIME      | DATE        | Sample Time                                               |
| SESSION_ID       | NUMBER      | Session ID                                                |
| SESS_SERIAL_NO   | NUMBER      | A serial number of a session                              |
| USER_NO          | NUMBER      | The user's ID                                             |
| WAIT_EVENT       | NUMBER      | Type of the wait event which the session waits            |
| TIME_WAITED      | NUMBER      | The amount of time waited for this event (msec)           |
| SQL_ID           | VARCHAR(13) | SQL identifier                                            |
| SQL_CHILD_NUMBER | NUMBER      | The number of this child physical plan                    |
| SUB_SQL_ID       | VARCHAR(13) | Identifier of the internal SQL referenced by the SQL ID   |
| SUB_CHILD_NUMBER | NUMBER      | Child number of the internal SQL referenced by the SQL ID |
| CURR_HASHVAL     | NUMBER      | SQL hash value                                            |

| Column Name         | Data Type   | Description                                                                                                                                                                                                                                                                                                                          |
|---------------------|-------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| MODULE_NAME         | VARCHAR(64) | Name of a module specified by dbms_application_info.set_module                                                                                                                                                                                                                                                                       |
| ACTION_NAME         | VARCHAR(64) | Name of an action specified by dbms_application_info.set_module/action                                                                                                                                                                                                                                                               |
| CLIENT_INFO_NAME    | VARCHAR(64) | Name of client_info specified by dbms_application_info.set_client_info                                                                                                                                                                                                                                                               |
| PROG_NAME           | VARCHAR(30) | Client program name                                                                                                                                                                                                                                                                                                                  |
| SQL_EXEC_START      | DATE        | The time when the execution of the SQL referenced by SUB_SQL_ID started                                                                                                                                                                                                                                                              |
| SQL_EXEC_ID         | NUMBER      | Execution identifier of the SQL referenced by SUB_SQL_ID. It is necessary to identify multiple simultaneous SQL executions. To identify a single SQL execution, three columns; SUB_SQL_ID, SQL_EXEC_START, and SQL_EXEC_ID must be defined. The three column values are used to match with corresponding rows in the V\$SQL_MONITOR. |
| SQL_PLAN_LINE_ID    | NUMBER      | Identifier of the SQL plan operation referenced by SUB_SQL_ID                                                                                                                                                                                                                                                                        |
| ID1                 | NUMBER      | First Parameter                                                                                                                                                                                                                                                                                                                      |
| ID2                 | NUMBER      | Second Parameter                                                                                                                                                                                                                                                                                                                     |
| WE_SEQ              | NUMBER      | Sequence# of the Wait Event                                                                                                                                                                                                                                                                                                          |
| USGMT_ID            | NUMBER      | USGMT number at the time of sampling                                                                                                                                                                                                                                                                                                 |
| SLOTNO              | NUMBER      | Slot number at the time of sampling                                                                                                                                                                                                                                                                                                  |
| WRAPNO              | NUMBER      | Wrap number at the time of sampling                                                                                                                                                                                                                                                                                                  |
| PORT                | NUMBER      | Client port number                                                                                                                                                                                                                                                                                                                   |
| DELTA_TIME          | NUMBER      | Time interval (in seconds) since the last time this session was sampled or created.                                                                                                                                                                                                                                                  |
| DELTA_PHY_READ_BLKs | NUMBER      | The number of blocks read from disk over the last DELTA_TIME seconds                                                                                                                                                                                                                                                                 |

| Column Name         | Data Type    | Description                                                |
|---------------------|--------------|------------------------------------------------------------|
| DELTA_LOG_READ_BKLS | NUMBER       | The number of blocks read over the last DELTA_TIME seconds |
| PGA_SIZE            | NUMBER       | The amount of PGA memory consumed by this session          |
| WAIT_OBJ_ID         | NUMBER       | ID of the object which the session waits                   |
| WAIT_FILE_NO        | NUMBER       | Number of the file which the session waits                 |
| WAIT_BLOCK_NO       | NUMBER       | Number of the block which the session waits                |
| WAIT_ROW_NO         | NUMBER       | Number of the row which the session waits                  |
| WAIT_EVENT_NAME     | VARCHAR(100) | Name of wait_event                                         |

### 3.2.130. DBA\_HIST\_FILESTAT

Displays historical statistics of data file and temp file I/O. This view includes snapshots of V\$ALL\_FILESTAT.

- Columns

| Column Name     | Data Type | Description                              |
|-----------------|-----------|------------------------------------------|
| SNAP_ID         | NUMBER    | Unique ID of a snapshot                  |
| INSTANCE_NUMBER | NUMBER    | Instance number of the snapshot          |
| FILE#           | NUMBER    | File number                              |
| PHYRDS          | NUMBER    | The number of read operations performed  |
| PHYWRTS         | NUMBER    | The number of write operations performed |
| READTIM         | NUMBER    | Time spent on read operations (msec)     |
| WRITETIM        | NUMBER    | Time spent on write operations (msec)    |
| PHYBLKRD        | NUMBER    | The number of blocks read                |
| PHYBLKWRT       | NUMBER    | The number of blocks written             |

### 3.2.131. DBA\_HIST\_LATCH

Displays the historical information on spinlocks. This view includes snapshots of V\$LATCH.

- Columns

| Column Name      | Data Type | Description                                                                                    |
|------------------|-----------|------------------------------------------------------------------------------------------------|
| SNAP_ID          | NUMBER    | Unique ID of a snapshot                                                                        |
| INSTANCE_NUMBER  | NUMBER    | Instance number of the snapshot                                                                |
| LATCH#           | NUMBER    | Spinlock number                                                                                |
| GETS             | NUMBER    | The number of times a spinlock was acquired                                                    |
| MISSES           | NUMBER    | The number of times the requester had to wait for a spinlock                                   |
| IMMEDIATE_GETS   | NUMBER    | The number of times a trylock was acquired                                                     |
| IMMEDIATE_MISSES | NUMBER    | The number of times the requester failed to acquire a trylock                                  |
| SPIN_GETS        | NUMBER    | The number of times the requester failed to acquire a spinlock but succeeded in gaining a spin |
| WAIT_TIME        | NUMBER    | The total sleep time of spinlock                                                               |
| SLEEPS_CNT       | NUMBER    | The number of actual sleep count while waiting for a spinlock                                  |

### 3.2.132. DBA\_HIST\_LIBRARYCACHE

Displays information of PP cache and DD cache. This view includes snapshots of V\$LIBRARYCACHE.

- Columns

| Column Name     | Data Type   | Description                     |
|-----------------|-------------|---------------------------------|
| SNAP_ID         | NUMBER      | Unique ID of a snapshot         |
| INSTANCE_NUMBER | NUMBER      | Instance number of the snapshot |
| NAMESPACE       | VARCHAR(16) | Name of the table               |

| Column Name | Data Type | Description              |
|-------------|-----------|--------------------------|
| GETS        | NUMBER    | Cache total access count |
| GETHITS     | NUMBER    | Cache hit count          |

### 3.2.133. DBA\_HIST\_OSSTAT

Displays historical information on various OS statistics. This view includes snapshots of V\$OSSTAT2.

- Columns

| Column Name     | Data Type | Description                     |
|-----------------|-----------|---------------------------------|
| SNAP_ID         | NUMBER    | Unique ID of a snapshot         |
| INSTANCE_NUMBER | NUMBER    | Instance number of the snapshot |
| STAT_ID         | NUMBER    | Identifier of OS statistics     |
| VALUE           | NUMBER    | OS statistics value             |

### 3.2.134. DBA\_HIST\_PGASTAT

Displays the historical statistics of PGA. This view includes the snapshot of V\$PGASTAT.

- Columns

| Column Name     | Data Type   | Description                     |
|-----------------|-------------|---------------------------------|
| SNAP_ID         | NUMBER      | Unique ID of a snapshot         |
| INSTANCE_NUMBER | NUMBER      | Instance number of the snapshot |
| NAME            | VARCHAR(64) | Name of a PGA statistics        |
| VALUE           | NUMBER      | Value of the PGA statistics     |
| SNAP_TIME       | DATE        | Snapshot time of PGA statistics |

### 3.2.135. DBA\_HIST\_PROCESS\_MEM\_SUMMARY

Displays historical information on PGA memory usage. This view gets information from V\$PROCESS.

- Columns

| Column Name     | Data Type | Description                                    |
|-----------------|-----------|------------------------------------------------|
| SNAP_ID         | NUMBER    | Unique ID of a snapshot.                       |
| INSTANCE_NUMBER | NUMBER    | Instance number of the snapshot                |
| NUM_PROCESSES   | NUMBER    | Number of processes                            |
| NON_ZERO_USES   | NUMBER    | Number of processes with non-zero memory usage |
| USED_TOTAL      | NUMBER    | Total size of local memory occupied            |
| USED_AVG        | NUMBER    | Average size of local memory occupied          |
| USED_STDDEV     | NUMBER    | Standard deviation of local memory occupied    |
| USED_MAX        | NUMBER    | Maximum size of local memory occupied          |

### 3.2.136. DBA\_HIST\_SEG\_STAT

Displays historical information of segment-level statistics. This view includes snapshots of V\$SEGMENT\_STATISTICS.

- Columns

| Column Name     | Data Type   | Description                     |
|-----------------|-------------|---------------------------------|
| SNAP_ID         | NUMBER      | Unique ID of a snapshot         |
| OBJ_ID          | NUMBER      | Object ID                       |
| STATISTICS#     | NUMBER      | Segment statistics type         |
| STATISTICS_NAME | VARCHAR(64) | Name of the segment statistics  |
| VALUE           | NUMBER      | Value of the segment statistics |

### 3.2.137. DBA\_HIST\_SGASTAT

Displays the historical information on the shared pool for a database. This view includes snapshots of V\$SGASTAT.

- Columns

| Column Name     | Data Type   | Description                           |
|-----------------|-------------|---------------------------------------|
| SNAP_ID         | NUMBER      | Unique ID of a snapshot               |
| INSTANCE_NUMBER | NUMBER      | Instance number of the snapshot       |
| NAME            | VARCHAR(40) | Purpose for which shared pool is used |
| SIZE            | NUMBER      | Usage of shared pool                  |

### 3.2.138. DBA\_HIST\_SNAPSHOT

Displays information about TPR snapshots.

- Columns

| Column Name             | Data Type | Description                       |
|-------------------------|-----------|-----------------------------------|
| SNAP_ID                 | NUMBER    | Unique ID of a snapshot           |
| INSTANCE_NUMBER         | NUMBER    | Instance number of the snapshot   |
| BEGIN_INTER<br>VAL_TIME | DATE      | Start time of a snapshot interval |
| END_INTERVAL_TIME       | DATE      | End time of a snapshot interval   |

### 3.2.139. DBA\_HIST\_SQLBIND

Displays historical information on bind variables used by SQL cursors. This view contains snapshots of V\$SQL\_BIND\_CAPTURE.

- Columns

| Column Name | Data Type | Description             |
|-------------|-----------|-------------------------|
| SNAP_ID     | NUMBER    | Unique ID of a snapshot |

| Column Name     | Data Type      | Description                                                      |
|-----------------|----------------|------------------------------------------------------------------|
| INSTANCE_NUMBER | NUMBER         | Instance number of the snapshot                                  |
| SQL_HASH_VALUE  | NUMBER         | The Hash value of the sql                                        |
| PLAN_HASH_VALUE | NUMBER         | The Hash value of the physical plan                              |
| SQL_ID          | VARCHAR(13)    | The SQL identifier                                               |
| CHILD_NUMBER    | NUMBER         | The number of this child physical plan                           |
| USER_ID         | NUMBER         | The user id that performed the SQL                               |
| NAME            | VARCHAR(128)   | The name of the bind variable                                    |
| POSITION        | NUMBER         | The position of the bind variable in the SQL                     |
| TYPE            | NUMBER         | The type number of the bind variable                             |
| TYPE_NAME       | VARCHAR(128)   | The type of the bind variable                                    |
| PRECISION       | NUMBER         | The precision of the bind variable                               |
| SCALE           | NUMBER         | The scale of the bind variable                                   |
| LENGTH          | NUMBER         | The length of the bind variable                                  |
| TIME_CAPTURED   | DATE           | The time when the bind variable was captured                     |
| IS_FIRST        | VARCHAR(4)     | Whether or not the first value of the bind variable was captured |
| VALUE           | VARCHAR(32413) | The value of the bind variable                                   |

### 3.2.140. DBA\_HIST\_SQLSTAT

Displays historical information for SQL statistics. This view displays the priority of items of specific criteria.

- Columns

| Column Name     | Data Type | Description                     |
|-----------------|-----------|---------------------------------|
| SNAP_ID         | NUMBER    | Unique ID of a snapshot         |
| INSTANCE_NUMBER | NUMBER    | Instance number of the snapshot |

| Column Name          | Data Type   | Description                                          |
|----------------------|-------------|------------------------------------------------------|
| SQL_ID               | VARCHAR(13) | SQL identifier                                       |
| SQL_HASH_VALUE       | NUMBER      | Hash value of the SQL statement                      |
| PLAN_HASH_VALUE      | NUMBER      | Hash value of the physical plan                      |
| LAST_ACTIVE_TIME     | DATE        | Last time that statistics were collected             |
| PARSE_CALLS          | NUMBER      | The number of parsing requests                       |
| FETCHES              | NUMBER      | The number of fetching times                         |
| EXECUTIONS           | NUMBER      | The number of times for executing an SQL statement   |
| BUFFER_GETS          | NUMBER      | Total number of buffers read in Consistent-Read mode |
| ROWS_PROCESSED       | NUMBER      | Total number of processed rows                       |
| ELAPSED_TIME         | NUMBER      | Total execution time (in microseconds)               |
| DISK_READ_TIME       | NUMBER      | Total disk read time                                 |
| TEMP_SGMT_READ_TIME  | NUMBER      | Total temp segment read time                         |
| TEMP_SGMT_WRITE_TIME | NUMBER      | Total temp segment write time                        |

### 3.2.141. DBA\_HIST\_SQLTEXT

Displays historical information about texts of SQL statements.

- Columns

| Column Name    | Data Type   | Description                                                                                      |
|----------------|-------------|--------------------------------------------------------------------------------------------------|
| SQL_ID         | VARCHAR(13) | SQL identifier                                                                                   |
| SQL_HASH_VALUE | NUMBER      | Hash value of the SQL statement                                                                  |
| SQL_TEXT       | VARCHAR(64) | Text of an SQL statement split into 64-byte texts                                                |
| COMMAND_TYPE   | NUMBER      | Number code of SQL statement type 1 - SELECT 2 - INSERT 3 - UPDATE 4 - DELETE 5 - MERGE 6 - CALL |

### 3.2.142. DBA\_HIST\_SQL\_PLAN

Displays historical information about SQL execution plans. This view includes snapshots of V\$SQL\_PLAN.

- Columns

| Column Name     | Data Type     | Description                                                                                           |
|-----------------|---------------|-------------------------------------------------------------------------------------------------------|
| SQL_ID          | VARCHAR(13)   | SQL identifier                                                                                        |
| SQL_HASH_VALUE  | NUMBER        | Hash value of the SQL statement                                                                       |
| PLAN_HASH_VALUE | NUMBER        | Hash value of the physical plan                                                                       |
| ID              | NUMBER        | Number given to each job of the physical plan                                                         |
| OPERATION       | VARCHAR(128)  | Name of operation job                                                                                 |
| OBJECT_ID       | NUMBER        | Identifier of object accessed by the job                                                              |
| OBJECT_OWNER    | VARCHAR(128)  | Name of user to whom the object belongs                                                               |
| OBJECT_NAME     | VARCHAR(128)  | Object name                                                                                           |
| OBJECT_TYPE     | VARCHAR(20)   | Object type                                                                                           |
| PARENT_ID       | NUMBER        | ID of a job to get the output of the current job of the physical plan as input                        |
| DEPTH           | NUMBER        | Tree level of the physical plan                                                                       |
| POSITION        | NUMBER        | Positions between all the jobs with the same PARENT_ID                                                |
| COST            | NUMBER        | Job cost estimated by query optimizer                                                                 |
| CARDINALITY     | NUMBER        | The number of output results estimated by query optimizer                                             |
| PARTITION_START | VARCHAR(38)   | In a partitioned table, the start partition for access                                                |
| PARTITION_STOP  | VARCHAR(38)   | In a partitioned table, the end partition for access                                                  |
| OTHER           | VARCHAR(4000) | Other information specific to the execution step that may be useful<br><br>-Database link information |

| Column Name       | Data Type     | Description                                                                                                                                                                                 |
|-------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                   |               | -Dynamic sampling block count for table scans or index scans                                                                                                                                |
| CPU_COST          | NUMBER        | CPU cost of the operation as estimated by query optimizer                                                                                                                                   |
| IO_COST           | NUMBER        | I/O cost of the operation as estimated by query optimizer                                                                                                                                   |
| ACCESS_PREDICATES | VARCHAR(4000) | Predicate information to access indexes indexes or to handle joins                                                                                                                          |
| FILTER_PREDICATES | VARCHAR(4000) | Predicate information to handle filters                                                                                                                                                     |
| OTHER_XML         | CLOB          | Extra information specific to an execution step of the execution plan, structured using XML<br><br>-The outline data, a set of optimizer hints that can be used to regenerate the same plan |

### 3.2.143. DBA\_HIST\_SQL\_WORKAREA\_HSTGRM

Displays historical information about work areas used by SQL cursors. This view includes snapshots of V\$SQL\_WORKAREA\_HISTOGRAM.

- Columns

| Column Name        | Data Type | Description                                                                                                          |
|--------------------|-----------|----------------------------------------------------------------------------------------------------------------------|
| SNAP_ID            | NUMBER    | Unique ID of a snapshot                                                                                              |
| INSTANCE_NUMBER    | NUMBER    | Instance number of the snapshot                                                                                      |
| LOW_OPTIMAL_SIZE   | NUMBER    | The lower bounding value (in bytes) for the optimal size                                                             |
| HIGH_OPTIMAL_SIZE  | NUMBER    | The higher bounding value (in bytes) for the optimal size                                                            |
| OPTIMAL_EXECUTIONS | NUMBER    | The number of work areas that were executed in optimal mode within each LOW_OPTIMAL_SIZE and HIGH_OPTIMAL_SIZE bound |
| ONEPASS_EXECUTIONS | NUMBER    | The number of work areas that were executed in onepass mode within each LOW_OPTIMAL_SIZE and HIGH_OPTIMAL_SIZE bound |

| Column Name            | Data Type | Description                                                                                                             |
|------------------------|-----------|-------------------------------------------------------------------------------------------------------------------------|
| MULTIPASSES_EXECUTIONS | NUMBER    | The number of work areas that were executed in multi pass mode within each LOW_OPTIMAL_SIZE and HIGH_OPTIMAL_SIZE bound |
| TOTAL_EXECUTIONS       | NUMBER    | The total number of work areas that were excuted within each LOW_OPTIMAL_SIZE and HIGH_OPTIMAL_SIZE bound               |

### 3.2.144. DBA\_HIST\_SYSMETRIC\_SUMMARY

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

- Columns

| Column Name | Data Type   | Description                     |
|-------------|-------------|---------------------------------|
| SNAP_ID     | NUMBER      | Unique ID of a snapshot         |
| THREAD#     | NUMBER      | Thread#                         |
| BEGIN_TIME  | DATE        | Begin Time of the Sample        |
| END_TIME    | DATE        | End Time of the Sample          |
| GROUP_ID    | NUMBER      | Group ID                        |
| METRIC_ID   | NUMBER      | Metric ID                       |
| METRIC_NAME | VARCHAR(64) | Metric Name                     |
| METRIC_UNIT | VARCHAR(64) | Descriptions of the Metric Unit |
| MINVAL      | NUMBER      | Minimum value observed          |
| MINVAL      | NUMBER      | Maximum value observed          |
| AVERAGE     | NUMBER      | Average over the period         |
| STD         | NUMBER      | One standard deviation          |

### 3.2.145. DBA\_HIST\_SYSTEM\_EVENT

Displays historical information of events performing in the system. This view includes snapshots of V\$SYSTEM\_EVENT.

- Columns

| Column Name     | Data Type | Description                                              |
|-----------------|-----------|----------------------------------------------------------|
| SNAP_ID         | NUMBER    | Unique ID of a snapshot                                  |
| INSTANCE_NUMBER | NUMBER    | Instance number of the snapshot                          |
| EVENT_NUMBER    | NUMBER    | Event number                                             |
| TOTAL_WAITS     | NUMBER    | Total number of event calls                              |
| TIME_WAITED     | NUMBER    | The amount of time waited for this event (msec)          |
| MAX_WAIT        | NUMBER    | The maximum amount of time waited for this event (msec)  |
| TOTAL_TIMEOUTS  | NUMBER    | The number of times that a time-out occurred in an event |

### 3.2.146. DBA\_HIST\_UNDOSTAT

Displays historical statistics of undo storage space. This view includes snapshots of V\$UNDOSTAT.

- Columns

| Column Name     | Data Type | Description                                              |
|-----------------|-----------|----------------------------------------------------------|
| END_TIME        | DATE      | End time of statistics                                   |
| INSTANCE_NUMBER | NUMBER    | Instance number of the snapshot                          |
| SNAP_ID         | NUMBER    | Unique ID of a snapshot                                  |
| UNDO_TSNO       | NUMBER    | Tablespace number                                        |
| UNDOBLKS        | NUMBER    | The number of used undo blocks                           |
| TXCOUNT         | NUMBER    | Total number of executed transactions                    |
| MAXCONCURRENCY  | NUMBER    | The largest number of concurrently executed transactions |

| Column Name      | Data Type | Description                                               |
|------------------|-----------|-----------------------------------------------------------|
| SNAP_TOO_OLD_CNT | NUMBER    | The number of errors caused by failing to read old data   |
| NOSPACEERRCNT    | NUMBER    | The number of errors caused by lack of undo storage space |
| ACTIVEBLKS       | NUMBER    | The number of blocks in use                               |
| UNEXPIREDBLKS    | NUMBER    | The number of non-reusable blocks                         |
| EXPIREDBLKS      | NUMBER    | The number of reusable blocks                             |

### 3.2.147. DBA\_HIST\_WAITSTAT

Displays the statistical history of block wait events for each block type and class. This view includes snapshots of V\$WAITSTAT.

- Columns

| Column Name     | Data Type   | Description                                     |
|-----------------|-------------|-------------------------------------------------|
| SNAP_ID         | NUMBER      | Unique ID of a snapshot                         |
| INSTANCE_NUMBER | NUMBER      | Instance number of the snapshot                 |
| CLASS           | VARCHAR(20) | Block class (Block type)                        |
| WAIT_COUNT      | NUMBER      | The number of times waiting for the block class |
| TIME            | NUMBER      | Total time spent waiting for the block class    |

### 3.2.148. DBA\_IDX\_COLUMNS

Displays index columns of all tables in the database.

- See also

[ALL\\_IDX\\_COLUMNS](#)

[USER\\_IDX\\_COLUMNS](#)

### 3.2.149. DBA\_IDX\_EXPRESSIONS

Displays expressions on the index columns of all tables in the database.

- See also

[ALL\\_IDX\\_EXPRESSIONS](#)

[USER\\_IDX\\_EXPRESSIONS](#)

### 3.2.150. DBA\_IDX\_PARTITIONS

displays all index partitions in the database.

- See also

[ALL\\_IDX\\_PARTITIONS](#)

[USER\\_IDX\\_PARTITIONS](#)

### 3.2.151. DBA\_IDX\_SUBPARTITIONS

Displays subpartitions of all indexes in the database.

- See also

[ALL\\_IDX\\_SUBPARTITIONS](#)

### 3.2.152. DBA\_INDEXES

Displays indexes of all tables in the database.

- See also

[ALL\\_INDEXES](#)

[USER\\_INDEXES](#)

### 3.2.153. DBA\_IND\_PENDING\_STATS

Displays indexes of all tables in the database.

- See also

[ALL\\_IND\\_PENDING\\_STATS](#)

[USER\\_IND\\_PENDING\\_STATS](#)

### 3.2.154. DBA\_JAVA\_ARGUMENTS

Displays all parameters in the database.

- See also

[ALL\\_JAVA\\_ARGUMENTS](#)

[USER\\_JAVA\\_ARGUMENTS](#)

### 3.2.155. DBA\_JAVA\_CLASSES

Displays all classes in the database.

- See also

[ALL\\_JAVA\\_CLASSES](#)

[USER\\_JAVA\\_CLASSES](#)

### 3.2.156. DBA\_JAVA\_FIELDS

Displays all fields in the database.

- See also

[ALL\\_JAVA\\_FIELDS](#)

[USER\\_JAVA\\_FIELDS](#)

### 3.2.157. DBA\_JAVA\_METHODS

Displays all methods in the database.

- See also

[ALL\\_JAVA\\_METHODS](#)

[USER\\_JAVA\\_METHODS](#)

### 3.2.158. DBA\_JOBS

Displays all jobs in the database.

- See also

[ALL\\_JOBS](#)

[USER\\_JOBS](#)

### 3.2.159. DBA\_JOBS\_WITH\_NAME

Displays all jobs with job name in the database.

- See also

[ALL\\_JOBS\\_WITH\\_NAME](#)

[USER\\_JOBS\\_WITH\\_NAME](#)

### 3.2.160. DBA\_LIBRARIES

Displays all libraries in the database.

- See also

[ALL\\_LIBRARIES](#)

[USER\\_LIBRARIES](#)

### 3.2.161. DBA\_LOBS

Displays LOBs which belong to all tables in the database.

- See also

[ALL\\_LOBS](#)

[USER\\_LOBS](#)

### 3.2.162. DBA\_LOB\_PARTITIONS

Displays all LOB partitions in the database.

- See also

[ALL\\_LOB\\_PARTITIONS](#)

[USER\\_LOB\\_PARTITIONS](#)

### 3.2.163. DBA\_LOB\_SUBPARTITIONS

Displays all LOB subpartitions in the database.

- See also

[ALL\\_LOB\\_SUBPARTITIONS](#)

[USER\\_LOB\\_SUBPARTITIONS](#)

### 3.2.164. DBA\_LOG\_GROUPS

Displays the log group definitions on all tables in the database.

- See also

[ALL\\_LOG\\_GROUPS](#)

[USER\\_LOG\\_GROUPS](#)

### 3.2.165. DBA\_LOG\_GROUP\_COLUMNS

Displays all columns in the database that are specified in log groups.

- See also

[ALL\\_LOG\\_GROUP\\_COLUMNS](#)

[USER\\_LOG\\_GROUP\\_COLUMNS](#)

### 3.2.166. DBA\_METHOD\_PARAMS

Displays the method parameters of this type.

- See also

[ALL\\_METHOD\\_PARAMS](#)

[USER\\_METHOD\\_PARAMS](#)

### 3.2.167. DBA\_METHOD\_RESULTS

Displays the result of the method of this type.

- See also

[ALL\\_METHOD\\_RESULTS](#)

[USER\\_METHOD\\_RESULTS](#)

### 3.2.168. DBA\_MVIEWS

Displays all materialized views in the database. The columns are the same as those in ALL\_MVIEWS.

- See also

[ALL\\_MVIEWS](#)

[USER\\_MVIEWS](#)

### 3.2.169. DBA\_MVIEW\_COMMENTS

Displays comments on all materialized views in the database. The columns are the same as those in ALL\_MVIEW\_COMMENTS.

- See also

[ALL\\_MVIEW\\_COMMENTS](#)

[USER\\_MVIEW\\_COMMENTS](#)

### 3.2.170. DBA\_MVIEW\_DETAIL\_RELATIONS

Describes the named detail relations that are either specified in the FROM list of the subquery that defines all materialized views in the database, or that are indirectly referenced through views in that FROM list. Inline views in the materialized view definition are not represented in this view or the related views.

- See also

[ALL\\_MVIEW\\_DETAIL\\_RELATIONS](#)

[USER\\_MVIEW\\_DETAIL\\_RELATIONS](#)

### 3.2.171. DBA\_MVIEW\_LOGS

Displays all materialized view logs in the database. The columns are the same as those in ALL\_MVIEW\_LOGS.

- See also

[ALL\\_MVIEW\\_LOGS](#)

[USER\\_MVIEW\\_LOGS](#)

### 3.2.172. DBA\_MVIEW\_REFRESH\_TIMES

Describes refresh times of all materialized views in the database.

- See also

[ALL\\_MVIEW\\_REFRESH\\_TIMES](#)

[USER\\_MVIEW\\_REFRESH\\_TIMES](#)

### 3.2.173. DBA\_OBJECTS

Displays all objects in the database.

- See also

[ALL\\_OBJECTS](#)

[USER\\_OBJECTS](#)

### 3.2.174. DBA\_OBJ\_AUDIT\_OPTS

Displays objects being audited in the database.

- Columns

| Column Name | Data Type    | Description                   |
|-------------|--------------|-------------------------------|
| OWNER       | VARCHAR(128) | Owner of the object           |
| OBJECT_NAME | VARCHAR(128) | Name of the object            |
| OBJECT_TYPE | VARCHAR(9)   | Type of the object<br>- TABLE |

| Column Name | Data Type  | Description                                                  |
|-------------|------------|--------------------------------------------------------------|
|             |            | - VIEW<br>- SEQUENCE<br>- PROCEDURE<br>- DIRECTORY<br>- JAVA |
| ALT         | VARCHAR(5) | ALTER auditing option for success or failure                 |
| DEL         | VARCHAR(5) | DELETE auditing option for success or failure                |
| IND         | VARCHAR(5) | INDEX auditing option for success or failure                 |
| INS         | VARCHAR(5) | INSERT auditing option for success or failure                |
| LOC         | VARCHAR(5) | LOCK auditing option for success or failure                  |
| SEL         | VARCHAR(5) | SELECT auditing option for success or failure                |
| UPD         | VARCHAR(5) | UPDATE auditing option for success or failure                |
| REF         | VARCHAR(5) | REFERENCE auditing option for success or failure             |
| EXE         | VARCHAR(5) | EXECUTE auditing option for success or failure               |
| REA         | VARCHAR(5) | READ auditing option for success or failure                  |
| WRI         | VARCHAR(5) | WRITE auditing option for success or failure                 |

- See also

[USER\\_OBJ\\_AUDIT\\_OPTS](#)

### 3.2.175. DBA\_PACKAGE\_COLL\_TYPES

Displays types of the package.

- See also

[ALL\\_PACKAGE\\_COLL\\_TYPES](#)

[USER\\_PACKAGE\\_COLL\\_TYPES](#)

### 3.2.176. DBA\_PACKAGE\_MEMBER\_TYPES

Displays types of the package.

- See also

[ALL\\_PACKAGE\\_MEMBER\\_TYPES](#)

[USER\\_PACKAGE\\_MEMBER\\_TYPES](#)

### 3.2.177. DBA\_PACKAGE\_TYPES

Displays types of the package.

- See also

[ALL\\_PACKAGE\\_TYPES](#)

[USER\\_PACKAGE\\_TYPES](#)

### 3.2.178. DBA\_PARALLEL\_EXECUTE\_CHUNKS

Displays the chunks for all tasks in the database.

- Columns

| Column Name | Data Type     | Description                                                                                                |
|-------------|---------------|------------------------------------------------------------------------------------------------------------|
| CHUNK_ID    | NUMBER        | ID for the chunk                                                                                           |
| TASK_OWNER  | VARCHAR2(128) | Owner of the task                                                                                          |
| TASK_NAME   | VARCHAR2(128) | Name of the task                                                                                           |
| STATUS      | VARCHAR2(20)  | Status of the chunk<br><br>- UNASSIGNED<br><br>- ASSIGNED<br><br>- PROCESSED<br><br>- PROCESSED_WITH_ERROR |
| START_ROWID | ROWID         | Rowid for the first row in the chunk                                                                       |
| END_ROWID   | ROWID         | Rowid for the end row in the chunk                                                                         |

| Column Name   | Data Type      | Description                                                                                        |
|---------------|----------------|----------------------------------------------------------------------------------------------------|
| START_ID      | NUMBER         | Number column value of the first row in the chunk                                                  |
| END_ID        | NUMBER         | Number column value of the end 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         | Returned error code for the execution of the chunk if the STATUS column is PROCESSED_WITH_ERROR    |
| ERROR_MESSAGE | VARCHAR2(4000) | Returned error message for the execution of the chunk if the STATUS column is PROCESSED_WITH_ERROR |

- See also

[USER\\_PARALLEL\\_EXECUTE\\_CHUNKS](#)

### 3.2.179. DBA\_PARALLEL\_EXECUTE\_TASKS

Displays all tasks in the database.

- Columns

| Column Name | Data Type     | Description                                                           |
|-------------|---------------|-----------------------------------------------------------------------|
| TASK_OWNER  | VARCHAR2(128) | Owner of the task                                                     |
| TASK_NAME   | VARCHAR2(128) | Name of the task                                                      |
| CHUNK_TYPE  | VARCHAR2(12)  | Type for the chunk<br>- UNDECLARED<br>- ROWID_RANGE<br>- NUMBER_RANGE |
| STATUS      | VARCHAR2(19)  | Status of the task<br>- CREATED<br>- CHUNKING                         |

| Column Name                   | Data Type      | Description                                                                                                       |
|-------------------------------|----------------|-------------------------------------------------------------------------------------------------------------------|
|                               |                | - CHUNKING_FAILED<br>- NO_CHUNKS<br>- CHUNKED<br>- PROCESSING<br>- FINISHED<br>- FINISHED_WITH_ERROR<br>- CRASHED |
| TABLE_OWNER                   | VARCHAR2(128)  | Owner of the table which to be chunked                                                                            |
| TABLE_NAME                    | VARCHAR2(128)  | Name of the table which to be chunked                                                                             |
| NUMBER_COLUMN                 | VARCHAR2(128)  | Name of the column holding IDs (only if CHUNK_TYPE is NUMBER_RANGE)                                               |
| TASK_COMMENT                  | VARCHAR2(4000) | Task comment                                                                                                      |
| JOB_PREFIX                    | VARCHAR2(128)  | Prefix of the job name                                                                                            |
| SQL_STMT                      | CLOB           | Used in DBMS_PARALLEL_EXECUTE.RUN_TASK                                                                            |
| LANGUAGE_FLAG                 | NUMBER         | Used in DBMS_PARALLEL_EXECUTE.RUN_TASK                                                                            |
| APPLY_CROSSED<br>TION_TRIGGER | VARCHAR2(130)  | Used in DBMS_PARALLEL_EXECUTE.RUN_TASK                                                                            |
| FIRE_APPLY_TRIG<br>GER        | VARCHAR2(130)  | Used in DBMS_PARALLEL_EXECUTE.RUN_TASK                                                                            |
| PARALLEL_LEVEL                | NUMBER         | Used in DBMS_PARALLEL_EXECUTE.RUN_TASK                                                                            |
| JOB_CLASS                     | VARCHAR2(128)  | Used in DBMS_PARALLEL_EXECUTE.RUN_TASK                                                                            |

- See also

[USER\\_PARALLEL\\_EXECUTE\\_TASKS](#)

### 3.2.180. DBA\_PART\_INDEXES

Displays partitioning information for the partitioned indexes in the database.

- See also

[ALL\\_PART\\_INDEXES](#)

[USER\\_PART\\_INDEXES](#)

### 3.2.181. DBA\_PART\_KEY\_COLUMNS

Displays partitioning key columns for all partitioned tables or indexes in the database.

- See also

[ALL\\_PART\\_KEY\\_COLUMNS](#)

[USER\\_PART\\_KEY\\_COLUMNS](#)

### 3.2.182. DBA\_PART\_LOBS

Displays LOBs which belong to the partitioned tables in the database.

- See also

[ALL\\_PART\\_LOBS](#)

[USER\\_PART\\_LOBS](#)

### 3.2.183. DBA\_PART\_TABLES

Displays partitioning information of the partitioned tables in the database.

- See also

[ALL\\_PART\\_TABLES](#)

[USER\\_PART\\_TABLES](#)

### 3.2.184. DBA\_PRIV\_AUDIT\_OPTS

Displays users being audited in the database.

- Columns

| Column Name | Data Type    | Description                    |
|-------------|--------------|--------------------------------|
| USER_NAME   | VARCHAR(128) | Name of the user being audited |

| Column Name | Data Type    | Description                                |
|-------------|--------------|--------------------------------------------|
| PROXY_NAME  | VARCHAR(128) | PROXY name of the user being audited       |
| PRIVILEGE   | VARCHAR(40)  | Name of the system privilege being audited |
| SUCCESS     | VARCHAR(10)  | Auditing option at success                 |
| FAILURE     | VARCHAR(10)  | Auditing option at failure                 |

### 3.2.185. DBA\_PROCEEDURES

Displays all procedures in the database.

- See also

[ALL\\_PROCEEDURES](#)

[USER\\_PROCEEDURES](#)

### 3.2.186. DBA\_PROFILES

Describes all profiles in the database.

- Columns

| Column Name   | Data Type    | Description                                                                |
|---------------|--------------|----------------------------------------------------------------------------|
| PROFILE       | VARCHAR(255) | Name of the profile                                                        |
| RESOURCE_NAME | VARCHAR(255) | Name of the resource                                                       |
| RESOURCE_TYPE | VARCHAR(8)   | Indicates whether the resource profile is a KERNEL or a PASSWORD parameter |
| LIMIT         | VARCHAR(256) | Limit placed on this resource for this profile                             |

### 3.2.187. DBA\_QUEUES

Describes all queues in the database.

- See also

[ALL\\_QUEUES](#)

[USER\\_QUEUES](#)

### 3.2.188. DBA\_QUEUE\_SUBSCRIBERS

Shows the list of subscribers on all queues in the database.

- See also

[ALL\\_QUEUE\\_SUBSCRIBERS](#)

[USER\\_QUEUE\\_SUBSCRIBERS](#)

### 3.2.189. DBA\_QUEUE\_TABLES

Describes the queues in all queue tables in the database.

- See also

[ALL\\_QUEUE\\_TABLES](#)

[USER\\_QUEUE\\_TABLES](#)

### 3.2.190. DBA\_RECYCLEBIN

Displays information of all temporarily deleted objects in the database. The columns are the same as those in ALL\_RECYCLEBIN.

- Columns

| Column Name   | Data Type    | Description                                           |
|---------------|--------------|-------------------------------------------------------|
| OWNER         | VARCHAR(128) | Owner of the object                                   |
| OBJECT_NAME   | VARCHAR(128) | Current name of the object                            |
| ORIGINAL_NAME | VARCHAR(128) | Original name of the object                           |
| TYPE          | VARCHAR(7)   | Type of the object<br>- Table<br>- Index<br>- Trigger |

| Column Name | Data Type    | Description                                     |
|-------------|--------------|-------------------------------------------------|
| TS_NAME     | VARCHAR(128) | Name of the tablespace containing the object    |
| CREATETIME  | VARCHAR(19)  | Timestamp for the object creation               |
| DROPTIME    | VARCHAR(19)  | Timestamp for the object deletion               |
| DROPTSN     | NUMBER       | TSN of the transaction which dropped the object |
| BASE_OBJECT | VARCHAR(256) | Object ID of the base object                    |
| SPACE       | VARCHAR(0)   | Number of blocks used by the object             |

- See also

[ALL\\_RECYCLEBIN](#)

[USER\\_RECYCLEBIN](#)

### 3.2.191. DBA\_REDEFINITION\_OBJECTS

Displays all objects in database which are being redefined online.

- Columns

| Column Name          | Data Type    | Description                                                |
|----------------------|--------------|------------------------------------------------------------|
| OBJECT_TYPE          | VARCHAR(5)   | Type of the object. (Currently, only 'TABLE' is available) |
| OBJECT_OWNER         | VARCHAR(128) | Owner of the object.                                       |
| OBJECT_NAME          | VARCHAR(128) | Name of the object.                                        |
| BASE_TABLE_OWNER     | VARCHAR(128) | Owner of the base table of the object.                     |
| BASE_TABLE_NAME      | VARCHAR(128) | Name of the base table of the object.                      |
| INTERIM_OBJECT_OWNER | VARCHAR(128) | Owner of the intermediate table of the object.             |
| INTERIM_OBJECT_NAME  | VARCHAR(128) | Name of the intermediate table of the object.              |

## 3.2.192. DBA\_ROLES

Displays all roles in the database.

- Columns

| Column Name       | Data Type    | Description                           |
|-------------------|--------------|---------------------------------------|
| ROLE              | VARCHAR(128) | Name of the role                      |
| PASSWORD_REQUIRED | VARCHAR(3)   | Whether a password is required or not |

## 3.2.193. DBA\_ROLE\_PRIVS

Displays roles granted to the users in the database.

- Columns

| Column Name  | Data Type    | Description                                                                                  |
|--------------|--------------|----------------------------------------------------------------------------------------------|
| GRANTEE      | VARCHAR(128) | Name of the user or role which was granted the role                                          |
| GRANTED_ROLE | VARCHAR(128) | Granted role                                                                                 |
| ADMIN_OPTION | VARCHAR(3)   | Whether or not the role was assigned with ADMIN OPTION<br><br>- YES<br><br>- NO<br><br>- ERR |
| DEFAULT_ROLE | VARCHAR(3)   | Whether the role is set to default role for the user<br><br>- YES<br><br>- NO<br><br>- ERR   |

- See also

[USER\\_ROLE\\_PRIVS](#)

### 3.2.194. DBA\_SCHEDULER\_CHAINS

Displays all scheduler chains in the database.

- See also

[ALL\\_SCHEDULER\\_CHAINS](#)

[USER\\_SCHEDULER\\_CHAINS](#)

### 3.2.195. DBA\_SCHEDULER\_JOBS

Displays all scheduler jobs in the database.

- See also

[ALL\\_SCHEDULER\\_JOBS](#)

[USER\\_SCHEDULER\\_JOBS](#)

### 3.2.196. DBA\_SCHEDULER\_JOB\_LOG

Displays all scheduler job log in the database.

- See also

[ALL\\_SCHEDULER\\_JOB\\_LOG](#)

[USER\\_SCHEDULER\\_JOB\\_LOG](#)

### 3.2.197. DBA\_SCHEDULER\_JOB\_RUN\_DETAILS

Displays all scheduler job run details in the database.

- See also

[ALL\\_SCHEDULER\\_JOB\\_RUN\\_DETAILS](#)

[USER\\_SCHEDULER\\_JOB\\_RUN\\_DETAILS](#)

### 3.2.198. DBA\_SCHEDULER\_PROGRAMS

Displays all scheduler programs in the database.

- See also

[ALL\\_SCHEDULER\\_PROGRAMS](#)

[USER\\_SCHEDULER\\_PROGRAMS](#)

### 3.2.199. DBA\_SCHEDULER\_RULES

Displays all scheduler rules in the database.

- See also

[ALL\\_SCHEDULER\\_RULES](#)

[USER\\_SCHEDULER\\_RULES](#)

### 3.2.200. DBA\_SCHEDULER\_RULE\_CTX

Displays all scheduler rule ctx in the database.

- See also

[ALL\\_SCHEDULER\\_RULE\\_CTX](#)

[USER\\_SCHEDULER\\_RULE\\_CTX](#)

### 3.2.201. DBA\_SCHEDULER\_RUNNING\_JOBS

Displays all scheduler running jobs in the database.

- See also

[ALL\\_SCHEDULER\\_RUNNING\\_JOBS](#)

[USER\\_SCHEDULER\\_RUNNING\\_JOBS](#)

### 3.2.202. DBA\_SCHEDULER\_SCHEDULES

Displays all scheduler schedules in the database.

- See also

[ALL\\_SCHEDULER\\_SCHEDULES](#)

[USER\\_SCHEDULER\\_SCHEDULES](#)

### 3.2.203. DBA\_SCHEDULER\_STEPS

Displays all scheduler steps in the database.

- See also

[ALL\\_SCHEDULER\\_STEPS](#)

[USER\\_SCHEDULER\\_STEPS](#)

### 3.2.204. DBA\_SCHEDULER\_STEP\_CTX

Displays all scheduler step context in the database.

- See also

[ALL\\_SCHEDULER\\_STEP\\_CTX](#)

[USER\\_SCHEDULER\\_STEP\\_CTX](#)

### 3.2.205. DBA\_SEGMENTS

Displays all segments in the database.

- Columns

| Column Name    | Data Type    | Description                                                                                                          |
|----------------|--------------|----------------------------------------------------------------------------------------------------------------------|
| OWNER          | VARCHAR(128) | Owner of the segment                                                                                                 |
| SEGMENT_NAME   | VARCHAR(128) | Name of the segment                                                                                                  |
| PARTITION_NAME | VARCHAR(128) | Partition number of the object (Set to NULL for non-partitioned objects)                                             |
| SEGMENT_TYPE   | VARCHAR(18)  | Type of the segment<br>- TABLE<br>- TABLE PARTITION<br>- TABLE SUBPARTITION<br>- LOB<br>- INDEX<br>- INDEX PARTITION |

| Column Name     | Data Type    | Description                                                                      |
|-----------------|--------------|----------------------------------------------------------------------------------|
|                 |              | - INDEX SUBPARTITION<br>- LOB<br>- LOB PARTITION<br>- LOB SUBPARTITION<br>- UNDO |
| TABLESPACE_NAME | VARCHAR(128) | 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                                    |
| BLOCKS          | NUMBER       | Size of the segment in blocks                                                    |
| EXTENTS         | NUMBER       | Size of the segment in extents                                                   |
| RELATIVE_FNO    | NUMBER       | Relative file number of the segment header                                       |
| BUFFER_POOL     | VARCHAR(7)   | Type of the buffer pool<br>- DEFAULT<br>- KEEP<br>- RECYCLE                      |
| MAX_EXTENTS     | NUMBER       | Maximum number of extents that the segment can use                               |
| BYTES           | NUMBER       | Size of segment in bytes                                                         |
| INITIAL_EXTENT  | NUMBER       | Size in bytes requested for the initial extent of the segment                    |
| NEXT_EXTENT     | NUMBER       | Size in bytes of the next extent to be allocated to the segment                  |

- See also

[ALL\\_SEGMENTS](#)

[USER\\_SEGMENTS](#)

## 3.2.206. DBA\_SEQUENCES

Displays all sequences in the database.

- See also

[ALL\\_SEQUENCES](#)

[USER\\_SEQUENCES](#)

## 3.2.207. DBA\_SOURCE

Displays source texts of the stored objects in the database such as procedures, functions, package bodies, or triggers.

- See also

[ALL\\_SOURCE](#)

[USER\\_SOURCE](#)

## 3.2.208. DBA\_SQL\_PLAN\_BASELINES

Displays information about the created SQL plan baselines

- Columns

| Column Name             | Data Type    | Description                                                                                          |
|-------------------------|--------------|------------------------------------------------------------------------------------------------------|
| SIGNATURE               | NUMBER       | Unique SQL identifier generated from normalized SQL TEXT                                             |
| SQL_HANDLE              | VARCHAR2(30) | Unique SQL identifier                                                                                |
| SQL_TEXT                | CLOB         | Un-normalized SQL text                                                                               |
| PLAN_NAME               | VARCHAR2(30) | Unique plan identifier                                                                               |
| CREATOR                 | VARCHAR2(30) | User who created the plan baseline                                                                   |
| ORIGIN                  | VARCHAR2(14) | How the plan baseline was created: MANUAL-LOAD<br>AUTO-CAPTURE    MANUAL-SQLTUNE    AUTO-SQL<br>TUNE |
| PARS<br>ING_SCHEMA_NAME | VARCHAR2(30) | Parsing schema name                                                                                  |

| Column Name        | Data Type     | Description                                         |
|--------------------|---------------|-----------------------------------------------------|
| DESCRIPTION        | VARCHAR2(500) | Description about the plan baseline                 |
| VERSION            | VARCHAR2(64)  | Database version when the plan baseline was created |
| CREATED            | TIMESTAMP(6)  | 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  |
| LAST_VERIFIED      | TIMESTAMP(6)  | Timestamp when the plan baseline was last verified  |
| ENABLED            | VARCHAR2(3)   | Whether to enable the plan baseline                 |
| ACCEPTED           | VARCHAR2(3)   | Whether to accept the plan baseline                 |
| FIXED              | VARCHAR2(3)   | Whether to fix the plan baseline                    |
| AUTOPURGE          | VARCHAR2(3)   | Whether to automatically purge the plan baselines   |
| OPTIMIZER_COST     | NUMBER        | Optimizer cost when the plan baseline created       |
| MODULE             | VARCHAR2(48)  | Module name                                         |
| ACTION             | VARCHAR2(32)  | Action                                              |
| EXECUTIONS         | NUMBER        | Number of executions                                |
| ELAPSED_TIME       | NUMBER        | Total elapsed time                                  |
| CPU_TIME           | NUMBER        | Total CPU time                                      |
| BUFFER_GETS        | NUMBER        | Total buffer gets                                   |
| DISK_READS         | NUMBER        | Total disk reads                                    |
| DIRECT_WRITES      | NUMBER        | Total direct writes                                 |
| ROWS_PROCESSED     | NUMBER        | Total rows processed                                |
| FETCHES            | NUMBER        | Total number of fetches                             |
| END_OF_FETCH_COUNT | NUMBER        | Total number of full fetches                        |

## 3.2.209. DBA\_SQL\_PROFILES

displays information about SQL profiles currently created for specific SQL statements.

- Columns

| Column Name    | Data Type     | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|----------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| NAME           | VARCHAR2(128) | Name of the SQL profile                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| CATEGORY       | VARCHAR2(128) | Category of the SQL profile                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| SIGNATURE      | NUMBER        | Unique identifier generated from normalized SQL text                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| SQL_TEXT       | CLOB          | Un-normalized SQL text                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| CREATED        | TIMESTAMP(6)  | 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<br>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. |

## 3.2.210. DBA\_STAT\_EXTENSIONS

displays information about all optimizer statistics extensions in the database.

- See also

[ALL\\_STAT\\_EXTENSIONS](#)

### 3.2.211. DBA\_SUBPARTITION\_TEMPLATES

displays all subpartition templates in the database.

- See also

[ALL\\_SUBPARTITION\\_TEMPLATES](#)

[USER\\_SUBPARTITION\\_TEMPLATES](#)

### 3.2.212. DBA\_SUBPART\_KEY\_COLUMNS

Displays subpartitioning key columns of the partitioned tables or indexes in the database.

- See also

[ALL\\_SUBPART\\_KEY\\_COLUMNS](#)

[USER\\_SUBPART\\_KEY\\_COLUMNS](#)

### 3.2.213. DBA\_SYNONYMS

Displays synonyms in the database.

- See also

[ALL\\_SYNONYMS](#)

[USER\\_SYNONYMS](#)

### 3.2.214. DBA\_SYS\_PRIVS

Displays system privileges granted to users and roles in the database.

- Columns

| Column Name | Data Type    | Description                                        |
|-------------|--------------|----------------------------------------------------|
| GRANTEE     | VARCHAR(128) | Name of the user to whom the privilege was granted |
| PRIVILEGE   | VARCHAR(40)  | Name of the privilege                              |

| Column Name  | Data Type  | Description                                                                                      |
|--------------|------------|--------------------------------------------------------------------------------------------------|
| ADMIN_OPTION | VARCHAR(3) | Whether or not the privilege was granted with ADMIN OPTION<br><br>- YES<br><br>- NO<br><br>- ERR |

- See also

[USER\\_SYS\\_PRIVS](#)

### 3.2.215. DBA\_TABLES

Displays all tables in the database.

- See also

[ALL\\_TABLES](#)

[USER\\_TABLES](#)

### 3.2.216. DBA\_TABLESPACES

Displays all tablespaces in the database.

- Columns

| Column Name    | Data Type    | Description                                      |
|----------------|--------------|--------------------------------------------------|
| TABSPACE_NAME  | VARCHAR(128) | Name of the tablespace that contains the table   |
| TS_ID          | NUMBER       | ID of the tablespace                             |
| DATAFILE_COUNT | NUMBER       | The number of data files                         |
| BLOCK_SIZE     | NUMBER       | Size of the tablespace block                     |
| NEXT_EXTENT    | NUMBER       | Size of the next extent                          |
| STATUS         | VARCHAR(9)   | Current status of the tablespace<br><br>- ONLINE |

| Column Name              | Data Type  | Description                                                                              |
|--------------------------|------------|------------------------------------------------------------------------------------------|
|                          |            | - OFFLINE                                                                                |
| CONTENTS                 | VARCHAR(9) | Contents of the tablespace<br><br>- TEMPORARY<br><br>- PERMANENT<br><br>- UNDO           |
| LOGGING                  | VARCHAR(9) | Whether logging is enabled or not<br><br>- YES<br><br>- NO                               |
| FORCE_LOGGING            | VARCHAR(3) | Whether force logging is enabled or not<br><br>- YES<br><br>- NO                         |
| ALLOCATION_TYPE          | VARCHAR(7) | Type of extent allocation in effect for the tablespace:<br><br>- SYSTEM<br><br>- UNIFORM |
| ENCRYPTED                | VARCHAR(3) | Whether tablespace encryption is enabled or not<br><br>- YES<br><br>- NO                 |
| INITIAL_EXTENT           | NUMBER     | Default initial extent size                                                              |
| MIN_EXTENTS              | NUMBER     | Default minimum number of extents                                                        |
| MAX_EXTENTS              | NUMBER     | Default maximum number of extents                                                        |
| EXTENT_MANAGEMENT        | VARCHAR(5) | Type of extent management<br><br>- LOCAL                                                 |
| SEGMENT_SPACE_MANAGEMENT | VARCHAR(4) | Type of segment space management<br><br>- AUTO                                           |
| BIGFILE                  | VARCHAR(3) | Whether tablespace type is bigfile or not<br><br>- YES                                   |

| Column Name              | Data Type   | Description                                                                                                                            |
|--------------------------|-------------|----------------------------------------------------------------------------------------------------------------------------------------|
|                          |             | - NO                                                                                                                                   |
| DEF_INMEMORY             | VARCHAR(8)  | Default value of inmemory is enabled or not for tablespace<br>- ENABLED<br>- DISABLED                                                  |
| DEF_INMEMORY_PRIORITY    | VARCHAR(8)  | Default value of inmemory priority<br>- NONE : DEFAULT<br>- LOW<br>- MEDIUM<br>- HIGH<br>- CRITICAL<br>- NULL                          |
| DEF_INMEMORY_DISTRIBUTE  | VARCHAR(15) | Default value of inmemory distribution method<br>- AUTO : DEFAULT<br>- BY ROWID RANGE<br>- BY PARTITION<br>- BY SUBPARTITION<br>- NULL |
| DEF_INMEMORY_COMPRESSION | VARCHAR(14) | Default value of inmemory compression level<br>- NO MEMCOMPRESS : DEFAULT<br>- FOR QUERY LOW<br>- FOR QUERY HIGH<br>- NULL             |
| DEF_INMEMORY_DUPLICATE   | VARCHAR(13) | Default value of inmemory duplicate method<br>- NO DUPLICATE : DEFAULT<br>- DUPLICATE<br>- DUPLICATE ALL                               |

| Column Name | Data Type | Description |
|-------------|-----------|-------------|
|             |           | - NULL      |

- See also

[USER\\_TABLESPACES](#)

### 3.2.217. DBA\_TAB\_COMMENTS

Displays comments on all tables and views in the database.

- See also

[ALL\\_TAB\\_COMMENTS](#)

### 3.2.218. DBA\_TAB\_HISTGRM\_PENDING\_STATS

Displays histogram statistics of columns in pending state.

- See also

[ALL\\_TAB\\_HISTGRM\\_PENDING\\_STATS](#)

[USER\\_TAB\\_HISTGRM\\_PENDING\\_STATS](#)

### 3.2.219. DBA\_TAB\_IDENTITY\_COLS

Displays identity columns of all tables.

- See also

[ALL\\_TAB\\_IDENTITY\\_COLS](#)

[USER\\_TAB\\_IDENTITY\\_COLS](#)

### 3.2.220. DBA\_TAB\_PENDING\_STATS

Displays pending statistics for the tables in the database.

- See also

[ALL\\_TAB\\_PENDING\\_STATS](#)

[USER\\_TAB\\_PENDING\\_STATS](#)

### 3.2.221. DBA\_TBL\_COLS

Displays columns of all tables and views in the database.

- See also

[ALL\\_TBL\\_COLS](#)

[USER\\_TBL\\_COLS](#)

### 3.2.222. DBA\_TBL\_COLUMNS

Displays columns of all tables and views in the database.

- See also

[ALL\\_TBL\\_COLUMNS](#)

[ALL\\_TBL\\_COLUMNS\\_WITH\\_DNO](#)

[USER\\_TBL\\_COLUMNS](#)

### 3.2.223. DBA\_TBL\_COL\_STATISTICS

Displays statistics and a histogram of columns by extracting them from DBA\_TBL\_COLUMNS.

- See also

[ALL\\_TBL\\_COL\\_STATISTICS](#)

[USER\\_TBL\\_COL\\_STATISTICS](#)

### 3.2.224. DBA\_TBL\_HISTOGRAMS

Displays histogram information on tables and views in the database.

- See also

[ALL\\_TBL\\_HISTOGRAMS](#)

[USER\\_TBL\\_HISTOGRAMS](#)

### 3.2.225. DBA\_TBL\_PARTITIONS

Displays partition information of all tables in the database.

- See also

[ALL\\_TBL\\_PARTITIONS](#)

[USER\\_TBL\\_PARTITIONS](#)

### 3.2.226. DBA\_TBL\_PRIVS

Displays object privileges granted on all tables in the database.

- See also

[ALL\\_TBL\\_PRIVS](#)

[USER\\_TBL\\_PRIVS](#)

### 3.2.227. DBA\_TBL\_SUBPARTITIONS

Displays subpartitions of all tables in the database.

- See also

[ALL\\_TBL\\_SUBPARTITIONS](#)

### 3.2.228. DBA\_TEMP\_FILES

Displays information of the temporary data files.

- Columns

| Column Name     | Data Type    | Description                                             |
|-----------------|--------------|---------------------------------------------------------|
| FILE_NAME       | VARCHAR(256) | Name of the temporary data file                         |
| FILE_ID         | NUMBER       | ID of the data file                                     |
| TABLESPACE_NAME | VARCHAR(128) | Name of the tablespace where the data file is contained |
| BYTES           | NUMBER       | Size of the data file in bytes                          |
| BLOCKS          | NUMBER       | Size of the data file in blocks                         |

| Column Name    | Data Type  | Description                                                                                                  |
|----------------|------------|--------------------------------------------------------------------------------------------------------------|
| STATUS         | CHAR(9)    | Status of the data file<br><br>- UNUSED<br><br>- ONLINE<br><br>- OFFLINE<br><br>- RESETLOG<br><br>- RECOVERY |
| RELATIVE_FNO   | NUMBER     | Relative data file number                                                                                    |
| AUTOEXTENSIBLE | VARCHAR(3) | Auto extensible indicator<br><br>- YES<br><br>- NO                                                           |
| MAXBYTES       | NUMBER     | Maximum data file size in bytes                                                                              |
| MAXBLOCKS      | NUMBER     | Maximum data file size in blocks                                                                             |
| INCREMENT_BY   | NUMBER     | Auto increment value                                                                                         |

### 3.2.229. DBA\_TRIGGERS

Displays all triggers in the database.

- See also

[ALL\\_TRIGGERS](#)

[USER\\_TRIGGERS](#)

### 3.2.230. DBA\_TRIGGER\_COLS

Displays column information of all triggers in the database. The columns are the same as those in ALL\_TRIGGER\_COLS.

- See also

[ALL\\_TRIGGER\\_COLS](#)

[USER\\_TRIGGER\\_COLS](#)

### 3.2.231. DBA\_TS\_QUOTAS

Describe tablespace quotas for all user

- Columns

| Column Name     | Data Type    | Description                             |
|-----------------|--------------|-----------------------------------------|
| TABLESPACE_NAME | VARCHAR(128) | Name of the tablespace                  |
| USERNAME        | VARCHAR(128) | Name of the user                        |
| BYTES           | NUMBER       | Number of bytes charged to the user     |
| MAX_BYTES       | NUMBER       | User's quota in bytes                   |
| BLOCKS          | NUMBER       | Number of blocks charged to the user    |
| MAX_BLOCKS      | NUMBER       | User's quota in blocks                  |
| DROPPED         | VARCHAR(3)   | Whether the tablespace has been dropped |

### 3.2.232. DBA\_TYPES

Displays all types.

- See also

[ALL\\_TYPES](#)

[USER\\_TYPES](#)

### 3.2.233. DBA\_TYPE\_ATTRS

Displays all the attributes of this type.

- See also

[ALL\\_TYPE\\_ATTRS](#)

[USER\\_TYPE\\_ATTRS](#)

### 3.2.234. DBA\_TYPE\_METHODS

Displays all the methods of this type.

- See also

[ALL\\_TYPE\\_METHODS](#)

[USER\\_TYPE\\_METHODS](#)

### 3.2.235. DBA\_UNDO\_EXTENTS

Displays information about the extents in all undo segments.

- Columns

| Column Name | Data Type   | Description                                  |
|-------------|-------------|----------------------------------------------|
| USGMT_ID    | NUMBER      | ID of the undo segment containing the extent |
| EXTNO       | NUMBER      | Extent number                                |
| TS_ID       | NUMBER      | ID of the tablespace containing the extent   |
| TS_NAME     | VARCHAR     | Name of the tablespace containing the extent |
| FILE_ID     | NUMBER      | ID of the file containing the extent         |
| BLOCK_ID    | NUMBER      | Starting block number of the extent          |
| BYTES       | NUMBER      | Size of the extent in bytes                  |
| BLOCKS      | NUMBER      | Size of the extent in blocks                 |
| STATUS      | VARCHAR(30) | Current status of the extent                 |
| IS_CURRENT  | VARCHAR(12) | Whether the extent is current extent or not  |

### 3.2.236. DBA\_UNDO\_SGMTS

Displays undo segments in the database.

- Columns

| Column Name     | Data Type    | Description                                                                                                     |
|-----------------|--------------|-----------------------------------------------------------------------------------------------------------------|
| SEGMENT_ID      | NUMBER       | ID of the segment                                                                                               |
| TABLESPACE_NAME | VARCHAR(128) | Name of the undo tablespace                                                                                     |
| FILE_ID         | NUMBER       | ID of the file containing the segment header                                                                    |
| BLOCK_ID        | NUMBER       | ID of the block containing the segment header                                                                   |
| STATUS          | VARCHAR(15)  | Status of the segment<br><br>- ONLINE<br><br>- OFFLINE<br><br>- OFFLINE PENDING<br><br>- DROPPED (Never reused) |
| RELATIVE_FNO    | NUMBER       | Relative file number of the segment header                                                                      |

### 3.2.237. DBA\_UNUSED\_COL\_TABS

Displays all unused columns in the database.

- See also

[ALL\\_UNUSED\\_COL\\_TABS](#)

[USER\\_UNUSED\\_COL\\_TABS](#)

### 3.2.238. DBA\_UPDATABLE\_COLUMNS

Displays all join view columns that the DBA can update.

- See also

[ALL\\_UPDATABLE\\_COLUMNS](#)

[USER\\_UPDATABLE\\_COLUMNS](#)

### 3.2.239. DBA\_USERS

Displays all users in the database.

- Columns

| Column Name             | Data Type    | Description                                                                                                                                                                                                                                      |
|-------------------------|--------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| USERNAME                | VARCHAR(128) | Name of the user                                                                                                                                                                                                                                 |
| USER_ID                 | NUMBER       | ID of the user                                                                                                                                                                                                                                   |
| PASSWORD                | VARCHAR(24)  | Password of the user (Password is encrypted.)                                                                                                                                                                                                    |
| ACCOUNT_STATUS          | VARCHAR(30)  | Account status<br><br>- LOCKED<br><br>- EXPIRED<br><br>- LOCKED(TIMED)<br><br>- EXPIRED(GRACE)<br><br>- EXPIRED & LOCKED(TIMED)<br><br>- EXPIRED(GRACE) & LOCKED(TIMED)<br><br>- EXPIRED(GRACE) & LOCKED<br><br>- EXPIRED & LOCKED<br><br>- OPEN |
| LOCK_DATE               | DATE         | Date that the account was locked                                                                                                                                                                                                                 |
| EXPIRY_DATE             | DATE         | Date that the account was expired                                                                                                                                                                                                                |
| DEFAULT_TABLESPACE      | VARCHAR(128) | Default tablespace for data                                                                                                                                                                                                                      |
| CREATED                 | DATE         | Date that the user was created                                                                                                                                                                                                                   |
| PROFILE                 | VARCHAR(255) | Profile assigned to the user                                                                                                                                                                                                                     |
| DEFAULT_TEMP_TABLESPACE | VARCHAR(128) | Default temporary tablespace for data                                                                                                                                                                                                            |
| AUTHENTICATION_TYPE     | VARCHAR(8)   | Type of the authentication process for the user<br><br>- PASSWORD<br><br>- EXTERNAL                                                                                                                                                              |
| LAST_LOGIN              | TIMESTAMP(9) | The time of last login                                                                                                                                                                                                                           |

- See also

[ALL\\_USERS](#)

[USER\\_USERS](#)

### 3.2.240. DBA\_VIEWS

Displays all views in the database.

- See also

[ALL\\_VIEWS](#)

[ALL\\_VIEWS\\_INTERNAL](#)

[USER\\_VIEWS](#)

### 3.2.241. DBMSHSXP\_SQL\_PROFILE\_ATTR

displays outline hints that sql profile defines.

- Columns

| Column Name  | Data Type     | Description             |
|--------------|---------------|-------------------------|
| PROFILE_NAME | VARCHAR2(128) | Name of the SQL profile |
| COMP_DATA    | CLOB          | SQL profile attribute   |

### 3.2.242. DICTIONARY

Displays comments on tables and views in the data dictionary.

- Columns

| Column Name | Data Type     | Description                   |
|-------------|---------------|-------------------------------|
| TABLE_NAME  | VARCHAR(128)  | Name of the table or view     |
| COMMENTS    | VARCHAR(4000) | Comments on the table or view |

### 3.2.243. NLS\_DATABASE\_PARAMETERS

Displays NLS parameters of the current database.

- Columns

| Column Name | Data Type   | Description            |
|-------------|-------------|------------------------|
| PARAMETER   | VARCHAR(64) | Name of the parameter  |
| VALUE       | VARCHAR(64) | Value of the parameter |

### 3.2.244. NLS\_SESSION\_PARAMETERS

Displays NLS parameters of the current user session.

- Columns

| Column Name | Data Type   | Description            |
|-------------|-------------|------------------------|
| PARAMETER   | VARCHAR(64) | Name of the parameter  |
| VALUE       | VARCHAR(64) | Value of the parameter |

### 3.2.245. PUBLICSYN

Displays all public synonyms in the database.

- Columns

| Column Name | Data Type    | Description         |
|-------------|--------------|---------------------|
| SYN_NAME    | VARCHAR(128) | Name of the synonym |
| OBJ_OWNER   | VARCHAR(128) | Owner of the object |
| OBJ_NAME    | VARCHAR(257) | Name of the object  |
| OBJ_TYPE    | VARCHAR(9)   | Type of the object  |

## 3.2.246. REDACTION\_COLUMNS

Describes all redacted columns in the database

- Columns

| Column Name            | Data Type      | Description                                                                                                                                                                                                                                                                                                                                                                     |
|------------------------|----------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| OBJECT_OWNER           | VARCHAR2(128)  | Owner of the object that is redacted                                                                                                                                                                                                                                                                                                                                            |
| OBJECT_NAME            | VARCHAR2(128)  | Name of the object that is redacted                                                                                                                                                                                                                                                                                                                                             |
| COLUMN_NAME            | VARCHAR2(128)  | Name of the column that is redacted                                                                                                                                                                                                                                                                                                                                             |
| FUNCTION_TYPE          | VARCHAR2(27)   | Redaction function for this column                                                                                                                                                                                                                                                                                                                                              |
| FUNCTION_PARAMETERS    | VARCHAR2(1000) | Redaction function_parameters for this column                                                                                                                                                                                                                                                                                                                                   |
| REGEXP_PATTERN         | VARCHAR2(512)  | Regular expression pattern to search for                                                                                                                                                                                                                                                                                                                                        |
| REGEXP_REPLACE_STRING  | VARCHAR2(4000) | Replacement string                                                                                                                                                                                                                                                                                                                                                              |
| REGEXP_POSITION        | NUMBER         | Giving the position where the search should begin (counting from 1)                                                                                                                                                                                                                                                                                                             |
| REGEXP_OCCURRENCE      | NUMBER         | Specify which occurrence of the regular expression match to replace<br><br>- 0 : To replace all occurrences of the match<br><br>- N (Positive Integer) : To replace the Nth occurrence of the match                                                                                                                                                                             |
| REGEXP_MATCH_PARAMETER | VARCHAR2(10)   | Change the default matching behavior using a combination of specific values.<br><br>- i : Case-insensitive matching<br><br>- c : Case-sensitive matching<br><br>- n : Matches newline characters (\n)<br><br>- m : ^ and \$ match the start and end of each line (multiple lines mode)<br><br>- x : Ignore whitespace and allow comments in the pattern (extended/verbose mode) |

| Column Name        | Data Type      | Description                                                                         |
|--------------------|----------------|-------------------------------------------------------------------------------------|
| COLUMN_DESCRIPTION | VARCHAR2(4000) | User-provided description of the redaction function that is performed on the column |

### 3.2.247. REDACTION\_COLUMNS\_DBMS\_ERRLOG

Logs DML errors that occur on columns protected by data redaction policies (Currently not supported)

- Columns

| Column Name  | Data Type     | Description                          |
|--------------|---------------|--------------------------------------|
| OBJECT_OWNER | VARCHAR2(128) | Owner of the object that is redacted |
| OBJECT_NAME  | VARCHAR2(128) | Name of the object that is redacted  |
| COLUMN_NAME  | VARCHAR2(128) | Name of the column that is redacted  |

### 3.2.248. REDACTION\_POLICIES

Displays all redaction policies in the database.

- Columns

| Column Name        | Data Type      | Description                                                      |
|--------------------|----------------|------------------------------------------------------------------|
| OBJECT_OWNER       | VARCHAR2(128)  | Owner of the object with the policy                              |
| OBJECT_NAME        | VARCHAR2(128)  | Name of the object with the policy                               |
| POLICY_NAME        | VARCHAR2(128)  | Name of the policy                                               |
| EXPRESSION         | VARCHAR2(4000) | Expression for this policy                                       |
| ENABLE             | VARCHAR2(3)    | Indicates whether the policy is enabled<br><br>- YES<br><br>- NO |
| POLICY_DESCRIPTION | VARCHAR2(4000) | Description of the policy                                        |

### 3.2.249. REDACTION\_VALUES\_FOR\_TYPE\_FULL

Shows all of the current values for full redaction.

- Columns

| Column Name                    | Data Type     | Description                                                             |
|--------------------------------|---------------|-------------------------------------------------------------------------|
| NUMBER_VALUE                   | NUMBER        | Redaction result for full redaction on NUMBER columns                   |
| BINARY_FLOAT_VALUE             | BINARY_FLOAT  | Redaction result for full redaction on BINARY_FLOAT columns             |
| BINARY_DOUBLE_VALUE            | BINARY_DOUBLE | Redaction result for full redaction on BINARY_DOUBLE columns            |
| CHAR_VALUE                     | VARCHAR2(1)   | Redaction result for full redaction on CHAR columns                     |
| VARCHAR_VALUE                  | VARCHAR2(1)   | Redaction result for full redaction on VARCHAR2 columns                 |
| NCHAR_VALUE                    | NCHAR(1)      | Redaction result for full redaction on NCHAR columns                    |
| NVARCHAR_VALUE                 | NVARCHAR2(1)  | Redaction result for full redaction on NVARCHAR2 columns                |
| DATE_VALUE                     | DATE          | Redaction result for full redaction on DATE columns                     |
| TIMESTAMP_VALUE                | TIMESTAMP(6)  | Redaction result for full redaction on TIMESTAMP columns                |
| TIMESTAMP_WITH_TIME_ZONE_VALUE | TIMESTAMP(6)  | Redaction result for full redaction on TIMESTAMP WITH TIME ZONE columns |
| BLOB_VALUE                     | BLOB          | Redaction result for full redaction on BLOB columns                     |
| CLOB_VALUE                     | CLOB          | Redaction result for full redaction on CLOB columns                     |
| NCLOB_VALUE                    | NCLOB         | Redaction result for full redaction on NCLOB columns                    |

### 3.2.250. ROLE\_ROLE\_PRIVS

Displays the roles granted to other roles which the current user can access.

- Columns

| Column Name  | Data Type    | Description                                                                                  |
|--------------|--------------|----------------------------------------------------------------------------------------------|
| ROLE         | VARCHAR(128) | Name of the role                                                                             |
| GRANTED_ROLE | VARCHAR(128) | Role that was granted                                                                        |
| ADMIN_OPTION | VARCHAR(3)   | Whether or not the role was assigned with ADMIN OPTION<br><br>- YES<br><br>- NO<br><br>- ERR |

### 3.2.251. ROLE\_SYS\_PRIVS

Displays system privileges granted to roles which the current user can access.

- Columns

| Column Name  | Data Type    | Description                                                                                   |
|--------------|--------------|-----------------------------------------------------------------------------------------------|
| ROLE         | VARCHAR(128) | Name of the role                                                                              |
| PRIVILEGE    | VARCHAR(40)  | Name of the privilege                                                                         |
| ADMIN_OPTION | VARCHAR(3)   | Whether or not the grant was assigned with ADMIN OPTION<br><br>- YES<br><br>- NO<br><br>- ERR |

### 3.2.252. ROLE\_TAB\_PRIVS

Displays object privileges granted to roles which the current user can access.

- Columns

| Column Name | Data Type    | Description      |
|-------------|--------------|------------------|
| ROLE        | VARCHAR(128) | Name of the role |

| Column Name  | Data Type    | Description                                                                                   |
|--------------|--------------|-----------------------------------------------------------------------------------------------|
| OWNER        | VARCHAR(128) | Owner of the object                                                                           |
| TABLE_NAME   | VARCHAR(128) | Name of the object                                                                            |
| COLUMN_NAME  | VARCHAR(128) | Name of the column, if applicable                                                             |
| PRIVILEGE    | VARCHAR(40)  | Object privilege granted to the role                                                          |
| ADMIN_OPTION | VARCHAR(3)   | Whether or not the grant was assigned with ADMIN OPTION<br><br>- YES<br><br>- NO<br><br>- ERR |

### 3.2.253. SESSION\_CONTEXT

Displays context attributes in the current session.

- Columns

| Column Name | Data Type      | Description                            |
|-------------|----------------|----------------------------------------|
| NAMESPACE   | VARCHAR(128)   | The namespace of the context attribute |
| ATTRIBUTE   | VARCHAR(128)   | The name of the attribute              |
| VALUE       | VARCHAR(65532) | The value of the attribute             |

### 3.2.254. SESSION\_PRIVS

Displays the system privileges granted to current session user.

- Columns

| Column Name | Data Type   | Description           |
|-------------|-------------|-----------------------|
| PRIVILEGE   | VARCHAR(40) | Name of the privilege |

### 3.2.255. SESSION\_ROLES

Displays roles granted to the current session user.

- Columns

| Column Name | Data Type    | Description      |
|-------------|--------------|------------------|
| ROLE        | VARCHAR(128) | Name of the role |

### 3.2.256. SYS\_OBJECTS

Displays IDs and types of all schema objects, and IDs of the segment header blocks in the database.

- Columns

| Column Name     | Data Type   | Description                                        |
|-----------------|-------------|----------------------------------------------------|
| OBJECT_TYPE     | VARCHAR(18) | Type of the object                                 |
| OBJECT_TYPE_NO  | NUMBER      | ID of the object's type                            |
| SEGMENT_TYPE_NO | NUMBER      | ID of the segment's type                           |
| OBJECT_ID       | NUMBER      | ID of the object                                   |
| HEADER_FILE     | NUMBER      | ID of the data file containing the segment header  |
| HEADER_BLOCK    | NUMBER      | ID of the disk block containing the segment header |
| TS_NUMBER       | NUMBER      | ID of the tablespace containing the object         |
| SEGMENT_ID      | NUMBER      | ID of the segment                                  |

### 3.2.257. USER\_ARGUMENTS

Displays all parameters of the procedure or function that the current user owns.

- See also

[ALL\\_ARGUMENTS](#)

## 3.2.258. USER\_AUDIT\_TRAIL

Displays audit trails associated with the current user.

- Columns

| Column Name     | Data Type      | Description                                                  |
|-----------------|----------------|--------------------------------------------------------------|
| OS_USERNAME     | VARCHAR(128)   | Operating system user name who was set to be audited         |
| USERNAME        | VARCHAR(128)   | Name of the user who was set to be audited                   |
| USERHOST        | VARCHAR(21)    | Name of the host machine from which the client has connected |
| TIMESTAMP       | VARCHAR(19)    | Audit trail-created time                                     |
| OWNER           | VARCHAR(128)   | Owner of the object associated with the audited action       |
| OBJ_NAME        | VARCHAR(128)   | Name of the object associated with the audited action        |
| SESSIONID       | NUMBER         | Session ID                                                   |
| SERIALNO        | NUMBER         | Sequence number of the session                               |
| ENTRYID         | NUMBER         | Sequence number for each audit trail in the session          |
| STATEMENTID     | NUMBER         | ID of the executed statement                                 |
| PRIV_USED       | VARCHAR(80)    | Name of the privilege used to execute the audited action     |
| CLIENT_ID       | VARCHAR(30)    | Client identifier                                            |
| INSTANCE_NUMBER | NUMBER         | Instance number of RDBMS which performed the action          |
| OS_PROCESS      | VARCHAR(16)    | Process ID of RDBMS                                          |
| TRANSACTIONID   | VARCHAR(65532) | Transaction ID that participated in the audited action       |
| TSN             | NUMBER         | TSN of the audited action                                    |
| SQL_TEXT        | LONG           | SQL of the audited action                                    |

- See also

[DBA\\_AUDIT\\_TRAIL](#)

### 3.2.259. USER\_CATALOG

Displays all tables, views, synonyms, and sequences that the current user owns.

- See also

[ALL\\_CATALOG](#)

[DBA\\_CATALOG](#)

### 3.2.260. USER\_COLL\_TYPES

Displays all the collection types that the current user owns.

- See also

[ALL\\_COLL\\_TYPES](#)

[DBA\\_COLL\\_TYPES](#)

### 3.2.261. USER\_COL\_COMMENTS

Displays comments on all columns in the tables or views that the current user owns.

- See also

[ALL\\_COL\\_COMMENTS](#)

[DBA\\_COL\\_COMMENTS](#)

### 3.2.262. USER\_COL\_PENDING\_STATS

Displays statistics and a histogram of columns in pending state current user owns.

- See also

[ALL\\_COL\\_PENDING\\_STATS](#)

[DBA\\_COL\\_PENDING\\_STATS](#)

### 3.2.263. USER\_COL\_PRIVS

Displays object privileges of the columns for which the current user is the object owner, grantor, or grantee.

- See also

[ALL\\_COL\\_PRIVS](#)

[DBA\\_COL\\_PRIVS](#)

### 3.2.264. USER\_COL\_PRIVS\_MADE

Displays columns for which the current user is the grantor.

- See also

[ALL\\_COL\\_PRIVS\\_MADE](#)

### 3.2.265. USER\_COL\_PRIVS\_RECD

Displays columns for which the current user is the grantee.

- See also

[ALL\\_COL\\_PRIVS\\_RECD](#)

### 3.2.266. USER\_CONSTRAINTS

Displays constraints defined in the table that the current user owns.

- See also

[ALL\\_CONSTRAINTS](#)

[DBA\\_CONSTRAINTS](#)

### 3.2.267. USER\_CONS\_COLUMNS

Displays columns that are owned by the current user and that are specified in integrity constraint definitions.

- See also

[ALL\\_CONS\\_COLUMNS](#)

[DBA\\_CONS\\_COLUMNS](#)

### 3.2.268. USER\_DB\_LINKS

Displays information for the database links that the current user owns.

- See also

[ALL\\_DB\\_LINKS](#)

[DBA\\_DB\\_LINKS](#)

### 3.2.269. USER\_DEPENDENCIES

Displays dependencies between procedures, functions, packages, package bodies, triggers, and views that the current user owns.

- See also

[ALL\\_DEPENDENCIES](#)

[DBA\\_DEPENDENCIES](#)

### 3.2.270. USER\_ENCRYPTED\_COLUMNS

Displays information of all encrypted columns that the current user owns. The columns are the same as those in ALL\_ENCRYPTED\_COLUMNS(except for OWNER)

- See also

[ALL\\_ENCRYPTED\\_COLUMNS](#)

[DBA\\_ENCRYPTED\\_COLUMNS](#)

### 3.2.271. USER\_ERRORS

Displays the recent errors in every view, procedure, function, package, package body, trigger, library, and java object owned by current user. The columns are the same as those in ALL\_ERRORS.(except for OWNER)

- See also

[ALL\\_ERRORS](#)

[DBA\\_ERRORS](#)

### 3.2.272. USER\_EXTENTS

Displays information of the extents contained in the segments that the current user owns.

- Columns

| Column Name     | Data Type    | Description                                                           |
|-----------------|--------------|-----------------------------------------------------------------------|
| SEGMENT_NAME    | VARCHAR(128) | Name of the segment containing the extent                             |
| PARTITION_NAME  | VARCHAR(128) | Partition name of the object (Null value for non-partitioned objects) |
| SEGMENT_TYPE    | VARCHAR(18)  | Type of the segment                                                   |
| TABLESPACE_NAME | VARCHAR(30)  | Name of the tablespace containing the extent                          |
| EXTENT_ID       | NUMBER       | ID of the extent                                                      |
| BYTES           | NUMBER       | Size of the extent in bytes                                           |
| BLOCKS          | NUMBER       | Size of the extent in blocks                                          |

- See also

[DBA\\_EXTENTS](#)

### 3.2.273. USER\_EXTERNAL\_LOCATIONS

Displays all external table locations that the current user owns.

- See also

[ALL\\_EXTERNAL\\_LOCATIONS](#)

[DBA\\_EXTERNAL\\_LOCATIONS](#)

### 3.2.274. USER\_EXTERNAL\_TABLES

Displays all external tables that the current user owns.

- See also

[ALL\\_EXTERNAL\\_TABLES](#)

[DBA\\_EXTERNAL\\_TABLES](#)

### 3.2.275. USER\_FREE\_SPACE

Displays information of the extents available for the tablespaces that the current user can access.

- Columns

| Column Name     | Data Type    | Description                                  |
|-----------------|--------------|----------------------------------------------|
| TABLESPACE_NAME | VARCHAR(128) | Name of the tablespace containing the extent |
| FILE_ID         | NUMBER       | ID of the file containing the extent         |
| BLOCK_ID        | NUMBER       | Starting block number of the extent          |
| BYTES           | NUMBER       | Size of the extent in bytes                  |
| BLOCKS          | NUMBER       | Size of the extent in blocks                 |
| RELATIVE_FNO    | NUMBER       | Relative file number of the extent           |

- See also

[DBA\\_FREE\\_SPACE](#)

### 3.2.276. USER\_IDX\_COLUMNS

Displays index columns of all tables that the current user owns.

- See also

[ALL\\_IDX\\_COLUMNS](#)

[DBA\\_IDX\\_COLUMNS](#)

### 3.2.277. USER\_IDX\_EXPRESSIONS

Displays expressions on function-based index columns of all tables that the current user owns.

- See also

[ALL\\_IDX\\_EXPRESSIONS](#)

[DBA\\_IDX\\_EXPRESSIONS](#)

### 3.2.278. USER\_IDX\_PARTITIONS

displays partitions of the indexes that the current user owns.

- See also

[ALL\\_IDX\\_PARTITIONS](#)

[DBA\\_IDX\\_PARTITIONS](#)

### 3.2.279. USER\_INDEXES

Displays indexes of all tables that the current user owns.

- See also

[ALL\\_INDEXES](#)

[DBA\\_INDEXES](#)

### 3.2.280. USER\_IND\_PENDING\_STATS

Displays indexes of all tables that the current user owns.

- See also

[ALL\\_IND\\_PENDING\\_STATS](#)

[DBA\\_IND\\_PENDING\\_STATS](#)

### 3.2.281. USER\_JAVA\_ARGUMENTS

Displays all parameters that the current user owns.

- See also

[ALL\\_JAVA\\_ARGUMENTS](#)

[DBA\\_JAVA\\_ARGUMENTS](#)

### 3.2.282. USER\_JAVA\_CLASSES

Displays all classes that the current user owns.

- See also

[ALL\\_JAVA\\_CLASSES](#)

[DBA\\_JAVA\\_CLASSES](#)

### 3.2.283. USER\_JAVA\_FIELDS

Displays all fields that the current user owns.

- See also

[ALL\\_JAVA\\_FIELDS](#)

[DBA\\_JAVA\\_FIELDS](#)

### 3.2.284. USER\_JAVA\_METHODS

Displays all methods that the current user owns.

- See also

[ALL\\_JAVA\\_METHODS](#)

[DBA\\_JAVA\\_METHODS](#)

### 3.2.285. USER\_JOBS

Displays all jobs that the current user owns.

- See also

[ALL\\_JOBS](#)

[DBA\\_JOBS](#)

### 3.2.286. USER\_JOBS\_WITH\_NAME

Displays all jobs with job name that the current user owns.

- See also

[ALL\\_JOBS\\_WITH\\_NAME](#)

[DBA\\_JOBS\\_WITH\\_NAME](#)

### 3.2.287. USER\_LIBRARIES

Displays all libraries that the current user owns.

- See also

[ALL\\_LIBRARIES](#)

[DBA\\_LIBRARIES](#)

### 3.2.288. USER\_LOBS

Displays large objects (LOB) which belong to all tables that the current user owns.

- See also

[ALL\\_LOBS](#)

[DBA\\_LOBS](#)

### 3.2.289. USER\_LOB\_PARTITIONS

Displays all LOB partitions which belong to all tables that the current user owns.

- See also

[ALL\\_LOB\\_PARTITIONS](#)

[DBA\\_LOB\\_PARTITIONS](#)

### 3.2.290. USER\_LOB\_SUBPARTITIONS

Displays all LOB subpartitions which belong to all tables that the current user owns.

- See also

[ALL\\_LOB\\_SUBPARTITIONS](#)

[DBA\\_LOB\\_SUBPARTITIONS](#)

### 3.2.291. USER\_LOG\_GROUPS

Displays the log group definitions on the tables owned by the current user.

- See also

[ALL\\_LOG\\_GROUPS](#)

[DBA\\_LOG\\_GROUPS](#)

### 3.2.292. USER\_LOG\_GROUP\_COLUMNS

Displays columns that are owned by the current user and that are specified in log groups.

- See also

[ALL\\_LOG\\_GROUP\\_COLUMNS](#)

[DBA\\_LOG\\_GROUP\\_COLUMNS](#)

### 3.2.293. USER\_METHOD\_PARAMS

Displays the method parameters of the type that the current user owns.

- See also

[ALL\\_METHOD\\_PARAMS](#)

[DBA\\_METHOD\\_PARAMS](#)

### 3.2.294. USER\_METHOD\_RESULTS

Displays the result of the method of this type that the current user owns.

- See also

[ALL\\_METHOD\\_RESULTS](#)

[DBA\\_METHOD\\_RESULTS](#)

### 3.2.295. USER\_MVIEWS

Displays all materialized views that the current user owns. The columns are the same as those in ALL\_MVIEWS.

- See also

[ALL\\_MVIEWS](#)

[DBA\\_MVIEWS](#)

### 3.2.296. USER\_MVIEW\_COMMENTS

Displays comments on all materialized views that the current user owns.

- See also

[ALL\\_MVIEW\\_COMMENTS](#)

[DBA\\_MVIEW\\_COMMENTS](#)

### 3.2.297. USER\_MVIEW\_DETAIL\_RELATIONS

Describes all such detail relations defined for materialized views in the database owned by the current user.

- See also

[ALL\\_MVIEW\\_DETAIL\\_RELATIONS](#)

[DBA\\_MVIEW\\_DETAIL\\_RELATIONS](#)

### 3.2.298. USER\_MVIEW\_LOGS

Displays all materialized view logs that the current user owns. The columns are the same as those in ALL\_MVIEW\_LOGS.

- See also

[ALL\\_MVIEW\\_LOGS](#)

[DBA\\_MVIEW\\_LOGS](#)

### 3.2.299. USER\_MVIEW\_REFRESH\_TIMES

Describes refresh times of the materialized views owned by the current user.

- See also

[ALL\\_MVIEW\\_REFRESH\\_TIMES](#)

[DBA\\_MVIEW\\_REFRESH\\_TIMES](#)

## 3.2.300. USER\_OBJECTS

Displays all objects that the current user owns.

- See also

[ALL\\_OBJECTS](#)

[DBA\\_OBJECTS](#)

## 3.2.301. USER\_OBJ\_AUDIT\_OPTS

Displays the current user's own objects which are being audited.

- Columns

| Column Name | Data Type    | Description                                                                                   |
|-------------|--------------|-----------------------------------------------------------------------------------------------|
| OBJECT_NAME | VARCHAR(128) | Name of the object                                                                            |
| OBJECT_TYPE | VARCHAR(9)   | Type of the object<br>- TABLE<br>- VIEW<br>- SEQUENCE<br>- PROCEDURE<br>- DIRECTORY<br>- JAVA |
| ALT         | VARCHAR(5)   | ALTER auditing option for success or failure                                                  |
| DEL         | VARCHAR(5)   | DELETE auditing option for success or failure                                                 |
| IND         | VARCHAR(5)   | INDEX auditing option for success or failure                                                  |
| INS         | VARCHAR(5)   | INSERT auditing option for success or failure                                                 |
| LOC         | VARCHAR(5)   | LOCK auditing option for success or failure                                                   |
| SEL         | VARCHAR(5)   | SELECT auditing option for success or failure                                                 |
| UPD         | VARCHAR(5)   | UPDATE auditing option for success or failure                                                 |
| REF         | VARCHAR(5)   | REFERENCE auditing option for success or failure                                              |

| Column Name | Data Type  | Description                                    |
|-------------|------------|------------------------------------------------|
| EXE         | VARCHAR(5) | EXECUTE auditing option for success or failure |
| REA         | VARCHAR(5) | READ auditing option for success or failure    |
| WRI         | VARCHAR(5) | WRITE auditing option for success or failure   |

- See also

[DBA\\_OBJ\\_AUDIT\\_OPTS](#)

### 3.2.302. USER\_PACKAGE\_COLL\_TYPES

Displays types of the package that the current user owns.

- See also

[ALL\\_PACKAGE\\_COLL\\_TYPES](#)

[DBA\\_PACKAGE\\_COLL\\_TYPES](#)

### 3.2.303. USER\_PACKAGE\_MEMBER\_TYPES

Displays types of the package that the current user owns.

- See also

[ALL\\_PACKAGE\\_MEMBER\\_TYPES](#)

[DBA\\_PACKAGE\\_MEMBER\\_TYPES](#)

### 3.2.304. USER\_PACKAGE\_TYPES

Displays types of the package that the current user owns.

- See also

[ALL\\_PACKAGE\\_TYPES](#)

[DBA\\_PACKAGE\\_TYPES](#)

### 3.2.305. USER\_PARALLEL\_EXECUTE\_CHUNKS

Displays the chunks for all tasks that created by current user.

- Columns

| Column Name   | Data Type      | Description                                                                                                |
|---------------|----------------|------------------------------------------------------------------------------------------------------------|
| CHUNK_ID      | NUMBER         | ID for the chunk                                                                                           |
| TASK_NAME     | VARCHAR2(128)  | Name of the task                                                                                           |
| STATUS        | VARCHAR2(20)   | Status of the chunk<br><br>- UNASSIGNED<br><br>- ASSIGNED<br><br>- PROCESSED<br><br>- PROCESSED_WITH_ERROR |
| START_ROWID   | ROWID          | Rowid for the first row in the chunk                                                                       |
| END_ROWID     | ROWID          | Rowid for the end 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 end 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         | Returned error code for the execution of the chunk if the STATUS column is PROCESSED_WITH_ERROR            |
| ERROR_MESSAGE | VARCHAR2(4000) | Returned error message for the execution of the chunk if the STATUS column is PROCESSED_WITH_ERROR         |

- See also

[DBA\\_PARALLEL\\_EXECUTE\\_CHUNKS](#)

### 3.2.306. USER\_PARALLEL\_EXECUTE\_TASKS

Displays all tasks that created by current user.

- Columns

| Column Name   | Data Type      | Description                                                                                                                                                                                            |
|---------------|----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| TASK_NAME     | VARCHAR2(128)  | Name of the task                                                                                                                                                                                       |
| CHUNK_TYPE    | VARCHAR2(12)   | Type for the chunk<br><br>- UNDECLARED<br><br>- ROWID_RANGE<br><br>- NUMBER_RANGE                                                                                                                      |
| STATUS        | VARCHAR2(19)   | Status of the task<br><br>- CREATED<br><br>- CHUNKING<br><br>- CHUNKING_FAILED<br><br>- NO_CHUNKS<br><br>- CHUNKED<br><br>- PROCESSING<br><br>- FINISHED<br><br>- FINISHED_WITH_ERROR<br><br>- CRASHED |
| TABLE_OWNER   | VARCHAR2(128)  | Owner of the table which to be chunked                                                                                                                                                                 |
| TABLE_NAME    | VARCHAR2(128)  | Name of the table which to be chunked                                                                                                                                                                  |
| NUMBER_COLUMN | VARCHAR2(128)  | Name of the column holding IDs (only if CHUNK_TYPE is NUMBER_RANGE)                                                                                                                                    |
| TASK_COMMENT  | VARCHAR2(4000) | Task comment                                                                                                                                                                                           |
| JOB_PREFIX    | VARCHAR2(128)  | Prefix of the job name                                                                                                                                                                                 |
| SQL_STMT      | CLOB           | Used in DBMS_PARALLEL_EXECUTE.RUN_TASK                                                                                                                                                                 |
| LANGUAGE_FLAG | NUMBER         | Used in DBMS_PARALLEL_EXECUTE.RUN_TASK                                                                                                                                                                 |

| Column Name                   | Data Type     | Description                            |
|-------------------------------|---------------|----------------------------------------|
| APPLY_CROSSED<br>TION_TRIGGER | VARCHAR2(130) | Used in DBMS_PARALLEL_EXECUTE.RUN_TASK |
| FIRE_APPLY_TRIG<br>GER        | VARCHAR2(130) | Used in DBMS_PARALLEL_EXECUTE.RUN_TASK |
| PARALLEL_LEVEL                | NUMBER        | Used in DBMS_PARALLEL_EXECUTE.RUN_TASK |
| JOB_CLASS                     | VARCHAR2(128) | Used in DBMS_PARALLEL_EXECUTE.RUN_TASK |

- See also

[DBA\\_PARALLEL\\_EXECUTE\\_TASKS](#)

### 3.2.307. USER\_PART\_INDEXES

Displays partitioning information for the partitioned indexes that the current user owns.

- See also

[ALL\\_PART\\_INDEXES](#)

[DBA\\_PART\\_INDEXES](#)

### 3.2.308. USER\_PART\_KEY\_COLUMNS

Displays partitioning key columns for all partitioned tables or indexes that the current user owns.

- See also

[ALL\\_PART\\_KEY\\_COLUMNS](#)

[DBA\\_PART\\_KEY\\_COLUMNS](#)

### 3.2.309. USER\_PART\_LOBS

Displays LOBs which belong to the partitioned tables that the current user owns.

- See also

[ALL\\_PART\\_LOBS](#)

[DBA\\_PART\\_LOBS](#)

### 3.2.310. USER\_PART\_TABLES

Displays partitioning information of the partitioned tables that the current user owns.

- See also

[ALL\\_PART\\_TABLES](#)

[DBA\\_PART\\_TABLES](#)

### 3.2.311. USER\_PROCEduRES

Displays all procedures that the current user owns.

- See also

[ALL\\_PROCEduRES](#)

[DBA\\_PROCEduRES](#)

### 3.2.312. USER\_QUEUES

Describes the operational characteristics of every queue owned by the current user.

- See also

[ALL\\_QUEUES](#)

[DBA\\_QUEUES](#)

### 3.2.313. USER\_QUEUE\_SUBSCRIBERS

Shows the list of subscribers on queues that are under the current user's schema.

- See also

[ALL\\_QUEUE\\_SUBSCRIBERS](#)

[DBA\\_QUEUE\\_SUBSCRIBERS](#)

### 3.2.314. USER\_QUEUE\_TABLES

Describes the queues in the queue tables created in the current user's schema.

- See also

[ALL\\_QUEUE\\_TABLES](#)

[DBA\\_QUEUE\\_TABLES](#)

### 3.2.315. USER\_RECYCLEBIN

Displays information of all temporarily deleted objects that the current user owns. The columns are the same as those in ALL\_RECYCLEBIN.

- See also

[ALL\\_RECYCLEBIN](#)

[DBA\\_RECYCLEBIN](#)

### 3.2.316. USER\_ROLE\_PRIVS

Displays roles granted to the current user.

- Columns

| Column Name  | Data Type    | Description                                                                      |
|--------------|--------------|----------------------------------------------------------------------------------|
| GRANTEE      | VARCHAR(128) | Name of the user or role which was granted the role                              |
| GRANTED_ROLE | VARCHAR(128) | Granted role                                                                     |
| ADMIN_OPTION | VARCHAR(3)   | Whether or not the role was assigned with ADMIN OPTION<br>- YES<br>- NO<br>- ERR |
| DEFAULT_ROLE | VARCHAR(3)   | Whether the role is set to default role for the user<br>- YES<br>- NO<br>- ERR   |

- See also

[DBA\\_ROLE\\_PRIVS](#)

### 3.2.317. USER\_SCHEDULER\_CHAINS

Displays all scheduler chains that the current user owns.

- See also

[ALL\\_SCHEDULER\\_CHAINS](#)

[DBA\\_SCHEDULER\\_CHAINS](#)

### 3.2.318. USER\_SCHEDULER\_JOBS

Displays all scheduler jobs that the current user owns.

- See also

[ALL\\_SCHEDULER\\_JOBS](#)

[DBA\\_SCHEDULER\\_JOBS](#)

### 3.2.319. USER\_SCHEDULER\_JOB\_LOG

Displays all scheduler job log that the current user owns.

- See also

[ALL\\_SCHEDULER\\_JOB\\_LOG](#)

[DBA\\_SCHEDULER\\_JOB\\_LOG](#)

### 3.2.320. USER\_SCHEDULER\_JOB\_RUN\_DETAILS

Displays all scheduler job run details that the current user owns.

- See also

[ALL\\_SCHEDULER\\_JOB\\_RUN\\_DETAILS](#)

[DBA\\_SCHEDULER\\_JOB\\_RUN\\_DETAILS](#)

### 3.2.321. USER\_SCHEDULER\_PROGRAMS

Displays all scheduler programs that the current user owns.

- See also

[ALL\\_SCHEDULER\\_PROGRAMS](#)

[DBA\\_SCHEDULER\\_PROGRAMS](#)

### 3.2.322. USER\_SCHEDULER\_RULES

Displays all scheduler rules that the current user owns.

- See also

[ALL\\_SCHEDULER\\_RULES](#)

[DBA\\_SCHEDULER\\_RULES](#)

### 3.2.323. USER\_SCHEDULER\_RULE\_CTX

Displays all scheduler rule ctx that the current user owns.

- See also

[ALL\\_SCHEDULER\\_RULE\\_CTX](#)

[DBA\\_SCHEDULER\\_RULE\\_CTX](#)

### 3.2.324. USER\_SCHEDULER\_RUNNING\_JOBS

Displays all scheduler running jobs that the current user owns.

- See also

[ALL\\_SCHEDULER\\_RUNNING\\_JOBS](#)

[DBA\\_SCHEDULER\\_RUNNING\\_JOBS](#)

### 3.2.325. USER\_SCHEDULER\_SCHEDULES

Displays all scheduler schedules that the current user owns.

- See also

[ALL\\_SCHEDULER\\_SCHEDULES](#)

[DBA\\_SCHEDULER\\_SCHEDULES](#)

### 3.2.326. USER\_SCHEDULER\_STEPS

Displays all scheduler steps that the current user owns.

- See also

[ALL\\_SCHEDULER\\_STEPS](#)

[DBA\\_SCHEDULER\\_STEPS](#)

### 3.2.327. USER\_SCHEDULER\_STEP\_CTX

Displays all scheduler step context the current user owns.

- See also

[ALL\\_SCHEDULER\\_STEP\\_CTX](#)

[DBA\\_SCHEDULER\\_STEP\\_CTX](#)

### 3.2.328. USER\_SEGMENTS

Displays all segments which belong to the current user.

- See also

[ALL\\_SEGMENTS](#)

[DBA\\_SEGMENTS](#)

### 3.2.329. USER\_SEQUENCES

Displays all sequences that the current user owns.

- See also

[ALL\\_SEQUENCES](#)

[DBA\\_SEQUENCES](#)

### 3.2.330. USER\_SOURCE

Displays source texts of the stored objects in the database such as procedures, functions, package bodies, or triggers.

- See also

[ALL\\_SOURCE](#)

[DBA\\_SOURCE](#)

### 3.2.331. USER\_STAT\_EXTENSIONS

Displays information of all optimizer statistics extensions that the current user owns

- See also

[ALL\\_STAT\\_EXTENSIONS](#)

[DBA\\_STAT\\_EXTENSIONS](#)

### 3.2.332. USER\_SUBPARTITION\_TEMPLATES

displays subaprtition templates that the current user owns.

- See also

[ALL\\_SUBPARTITION\\_TEMPLATES](#)

[DBA\\_SUBPARTITION\\_TEMPLATES](#)

### 3.2.333. USER\_SUBPART\_KEY\_COLUMNS

Displays subpartitioning key columns of the partitioned tables or indexes that the current user owns.

- See also

[ALL\\_SUBPART\\_KEY\\_COLUMNS](#)

[DBA\\_SUBPART\\_KEY\\_COLUMNS](#)

### 3.2.334. USER\_SYNONYMS

Displays synonyms that the current user owns.

- See also

[ALL\\_SYNONYMS](#)

[DBA\\_SYNONYMS](#)

### 3.2.335. USER\_SYS\_PRIVS

Displays system privileges granted to the current user.

- Columns

| Column Name  | Data Type    | Description                                                                                      |
|--------------|--------------|--------------------------------------------------------------------------------------------------|
| USERNAME     | VARCHAR(128) | Name of the current user or PUBLIC                                                               |
| PRIVILEGE    | VARCHAR(40)  | Name of the privilege                                                                            |
| ADMIN_OPTION | VARCHAR(3)   | Whether or not the privilege was granted with ADMIN OPTION<br><br>- YES<br><br>- NO<br><br>- ERR |

- See also

[DBA\\_SYS\\_PRIVS](#)

### 3.2.336. USER\_TABLES

Displays all tables that the current user owns.

- See also

[ALL\\_TABLES](#)

[DBA\\_TABLES](#)

### 3.2.337. USER\_TABLESPACES

Synonym for DBA\_TABLESPACES.

- Columns

| Column Name     | Data Type    | Description                                                                      |
|-----------------|--------------|----------------------------------------------------------------------------------|
| TABSPACE_NAME   | VARCHAR(128) | Name of the tablespace that contains the table                                   |
| TS_ID           | NUMBER       | ID of the tablespace                                                             |
| DATAFILE_COUNT  | NUMBER       | The number of data files                                                         |
| BLOCK_SIZE      | NUMBER       | Size of the tablespace block                                                     |
| NEXT_EXTENT     | NUMBER       | Size of the next extent                                                          |
| STATUS          | VARCHAR(9)   | Current status of the tablespace<br>- ONLINE<br>- OFFLINE                        |
| CONTENTS        | VARCHAR(9)   | Contents of the tablespace<br>- TEMPORARY<br>- PERMANENT<br>- UNDO               |
| LOGGING         | VARCHAR(9)   | Whether logging is enabled or not<br>- YES<br>- NO                               |
| FORCE_LOGGING   | VARCHAR(3)   | Whether force logging is enabled or not<br>- YES<br>- NO                         |
| ALLOCATION_TYPE | VARCHAR(7)   | Type of extent allocation in effect for the tablespace:<br>- SYSTEM<br>- UNIFORM |
| ENCRYPTED       | VARCHAR(3)   | Whether tablespace encryption is enabled or not<br>- YES<br>- NO                 |

| Column Name              | Data Type   | Description                                                                                                                  |
|--------------------------|-------------|------------------------------------------------------------------------------------------------------------------------------|
| INITIAL_EXTENT           | NUMBER      | Default initial extent size                                                                                                  |
| MIN_EXTENTS              | NUMBER      | Default minimum number of extents                                                                                            |
| MAX_EXTENTS              | NUMBER      | Default maximum number of extents                                                                                            |
| EXTENT_MANAGEMENT        | VARCHAR(5)  | Type of extent management<br>- LOCAL                                                                                         |
| SEGMENT_SPACE_MANAGEMENT | VARCHAR(4)  | Type of segment space management<br>- AUTO                                                                                   |
| BIGFILE                  | VARCHAR(3)  | Whether tablespace type is bigfile or not<br>- YES<br>- NO                                                                   |
| DEF_INMEMORY             | VARCHAR(8)  | Default value of inmemory is enabled or not for tablespace<br>- ENABLED<br>- DISABLED                                        |
| DEF_INMEMORY_PRIORITY    | VARCHAR(8)  | Default value of inmemory priority<br>- NONE : DEFAULT<br>- LOW<br>- MEDIUM<br>- HIGH<br>- CRITICAL<br>- NULL                |
| DEF_INMEMORY_DISTRIBUTE  | VARCHAR(15) | Default value of inmemory distribution method<br>- AUTO : DEFAULT<br>- BY ROWID RANGE<br>- BY PARTITION<br>- BY SUBPARTITION |

| Column Name              | Data Type   | Description                                                                                                                |
|--------------------------|-------------|----------------------------------------------------------------------------------------------------------------------------|
|                          |             | - NULL                                                                                                                     |
| DEF_INMEMORY_COMPRESSION | VARCHAR(14) | Default value of inmemory compression level<br>- NO MEMCOMPRESS : DEFAULT<br>- FOR QUERY LOW<br>- FOR QUERY HIGH<br>- NULL |
| DEF_INMEMORY_DUPLICATE   | VARCHAR(13) | Default value of inmemory duplicate method<br>- NO DUPLICATE : DEFAULT<br>- DUPLICATE<br>- DUPLICATE ALL<br>- NULL         |

- See also

[DBA\\_TABLESPACES](#)

### 3.2.338. USER\_TAB\_COMMENTS

Displays comments on all tables and views that the current user owns.

- Columns

| Column Name | Data Type     | Description           |
|-------------|---------------|-----------------------|
| TABLE_NAME  | VARCHAR(128)  | Name of the object    |
| TABLE_TYPE  | VARCHAR(9)    | Type of the object    |
| COMMENTS    | VARCHAR(4000) | Comment on the object |

### 3.2.339. USER\_TAB\_HISTGRM\_PENDING\_STATS

Displays histogram statistics of columns in pending state current user owns.

- See also

[ALL\\_TAB\\_HISTGRM\\_PENDING\\_STATS](#)

[DBA\\_TAB\\_HISTGRM\\_PENDING\\_STATS](#)

### 3.2.340. USER\_TAB\_IDENTITY\_COLS

Displays identity columns of all tables that the current user owns.

- See also

[ALL\\_TAB\\_IDENTITY\\_COLS](#)

[DBA\\_TAB\\_IDENTITY\\_COLS](#)

### 3.2.341. USER\_TAB\_PENDING\_STATS

Displays pending statistics for the tables that the current user owns.

- See also

[ALL\\_TAB\\_PENDING\\_STATS](#)

[DBA\\_TAB\\_PENDING\\_STATS](#)

### 3.2.342. USER\_TBL\_COLS

Displays columns of all tables and views that the current user owns.

- See also

[ALL\\_TBL\\_COLS](#)

[DBA\\_TBL\\_COLS](#)

### 3.2.343. USER\_TBL\_COLUMNS

Displays columns of all tables and views that the current user owns.

- See also

[ALL\\_TBL\\_COLUMNS](#)

[ALL\\_TBL\\_COLUMNS\\_WITH\\_DNO](#)

[DBA\\_TBL\\_COLUMNS](#)

### 3.2.344. USER\_TBL\_COL\_STATISTICS

Displays statistics and a histogram of columns by extracting them from USER\_TBL\_COLUMNS.

- See also

[ALL\\_TBL\\_COL\\_STATISTICS](#)

[DBA\\_TBL\\_COL\\_STATISTICS](#)

### 3.2.345. USER\_TBL\_HISTOGRAMS

Displays histogram information of tables that the current user owns.

- See also

[ALL\\_TBL\\_HISTOGRAMS](#)

[DBA\\_TBL\\_HISTOGRAMS](#)

### 3.2.346. USER\_TBL\_PARTITIONS

Displays partition information of all tables that the current user owns.

- See also

[ALL\\_TBL\\_PARTITIONS](#)

[DBA\\_TBL\\_PARTITIONS](#)

### 3.2.347. USER\_TBL\_PRIVS

Displays object privileges granted on all tables for which the current user is the object owner, grantor, or grantee.

- See also

[ALL\\_TBL\\_PRIVS](#)

[DBA\\_TBL\\_PRIVS](#)

### 3.2.348. USER\_TBL\_PRIVS\_MADE

Displays object privileges granted on tables for which the current user is the object owner.

- See also

[ALL\\_TBL\\_PRIVS\\_MADE](#)

### 3.2.349. USER\_TBL\_PRIVS\_REC'D

Displays object privileges granted on tables for which the current user is the object grantee.

- Columns

| Column Name | Data Type    | Description                                                                            |
|-------------|--------------|----------------------------------------------------------------------------------------|
| OWNER       | VARCHAR(128) | Name of the object owner                                                               |
| TABLE_NAME  | VARCHAR(128) | Name of the object                                                                     |
| GRANTOR     | VARCHAR(128) | Name of the user who granted the privilege                                             |
| PRIVILEGE   | VARCHAR(40)  | Name of the privilege to the table                                                     |
| GRANTABLE   | VARCHAR(3)   | Whether or not the privilege was granted with ADMIN<br>OPTION<br><br>- YES<br><br>- NO |

### 3.2.350. USER\_TRIGGERS

Displays all triggers that the current user owns.

- See also

[ALL\\_TRIGGERS](#)

[DBA\\_TRIGGERS](#)

### 3.2.351. USER\_TRIGGER\_COLS

Displays column information of all triggers that the current user owns and all triggers on tables that the current user owns. The columns are the same as those in ALL\_TRIGGER\_COLS.

- See also

[ALL\\_TRIGGER\\_COLS](#)

[DBA\\_TRIGGER\\_COLS](#)

### 3.2.352. USER\_TYPES

Displays all types that the current user owns.

- See also

[ALL\\_TYPES](#)

[DBA\\_TYPES](#)

### 3.2.353. USER\_TYPE\_ATTRS

Displays all the attributes of this type that the current user owns.

- See also

[ALL\\_TYPE\\_ATTRS](#)

[DBA\\_TYPE\\_ATTRS](#)

### 3.2.354. USER\_TYPE\_METHODS

Displays all the methods of this type that the current user owns.

- See also

[ALL\\_TYPE\\_METHODS](#)

[DBA\\_TYPE\\_METHODS](#)

### 3.2.355. USER\_UNUSED\_COL\_TABS

Displays all unused columns that the current user owns.

- See also

[ALL\\_UNUSED\\_COL\\_TABS](#)

[DBA\\_UNUSED\\_COL\\_TABS](#)

### 3.2.356. USER\_UPDATABLE\_COLUMNS

Displays all join view columns that the current user can update.

- See also

[ALL\\_UPDATABLE\\_COLUMNS](#)

[DBA\\_UPDATABLE\\_COLUMNS](#)

### 3.2.357. USER\_USERS

Displays a description on the current user.

- Columns

| Column Name        | Data Type    | Description                                                                                                                                                                                                                                      |
|--------------------|--------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| USERNAME           | VARCHAR(128) | Name of the user                                                                                                                                                                                                                                 |
| USER_ID            | NUMBER       | ID of the user                                                                                                                                                                                                                                   |
| ACCOUNT_STATUS     | VARCHAR(30)  | Account status<br><br>- LOCKED<br><br>- EXPIRED<br><br>- LOCKED(TIMED)<br><br>- EXPIRED(GRACE)<br><br>- EXPIRED & LOCKED(TIMED)<br><br>- EXPIRED(GRACE) & LOCKED(TIMED)<br><br>- EXPIRED(GRACE) & LOCKED<br><br>- EXPIRED & LOCKED<br><br>- OPEN |
| LOCK_DATE          | DATE         | Date that the account was locked                                                                                                                                                                                                                 |
| EXPIRY_DATE        | DATE         | Date that the account was expired                                                                                                                                                                                                                |
| DEFAULT_TABLESPACE | VARCHAR(128) | Default tablespace for data                                                                                                                                                                                                                      |
| CREATED            | DATE         | Date that the user was created                                                                                                                                                                                                                   |

- See also

[ALL\\_USERS](#)

[DBA\\_USERS](#)

### 3.2.358. USER\_VIEWS

Displays all views that the current user owns.

- See also

[ALL\\_VIEWS](#)

[ALL\\_VIEWS\\_INTERNAL](#)

[DBA\\_VIEWS](#)



# 제4장 Dynamic View

## 4.1. Overview

A **dynamic view**, which is also called Dynamic Performance View (DPV), belongs to the SYS user schema. Dynamic views are used by the DBA to monitor the internal operating status of Tiberio and thus users are required to have DBA authority when viewing most of the dynamic views.

Unlike static views providing structural information of the database, dynamic views provide status information of the database at a specific point in time. As the information of dynamic views is provided by information stored in memory and control files, it provides accurate information of the current status of the database.

As the information written to memory and the control file reflects changes over time, dynamic views are recommended for database monitoring and tuning. The dynamic views can be seen using the tbSQL tool, but are not updatable or deletable as they are managed by system.

## 4.2. List of Dynamic Views

This section provides the list of dynamic views of Tiberio in alphabetical order and describes the columns which corresponds to each dynamic view in the table.

The following is the list of summarized dynamic views:

| Dynamic View              | Description                                                                                                                        |
|---------------------------|------------------------------------------------------------------------------------------------------------------------------------|
| V\$ACCESS                 | Displays information for the objects that the session is accessing.                                                                |
| V\$ACTIVE_SESSION_HISTORY | Displays information on the active session history. To enable ACTIVE_SESSION_HISTORY, TOTAL_SHM_SIZE needs to be greater than 2GB. |
| V\$ALL_FILESTAT           | Displays statistics of data file and temp file I/O.                                                                                |
| V\$ALL_LOG_DEST           | Displays the storage path of all logs.                                                                                             |
| V\$ARCHIVED_LOG           | Displays archive log information.                                                                                                  |
| V\$ARCHIVE_DEST_FILES     | Displays archive log information for archive log files in LOG_ARCHIVE_DEST.                                                        |
| V\$AS_ALIAS               | All Aliases in all disk spaces mounted by the AS instance.                                                                         |
| V\$AS_DISK                | All disks discovered by the AS instance.                                                                                           |
| V\$AS_DISKSPACE           | All disk spaces discovered by the AS instance.                                                                                     |

| Dynamic View               | Description                                                        |
|----------------------------|--------------------------------------------------------------------|
| V\$AS_DISK_HISTORY         | All disks discovered by the AS instance.                           |
| V\$AS_EXTENT_MAP           | Extent map of each file in each disk space                         |
| V\$AS_FILE                 | All files in all disk spaces mounted by the AS instance.           |
| V\$AS_OPERATION            | Ongoing operations in each disk space                              |
| V\$AS_USER                 | All users in all disk spaces mounted by the AS instance.           |
| V\$AS_USERGROUP            | All usergroups in all disk spaces mounted by the AS instance.      |
| V\$AS_USERGROUP_MEMBER     | Each group members in all disk spaces mounted by the AS instance.  |
| V\$AUTO_COALESCE           | Displays statistics of an index auto coalesce operations           |
| V\$BACKUP                  | Displays the status of online backup.                              |
| V\$BACKUP_ARCHIVED_LOG     | Displays information of backup archive logs.                       |
| V\$BACKUP_LIST             | Displays information of backed up files in view.                   |
| V\$BACKUP_LIST_ARCHIVELOG  | Displays information of backed up archive logfiles.                |
| V\$BACKUP_LIST_CONTROLFILE | Displays information of backed up datafiles.                       |
| V\$BACKUP_LIST_DATAFILE    | Displays information of backed up datafiles.                       |
| V\$BACKUP_SET              | Displays information of backup sets.                               |
| V\$BACKUP_SET_TABLESPACE   | Displays information of backup tablespace                          |
| V\$BG_SESSION              | Displays information on each background session.                   |
| V\$BG_SESSTAT              | Displays statistics of each background session.                    |
| V\$BH                      | Displays information for all buffers.                              |
| V\$BLOCKER_SESSION         | Displays blocker sessions which are not blocked by other session.  |
| V\$BLOCK_CHANGE_TRACKING   | Displays information for block change tracking.                    |
| V\$BPARAMETERS             | Views the value of the parameter set in BTIP.                      |
| V\$BUFFER_CACHE_ADVICE     | Display the estimate hit ratios                                    |
| V\$BUFFER_POOL             | Displays information for buffer pools.                             |
| V\$BUFFER_POOL_STATISTICS  | Displays statistics of buffer pools.                               |
| V\$CHECKPOINT              | Displays information for checkpoints.                              |
| V\$CLUSTER_INTERCONNECTS   | Displays interconnects available for use in cluster communication. |
| V\$CONTEXT                 | Displays context attributes in the current session.                |
| V\$CONTROLFILE             | Displays information for control files.                            |
| V\$CORRUPT_XID             | Displays transactions whose recovery is suspended.                 |
| V\$DATABASE                | Displays database status information.                              |
| V\$DATAFILE                | Displays information for data files in the database.               |

| Dynamic View              | Description                                                                                  |
|---------------------------|----------------------------------------------------------------------------------------------|
| V\$DATAFILE_HEADER        | Displays information of ONLINE datafiles by extracting it from file headers.                 |
| V\$DBFILE                 | Displays datafile names in the database.                                                     |
| V\$DBLINK                 | Displays information for the open database links in the current session.                     |
| V\$DBLINK_STATUS          | Displays gateway and remote session information for the open database links in each session. |
| V\$DB_PIPES               | Displays DBMS_PIPE information.                                                              |
| V\$DML_STATS              | Displays dml stats that have not yet been flushed                                            |
| V\$ENCRYPTED_TABLESPACES  | Displays information on encrypted tablespaces.                                               |
| V\$ENCRYPTION_WALLET      | Displays information of the status of the wallet.                                            |
| V\$ENQUEUE_STAT           | Displays statistics of locks.                                                                |
| V\$EST_CHECKPOINT_TIME    | Displays information for estimated checkpoint time.                                          |
| V\$EVENT_NAME             | Displays event names.                                                                        |
| V\$FILESTAT               | Displays file read/write statistics.                                                         |
| V\$FLASHBACK_ARCHIVED_LOG | Displays flashback archive log information.                                                  |
| V\$FLASHBACK_LOG          | Displays information of flashback log groups.                                                |
| V\$FLASHBACK_MARKER       | Displays flashback markers for flashback database.                                           |
| V\$GLOBAL_TRANSACTION     | Displays information of the currently running global transactions.                           |
| V\$IM_COLUMN_LEVEL        | displays specified inmemory column options for table.                                        |
| V\$IM_SEGMENTS            | Displays information about all the in-memory segments in the database                        |
| V\$INSTANCE               | Displays instance statuses.                                                                  |
| V\$INSTANCEMETRIC         | Shows the load status of the current node in TAC.                                            |
| V\$INSTANCEMETRIC_HISTORY | Shows the load status of the current node in TAC.                                            |
| V\$INTERCONNECTION        | Displays informations of interconnections in cluster                                         |
| V\$INTERCONNECT_LATENCY   | Displays interconnect latency in cluster                                                     |
| V\$IPARAM_DESC            | Displays initialization parameters.                                                          |
| V\$LATCH                  | Displays spinlock information.                                                               |
| V\$LIBRARYCACHE           | Displays information of PP cache and DD cache.                                               |
| V\$LICENSE                | Displays license information.                                                                |
| V\$LISTENER_PORT          | Displays listener port information                                                           |
| V\$LOCK                   | Displays locks that are currently in use.                                                    |

| Dynamic View               | Description                                                                                                 |
|----------------------------|-------------------------------------------------------------------------------------------------------------|
| V\$LOCKED_OBJECT           | Displays session, transaction, object, lock info that are currently locked.                                 |
| V\$LOG                     | Displays information of log groups.                                                                         |
| V\$LOGFILE                 | Displays information of log member files.                                                                   |
| V\$LOG_HISTORY             | Displays log history information.                                                                           |
| V\$MEDIA_RECOVERY_PROGRESS | Displays the progress of media recovery.                                                                    |
| V\$MEMORY_TABLESPACE       | Displays information for memory tablespace data files in the database.                                      |
| V\$MEM_MGR                 | Memory manager information.                                                                                 |
| V\$METRIC                  | Displays information on the metric history.                                                                 |
| V\$METRICNAME              | Displays specifications of the metrics.                                                                     |
| V\$MODIFIED_PARAMETERS     | Displays values of modified initialization parameters.                                                      |
| V\$MYSTAT                  | Displays statistics of the current session.                                                                 |
| V\$NLS_PARAMETERS          | Displays nls parameters of current session.                                                                 |
| V\$OBJECT_USAGE            | Displays statistics about index usage gathered from the database for the indexes owned by the current user. |
| V\$OPEN_CURSOR             | Displays information for all open cursors for each session                                                  |
| V\$OSSTAT                  | Displays various OS statistics.                                                                             |
| V\$OSSTAT2                 | Displays various OS statistics.                                                                             |
| V\$OUTLINE                 | Displays outline hints in PPC                                                                               |
| V\$PARAMETERS              | Displays values of initialization parameters currently in effect for the session.                           |
| V\$PATCH                   | Displays list of patch files.                                                                               |
| V\$PE_PESPLAN              | Displays elapsed time information of PES in the format of SQL execution plan.                               |
| V\$PE_PESSTAT              | Displays statistics of each PES for parallel processing.                                                    |
| V\$PE_PROCESS              | Displays information of each parallel execution process currently allocated by QC session.                  |
| V\$PE_SESSION              | Displays sessions in parallel processing.                                                                   |
| V\$PE_SYSSTAT              | Displays statistics of parallel execution.                                                                  |
| V\$PE_TQSTAT               | Displays statistics of table queues for parallel processing.                                                |
| V\$PGASTAT                 | Displays statistics of PGA.                                                                                 |
| V\$PROCESS                 | Displays processes and threads used in RDBMS.                                                               |

| Dynamic View                      | Description                                                                                      |
|-----------------------------------|--------------------------------------------------------------------------------------------------|
| V\$PROC_CONN_INFO                 | Displays status of session connection for each foreground working processes.                     |
| V\$PROXY_SQLTEXT                  | Displays texts of SQL statements that are remotely executed through the database link.           |
| V\$PROXY_SQL_PARAM                | Displays bind parameters of SQL statements that are remotely executed through the database link. |
| V\$RECOVERY_FILE_STATUS           | Displays the status of a file that is being recovered.                                           |
| V\$RECOVERY_PROGRESS              | Displays recovery progress.                                                                      |
| V\$RECOVERY_STATUS                | Displays the status of media recovery progress.                                                  |
| V\$RECOVER_FILE                   | Displays files which require media recovery.                                                     |
| V\$RESERVED_WORDS                 | Displays a list of all SQL keywords.                                                             |
| V\$RESULT_CACHE_DEPENDENCY        | Displays the depends-on relationship between cached result and dependencies.                     |
| V\$RESULT_CACHE_OBJECTS           | Displays Result Cache Objects and their attributes.                                              |
| V\$RESULT_CACHE_OBJECT_STATISTICS | Displays Result Cache Object statistics.                                                         |
| V\$RESULT_CACHE_STATISTICS        | Displays various Result Cache settings and usage statistics.                                     |
| V\$ROLLSTAT                       | Displays statistics of Undo usage.                                                               |
| V\$ROWCACHE                       | Displays details of status information on DD cache.                                              |
| V\$RSBSTAT                        | Displays Stats of RSBs                                                                           |
| V\$RSRC_CONSUMER_GROUP            | Displays data related to currently active resource consumer groups                               |
| V\$RSRC_PLAN                      | Displays the names of all currently active resource plans                                        |
| V\$RSRC_SESSION_INFO              | Displays resource manager statistics per session                                                 |
| V\$RUNNING_JOBS                   | Displays information on running jobs.                                                            |
| V\$RUNNING_JOBS_WITH_NAME         | Displays information on running jobs with name.                                                  |
| V\$SEGMENT_STATISTICS             | !!! No documentation yet !!!                                                                     |
| V\$SERVER_EVENT_STATUS            | Displays information about the server event status                                               |
| V\$SERVER_EVENT_SUBTYPE           | Displays information about the server event subtype                                              |
| V\$SERVER_EVENT_TYPE              | Displays information about the server event type                                                 |
| V\$SESSION                        | Displays information on each session.                                                            |
| V\$SESSION_EVENT                  | Displays events that are performed in each session.                                              |
| V\$SESSION_IO                     | Displays input/output which occurred in each session.                                            |

| Dynamic View              | Description                                                                                                                                                  |
|---------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------|
| V\$SESSION_LATEST_STAT    | Displays the most recently updated stat information of each session.                                                                                         |
| V\$SESSION_LONGOPS        | Displays status of sessions taking longer time than specified in LONGOPS_THRESHOLD_SEC seconds.                                                              |
| V\$SESSION_WAIT           | Displays the waiting status of each session.                                                                                                                 |
| V\$SESSTAT                | Displays statistics of each session.                                                                                                                         |
| V\$SESS_TIME_MODEL        | Displays the session-level time statistics.                                                                                                                  |
| V\$SGA                    | Displays information on shared memory used in the database.                                                                                                  |
| V\$SGASTAT                | Displays information on the shared pool for a database, such as its purpose and how much of the shared pool is used.                                         |
| V\$SLOG                   | Displays slog from the current system log file. Up to SLOG_VIEW_SIZE bytes are stored in the view.                                                           |
| V\$SPINLOCK HOLDER        | Displays spinlock information.                                                                                                                               |
| V\$SPINUSE_STAT           | Displays statistics of Spinlock.                                                                                                                             |
| V\$SQL                    | Displays details of a SQL child cursor execution.                                                                                                            |
| V\$SQLAREA                | Displays details of SQL statement execution.                                                                                                                 |
| V\$SQLSTATS               | Displays information on SQL statement execution.                                                                                                             |
| V\$SQLTEXT                | Displays texts of SQL statements belonging to the cursor.                                                                                                    |
| V\$SQLTEXT2               | Displays texts of SQL statements belonging to the cursor.                                                                                                    |
| V\$SQLTEXT_WITH_NEWLINES  | Displays texts of SQL statements belonging to the cursor, without removing newline characters or blank characters.                                           |
| V\$SQLTEXT_WITH_NEWLINES2 | Displays texts of SQL statements belonging to the cursor, without removing newline characters or blank characters.                                           |
| V\$SQL_BIND_CAPTURE       | Displays the values of the captured bind variables from the current session                                                                                  |
| V\$SQL_BIND_CAPTURE_ALL   | Displays the values of the captured bind variables from all sessions                                                                                         |
| V\$SQL_EXPC_INFO          | Displays information about expression execution of a physical plan                                                                                           |
| V\$SQL_MONITOR            | Shows the details of SQL statements that are currently being monitored or those that have been monitored through the real time SQL monitoring function.      |
| V\$SQL_PLAN               | Displays information on physical plans to execute SQL statements.                                                                                            |
| V\$SQL_PLAN_MONITOR       | Displays plan level information about each SQL execution registered in V\$SQL_MONITOR. Each row corresponds to a single plan operation of one SQL execution. |
| V\$SQL_PLAN_STATISTICS    | Displays statistics of results of each physical plan job.                                                                                                    |
| V\$SQL_WORKAREA           | Display information about work areas used by SQL cursors.                                                                                                    |

| Dynamic View               | Description                                                                             |
|----------------------------|-----------------------------------------------------------------------------------------|
| V\$SQL_WORKAREA_HISTOGRAM  | Display the cumulative work area execution statistics                                   |
| V\$SSVR_CLIENT             | Displays information of valid connection to the storage server in stance.               |
| V\$SSVR_FLASHCACHE         | Displays information of flashcaches used by the storage server in stance.               |
| V\$SSVR_GRID_DISK          | Displays information of grid disks.                                                     |
| V\$SSVR_IOLOAD             | IO load of storage server                                                               |
| V\$SSVR_IORM_CATEGORY_PLAN | Category Plan List used by the storage server instance                                  |
| V\$SSVR_IORM_DB_PLAN       | DB Plan List used by the storage server instance                                        |
| V\$SSVR_IOSTAT             | IO stat per sec of storage server                                                       |
| V\$SSVR_MEMSTAT            | Memory stat of storage server                                                           |
| V\$SSVR_SLAB_STAT          | used buffer slab information of storage server                                          |
| V\$SSVR_STORAGE_DISK       | Displays information of storage disks used by the storage server instance.              |
| V\$STANDBY                 | Displays information on the database duplication in the RDBMS Standby Cluster.          |
| V\$STANDBY_DEST            | Displays information on the standby DB in the RDBMS Standby Cluster.                    |
| V\$STANDBY_LOG             | Displays information of log groups.                                                     |
| V\$STANDBY_LOGFILE         | Displays information of log member files.                                               |
| V\$STANDBY_STATUS          | Displays information on log duplication of the primary DB in the RDBMS Standby Cluster. |
| V\$STATNAME                | Displays the detailed information on the statistics event.                              |
| V\$SYSSTAT                 | Displays statistics of a system.                                                        |
| V\$SYSSTAT_VALUE           | Displays values for getting statistics of a system.                                     |
| V\$SYSTEM_EVENT            | Displays events performing in the system.                                               |
| V\$SYSTEM_PARAMETERS       | Displays values of initialization parameters currently in effect for the instance.      |
| V\$SYS_TIME_MODEL          | Displays the system-level time statistics.                                              |
| V\$TABLESPACE              | Displays information on tablespaces.                                                    |
| V\$TACSTAT                 | Displays statistics of a TAC.                                                           |
| V\$TEMPFILE                | Displays temporary files in the database.                                               |

| Dynamic View                        | Description                                                                         |
|-------------------------------------|-------------------------------------------------------------------------------------|
| <a href="#">V\$TEMPSEG_OP_USAGE</a> | Displays temporary segments usage in the server.                                    |
| <a href="#">V\$TEMPSEG_USAGE</a>    | Displays temporary segments in the server.                                          |
| <a href="#">V\$TEMPSTAT</a>         | Displays statistics of file I/O.                                                    |
| <a href="#">V\$THRINFO</a>          | Information on thr_info                                                             |
| <a href="#">V\$THR_ACT_INFO</a>     | Displays information on activities of the threads.                                  |
| <a href="#">V\$THR_EVENT</a>        | Displays events that are performed in each thread.                                  |
| <a href="#">V\$TIMER</a>            | Displays information on the system timer.                                           |
| <a href="#">V\$TIMEZONE_NAMES</a>   | Displays timezone names.                                                            |
| <a href="#">V\$TRANSACTION</a>      | Displays ongoing transactions.                                                      |
| <a href="#">V\$TSN_TIME</a>         | Displays the mapping table of TSN and time.                                         |
| <a href="#">V\$UNDOSTAT</a>         | Displays statistics of Undo storage.                                                |
| <a href="#">V\$UNDO_FREE_SPACE</a>  | Displays free space in block count for each UNDO tablespace.                        |
| <a href="#">V\$USGMT_TXENTRY</a>    | Displays undo segment transaction entry information.                                |
| <a href="#">V\$VERSION</a>          | Displays RDBMS version information.                                                 |
| <a href="#">V\$WAITER_SESSION</a>   | Displays information on the sessions that wait for a lock acquisition               |
| <a href="#">V\$WAITSTAT</a>         | Displays information on block wait events for each block type and class             |
| <a href="#">V\$WAITUSE_STAT</a>     | Displays information on block wait events                                           |
| <a href="#">V\$WAITUSE_THR_STAT</a> | Displays information on block wait events for each block type, class, and pin usage |

### 4.2.1. V\$ACCESS

Displays information for the objects that the session is accessing.

- Columns

| Column Name | Data Type    | Description         |
|-------------|--------------|---------------------|
| SID         | NUMBER       | Session ID          |
| OWNER       | VARCHAR(128) | Owner of the object |
| OBJECT      | VARCHAR(128) | Name of the object  |
| TYPE        | VARCHAR(23)  | Type of the object  |

## 4.2.2. V\$ACTIVE\_SESSION\_HISTORY

Displays information on the active session history. To enable ACTIVE\_SESSION\_HISTORY, TO TAL\_SHM\_SIZE needs to be greater than 2GB.

- Columns

| Column Name    | Data Type    | Description                                     |
|----------------|--------------|-------------------------------------------------|
| SAMPLE_ID      | NUMBER       | Sample ID                                       |
| THREAD#        | NUMBER       | Thread#                                         |
| SAMPLE_TIME    | DATE         | Sample Time                                     |
| SID            | NUMBER       | Session ID                                      |
| SESS_SERIAL_NO | NUMBER       | A serial number of a session                    |
| USER_NO        | NUMBER       | The user's ID                                   |
| USER_NAME      | VARCHAR(128) | The user's Name                                 |
| IPADDR         | VARCHAR(21)  | IP to which the user connects                   |
| WAIT_EVENT     | NUMBER       | Type of the wait event which the session waits  |
| ID1            | NUMBER       | First Parameter                                 |
| ID2            | NUMBER       | Second Parameter                                |
| WE_SEQ         | NUMBER       | Sequence# of the Wait Event                     |
| TIME_WAITED    | NUMBER       | The amount of time waited for this event (msec) |
| WAIT_OBJ_ID    | NUMBER       | ID of the object which the session waits        |
| WAIT_FILE_NO   | NUMBER       | Number of the file which the session waits      |
| WAIT_BLOCK_NO  | NUMBER       | Number of the block which the session waits     |
| WAIT_ROW_NO    | NUMBER       | Number of the row which the session waits       |
| USGMT_ID       | NUMBER       | USGMT number at the time of sampling            |
| SLOTNO         | NUMBER       | Slot number at the time of sampling             |
| WRAPNO         | NUMBER       | Wrap number at the time of sampling             |

| Column Name         | Data Type   | Description                                                                                                                                                                                                                                                                                                                          |
|---------------------|-------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| SQL_ID              | VARCHAR(13) | SQL identifier                                                                                                                                                                                                                                                                                                                       |
| SQL_CHILD_NUMBER    | NUMBER      | The number of this child physical plan                                                                                                                                                                                                                                                                                               |
| SUB_SQL_ID          | VARCHAR(13) | Identifier of the internal SQL referenced by the SQL ID                                                                                                                                                                                                                                                                              |
| SUB_CHILD_NUMBER    | NUMBER      | Child number of the internal SQL referenced by the SQL ID                                                                                                                                                                                                                                                                            |
| CURR_HASHVAL        | NUMBER      | SQL hash value                                                                                                                                                                                                                                                                                                                       |
| MODULE_NAME         | VARCHAR(64) | Name of a module specified by dbms_application_info.set_module                                                                                                                                                                                                                                                                       |
| ACTION_NAME         | VARCHAR(64) | Name of an action specified by dbms_application_info.set_module/action                                                                                                                                                                                                                                                               |
| CLIENT_INFO_NAME    | VARCHAR(64) | Name of client_info specified by dbms_application_info.set_client_info                                                                                                                                                                                                                                                               |
| PROG_NAME           | VARCHAR(30) | Client program name                                                                                                                                                                                                                                                                                                                  |
| SQL_EXEC_START      | DATE        | The time when the execution of the SQL referenced by SUB_SQL_ID started                                                                                                                                                                                                                                                              |
| SQL_EXEC_ID         | NUMBER      | Execution identifier of the SQL referenced by SUB_SQL_ID. It is necessary to identify multiple simultaneous SQL executions. To identify a single SQL execution, three columns; SUB_SQL_ID, SQL_EXEC_START, and SQL_EXEC_ID must be defined. The three column values are used to match with corresponding rows in the V\$SQL_MONITOR. |
| SQL_PLAN_LINE_ID    | NUMBER      | Identifier of the SQL plan operation referenced by SUB_SQL_ID                                                                                                                                                                                                                                                                        |
| PORT                | NUMBER      | Client port number                                                                                                                                                                                                                                                                                                                   |
| DELTA_TIME          | NUMBER      | Time interval (in seconds) since the last time this session was sampled or created.                                                                                                                                                                                                                                                  |
| DELTA_PHY_READ_BLKs | NUMBER      | The number of blocks read from disk over the last DELTA_TIME seconds                                                                                                                                                                                                                                                                 |

| Column Name         | Data Type | Description                                                |
|---------------------|-----------|------------------------------------------------------------|
| DELTA_LOG_READ_BLKs | NUMBER    | The number of blocks read over the last DELTA_TIME seconds |
| PGA_SIZE            | NUMBER    | The amount of PGA memory consumed by this session          |

### 4.2.3. V\$ALL\_FILESTAT

Displays statistics of data file and temp file I/O.

- Columns

| Column Name    | Data Type | Description                                   |
|----------------|-----------|-----------------------------------------------|
| FILE#          | NUMBER    | File number                                   |
| TEMPFILE       | CHAR(1)   | Whether this is tempfile                      |
| PHYRDS         | NUMBER    | The number of reading times                   |
| PHYWRTS        | NUMBER    | The number of writing times                   |
| PHYBLKRD       | NUMBER    | The number of read blocks                     |
| PHYBLKWRT      | NUMBER    | The number of written blocks                  |
| SINGLEBLKRDS   | NUMBER    | The number of times of reading a single block |
| READTIM        | NUMBER    | Reading time (msec)                           |
| WRITETIM       | NUMBER    | Writing time (msec)                           |
| SINGLEBLKRDTIM | NUMBER    | Time for which a single block was read (msec) |
| AVGIOTIM       | NUMBER    | Average I/O time (msec)                       |
| LSTIOTIM       | NUMBER    | Last I/O time (msec)                          |
| MINIOTIM       | NUMBER    | Shortest I/O time (msec)                      |
| MAXIORTM       | NUMBER    | Longest reading time (msec)                   |
| MAXIOWTM       | NUMBER    | Longest writing time (msec)                   |

## 4.2.4. V\$ALL\_LOG\_DEST

Displays the storage path of all logs.

- Columns

| Column Name | Data Type    | Description           |
|-------------|--------------|-----------------------|
| NAME        | VARCHAR(32)  | Name of log dest      |
| PATH        | VARCHAR(256) | Full-path of log dest |

## 4.2.5. V\$ARCHIVED\_LOG

Displays archive log information.

- Columns

| Column Name        | Data Type    | Description                                                                |
|--------------------|--------------|----------------------------------------------------------------------------|
| RECID              | NUMBER       | Archive log record ID                                                      |
| THREAD#            | NUMBER       | Log group number recorded by instances of a server                         |
| NAME               | VARCHAR(256) | Archive log file path and name                                             |
| SEQUENCE#          | NUMBER       | Sequence number of the log group                                           |
| FIRST_CHANGE#      | NUMBER       | TSN associated with the first archive log that belongs to the log group    |
| FIRST_TIME         | DATE         | Time when the first archive log that belongs to the log group was recorded |
| NEXT_CHANGE#       | NUMBER       | TSN associated with the next archive log that belongs to the log group     |
| NEXT_TIME          | DATE         | Time when the next archive log that belongs to the log group was recorded  |
| RESET LOGS_CHANGE# | NUMBER       | Resetlogs TSN of the database when the log was written                     |
| RESETLOGS_TIME     | DATE         | Resetlogs time of the database when the log was written                    |
| FILESIZE           | NUMBER       | Size of logfile in bytes                                                   |

| Column Name | Data Type  | Description                            |
|-------------|------------|----------------------------------------|
| BLOCKS      | NUMBER     | Number of blocks in logfile            |
| BLOCK_SIZE  | NUMBER     | Size of log block                      |
| STANDBY_LOG | VARCHAR(3) | Whether it was from standby log or not |

## 4.2.6. V\$ARCHIVE\_DEST\_FILES

Displays archive log information for archive log files in LOG\_ARCHIVE\_DEST.

- Columns

| Column Name        | Data Type    | Description                                                                |
|--------------------|--------------|----------------------------------------------------------------------------|
| NAME               | VARCHAR(256) | Archive log file path and name                                             |
| THREAD#            | NUMBER       | Log group number recorded by instances of a server                         |
| SEQUENCE#          | NUMBER       | Sequence number of the log group                                           |
| FIRST_CHANGE#      | NUMBER       | TSN associated with the first archive log that belongs to the log group    |
| FIRST_TIME         | DATE         | Time when the first archive log that belongs to the log group was recorded |
| NEXT_CHANGE#       | NUMBER       | TSN associated with the next archive log that belongs to the log group     |
| RESET LOGS_CHANGE# | NUMBER       | Resetlogs TSN of the database when the log was written                     |
| RESETLOGS_TIME     | DATE         | Resetlogs time of the database when the log was written                    |

## 4.2.7. V\$AS\_ALIAS

All Aliases in all disk spaces mounted by the AS instance.

- Columns

| Column Name | Data Type   | Description                  |
|-------------|-------------|------------------------------|
| NAME        | VARCHAR(48) | File Alias(File Name String) |

| Column Name       | Data Type | Description                                 |
|-------------------|-----------|---------------------------------------------|
| DISKSPACE_NUMBER  | NUMBER    | Disk space number that the alias belongs to |
| FILE_NUMBER       | NUMBER    | File number for the alias                   |
| FILE_INCARNATION  | NUMBER    | File incarnation number for the alias       |
| ALIAS_INCARNATION | NUMBER    | Alias incarnation number                    |

## 4.2.8. V\$AS\_DISK

All disks discovered by the AS instance.

- Columns

| Column Name      | Data Type   | Description                                                                                                               |
|------------------|-------------|---------------------------------------------------------------------------------------------------------------------------|
| DISKSPACE_NUMBER | NUMBER      | Disk space number that the disk belongs to                                                                                |
| DISK_NUMBER      | NUMBER      | Disk number in the disk space                                                                                             |
| STATE            | VARCHAR(16) | Disk state in the disk space<br>- ONLINE<br>- PREPARE_ADD<br>- PREPARE_DROP<br>- ADDING<br>- DROPPING<br>- SYNC<br>- FAIL |
| OS_MB            | NUMBER      | Disk size recognized by the OS                                                                                            |
| TOTAL_MB         | VARCHAR(10) | Disk size                                                                                                                 |
| FREE_MB          | VARCHAR(10) | Unused disk space                                                                                                         |
| NAME             | VARCHAR(48) | Disk name                                                                                                                 |
| FAILGROUP        | VARCHAR(48) | Name of the failure group that the disk belongs to                                                                        |

| Column Name | Data Type    | Description                                    |
|-------------|--------------|------------------------------------------------|
| PATH        | VARCHAR(256) | Absolute path of the disk                      |
| CREATE_DATE | DATE         | Date when the disk was added to the disk space |

## 4.2.9. V\$AS\_DISKSPACE

All disk spaces discovered by the AS instance.

- Columns

| Column Name          | Data Type   | Description                                                                                     |
|----------------------|-------------|-------------------------------------------------------------------------------------------------|
| DISKSPACE_NUMBER     | NUMBER      | Number assigned to the disk space                                                               |
| NAME                 | VARCHAR(48) | Name of the disk space                                                                          |
| SECTOR_SIZE          | NUMBER      | Physical block size (in bytes)                                                                  |
| BLOCK_SIZE           | NUMBER      | AS metadata block size (in bytes)                                                               |
| ALLOCATION_UNIT_SIZE | NUMBER      | Size of the allocation unit (in bytes)                                                          |
| STATE                | VARCHAR(16) | State of the disk space on the AS instance<br>- ASSIGNED<br>- MOUNTING<br>- CREATING<br>- MOUNT |
| TYPE                 | VARCHAR(8)  | Redundancy level of the disk space<br>- EXTERN<br>- NORMAL<br>- HIGH                            |
| TOTAL_MB             | NUMBER      | Total capacity of the disk space                                                                |
| FREE_MB              | NUMBER      | Unused capacity of the disk space                                                               |

| Column Name             | Data Type | Description                                               |
|-------------------------|-----------|-----------------------------------------------------------|
| REQUIRED_MIRROR_FREE_MB | NUMBER    | Required free capacity of the disk space for disk failure |
| USABLE_FILE_MB          | NUMBER    | Usable capacity of the disk space                         |

## 4.2.10. V\$AS\_DISK\_HISTORY

All disks discovered by the AS instance.

- Columns

| Column Name      | Data Type   | Description                                                                                               |
|------------------|-------------|-----------------------------------------------------------------------------------------------------------|
| DISKSPACE_NUMBER | NUMBER      | Disk space number that the disk belongs to                                                                |
| DISK_NUMBER      | NUMBER      | Disk number in the disk space                                                                             |
| OPERATION_TYPE   | VARCHAR(32) | Operation type in the disk space<br>- REPAIR_DROP<br>- DROP_ROLLBACK<br>- DROP<br>- ADD_ROLLBACK<br>- ADD |
| DATETIME         | DATE        | Date when the disk was add or drop to the disk space                                                      |

## 4.2.11. V\$AS\_EXTENT\_MAP

Extent map of each file in each disk space

- Columns

| Column Name      | Data Type | Description                                |
|------------------|-----------|--------------------------------------------|
| DISKSPACE_NUMBER | NUMBER    | Disk space number that the file belongs to |
| FILE_NUMBER      | NUMBER    | File number in the disk space              |

| Column Name   | Data Type | Description                                                         |
|---------------|-----------|---------------------------------------------------------------------|
| EXTENT_NUMBER | NUMBER    | Extent number in the file                                           |
| REDUN_NUMBER  | NUMBER    | Redundancy number of the extent<br>- 0 primary<br>- 1, 2 redun copy |
| EXTENT_SIZE   | NUMBER    | # of AUs in the extent                                              |
| DISK_NUMBER   | NUMBER    | Disk number in the disk space                                       |
| AU_NUMBER     | NUMBER    | AU number in the disk                                               |

## 4.2.12. V\$AS\_FILE

All files in all disk spaces mounted by the AS instance.

- Columns

| Column Name      | Data Type   | Description                                          |
|------------------|-------------|------------------------------------------------------|
| DISKSPACE_NUMBER | NUMBER      | Disk space number that the file belongs to           |
| FILE_NUMBER      | NUMBER      | File number in the disk space                        |
| INCARNATION      | NUMBER      | File incarnation number                              |
| BLOCK_SIZE       | NUMBER      | Block size of the file (in bytes)                    |
| BYTES            | NUMBER      | Number of bytes of the file                          |
| TYPE             | VARCHAR(64) | File type<br>- DATA<br>- CTRL<br>- CM                |
| REDUNDANCY       | VARCHAR(8)  | Redundancy level of the file<br>- EXTERN<br>- NORMAL |

| Column Name | Data Type  | Description                         |
|-------------|------------|-------------------------------------|
|             |            | - HIGH                              |
| STRIPED     | VARCHAR(8) | Striping type<br>- COARSE<br>- FINE |

### 4.2.13. V\$AS\_OPERATION

Ongoing operations in each disk space

- Columns

| Column Name      | Data Type   | Description                                             |
|------------------|-------------|---------------------------------------------------------|
| DISKSPACE_NUMBER | NUMBER      | Disk space number that the operation belongs to         |
| OPERATION        | VARCHAR(16) | Type of the operation                                   |
| STATE            | VARCHAR(8)  | State of the operation                                  |
| POWER            | NUMBER      | Power requested for the operation                       |
| JOB_DONE         | NUMBER      | # of extents that have been processed so far            |
| JOB_REMAIN       | NUMBER      | # of remained extents that have to be processed         |
| JOB_TOTAL        | NUMBER      | # of extents that have to be processed by the operation |
| EST_SPEED        | NUMBER      | extents processed divided by elapsed time               |
| EST_REMAIN_TIME  | VARCHAR(32) | job_total divided by ext_speed                          |

### 4.2.14. V\$AS\_USER

All users in all disk spaces mounted by the AS instance.

- Columns

| Column Name      | Data Type    | Description                                |
|------------------|--------------|--------------------------------------------|
| DISKSPACE_NUMBER | NUMBER       | Disk space number that the file belongs to |
| USER_NUMBER      | NUMBER       | User number in the disk space              |
| INCARNATION      | NUMBER       | User incarnation number                    |
| USER_NAME        | VARCHAR(128) | User name                                  |

## 4.2.15. V\$AS\_USERGROUP

All usergroups in all disk spaces mounted by the AS instance.

- Columns

| Column Name       | Data Type    | Description                                |
|-------------------|--------------|--------------------------------------------|
| DISKSPACE_NUMBER  | NUMBER       | Disk space number that the file belongs to |
| USERGROUP_NUMBER  | NUMBER       | Usergroup number in the disk space         |
| INCARNATION       | NUMBER       | Usergroup incarnation number               |
| OWNER_NUMBER      | NUMBER       | Owner number of the usergroup              |
| OWNER_INCARNATION | NUMBER       | Owner incarnation number                   |
| USERGROUP_NAME    | VARCHAR(128) | Usergroup name                             |

## 4.2.16. V\$AS\_USERGROUP\_MEMBER

Each group members in all disk spaces mounted by the AS instance.

- Columns

| Column Name      | Data Type | Description                                |
|------------------|-----------|--------------------------------------------|
| DISKSPACE_NUMBER | NUMBER    | Disk space number that the file belongs to |
| MEMBER_NUMBER    | NUMBER    | Member number in the usergroup             |

| Column Name               | Data Type | Description                  |
|---------------------------|-----------|------------------------------|
| MEMBER_INCARNA<br>TION    | NUMBER    | Member incarnation number    |
| USERGROUP_NUM<br>BER      | NUMBER    | Usergroup number             |
| USERGROUP_INCAR<br>NATION | NUMBER    | Usergroup incarnation number |

## 4.2.17. V\$AUTO\_COALESCE

Displays statistics of an index auto coalesce operations

- Columns

| Column Name  | Data Type | Description                                           |
|--------------|-----------|-------------------------------------------------------|
| BUCKET#      | NUMBER    | Hash bucket number of the coalescing request list     |
| SGMT_ID      | NUMBER    | Segment ID of the index to be automatically coalesced |
| COAL_REQ_CNT | NUMBER    | Total number of the coalescing requests               |
| COAL_CNT     | NUMBER    | Total number of the coalescing operations             |

## 4.2.18. V\$BACKUP

Displays the status of online backup.

- Columns

| Column Name | Data Type   | Description                                                                                                                     |
|-------------|-------------|---------------------------------------------------------------------------------------------------------------------------------|
| FILE#       | NUMBER      | File number                                                                                                                     |
| STATUS      | VARCHAR(10) | Status information<br><br>- Active: online backup is being performed<br><br>- Not Active : online backup is not being performed |
| CHANGE#     | NUMBER      | TSN for which backup is started if online backup is being performed                                                             |

| Column Name | Data Type | Description                                                          |
|-------------|-----------|----------------------------------------------------------------------|
| TIME        | DATE      | Time when the backup was started if online backup is being performed |

## 4.2.19. V\$BACKUP\_ARCHIVED\_LOG

Displays information of backup archive logs.

- Columns

| Column Name      | Data Type | Description                                        |
|------------------|-----------|----------------------------------------------------|
| SET_ID           | NUMBER    | Id of backup archivelogs                           |
| THREAD#          | NUMBER    | Log group number recorded by instances of a server |
| MIN_LOG_TIME     | DATE      | Minimum begin time of backup archivelogs           |
| MAX_LOG_TIME     | DATE      | Maximum end time of backup archivelogs             |
| MIN_LOG_TSN      | NUMBER    | Minimum begin TSN of backup archivelogs            |
| MAX_LOG_TSN      | NUMBER    | Maximum end TSN of backup archivelogs              |
| MIN_LOG_SEQUENCE | NUMBER    | Minimum sequence of backup archivelogs             |
| MAX_LOG_SEQUENCE | NUMBER    | Maximum sequence of backup archivelogs             |
| RESETLOG_TIME    | DATE      | Resetlog time of backup archivelogs                |
| RESETLOG_TSN     | NUMBER    | Resetlog TSN of backup archivelogs                 |

## 4.2.20. V\$BACKUP\_LIST

Displays information of backed up files in view.

- Columns

| Column Name | Data Type | Description      |
|-------------|-----------|------------------|
| SET_ID      | NUMBER    | Id of backup set |

| Column Name      | Data Type | Description              |
|------------------|-----------|--------------------------|
| FILE_TYPE        | VARCHAR   | Type of backup file      |
| FILE_NAME        | VARCHAR   | Full-path of backup file |
| BACKUP_FILE_NAME | VARCHAR   | Name of backup file      |

#### 4.2.21. V\$BACKUP\_LIST\_ARCHIVELOG

Displays information of backed up archive logfiles.

- Columns

| Column Name      | Data Type | Description              |
|------------------|-----------|--------------------------|
| SET_ID           | NUMBER    | Id of backup set         |
| FILE_NAME        | VARCHAR   | Full-path of backup file |
| BACKUP_FILE_NAME | VARCHAR   | Name of backup file      |

#### 4.2.22. V\$BACKUP\_LIST\_CONTROLFILE

Displays information of backed up datafiles.

- Columns

| Column Name      | Data Type | Description              |
|------------------|-----------|--------------------------|
| SET_ID           | NUMBER    | Id of backup set         |
| FILE_NAME        | VARCHAR   | Full-path of backup file |
| BACKUP_FILE_NAME | VARCHAR   | Name of backup file      |

#### 4.2.23. V\$BACKUP\_LIST\_DATAFILE

Displays information of backed up datafiles.

- Columns

| Column Name      | Data Type | Description              |
|------------------|-----------|--------------------------|
| SET_ID           | NUMBER    | Id of backup set         |
| FILE_NAME        | VARCHAR   | Full-path of backup file |
| BACKUP_FILE_NAME | VARCHAR   | Name of backup file      |

## 4.2.24. V\$BACKUP\_SET

Displays information of backup sets.

- Columns

| Column Name           | Data Type | Description                         |
|-----------------------|-----------|-------------------------------------|
| SET_ID                | NUMBER    | Id of backup set                    |
| STATUS                | VARCHAR   | Current status of backup set        |
| BACKUP_TYPE           | VARCHAR   | Type of backup                      |
| BASE_SET              | NUMBER    | Base set id of backup set           |
| START_TIME            | DATE      | Begin time of backup                |
| FINISH_TIME           | DATE      | End time of backup                  |
| ELAPSED_SECONDS       | NUMBER    | Elapsed time(sec) of backup set     |
| START_TSN             | NUMBER    | Start TSN of backup set             |
| FINISH_TSN            | NUMBER    | Finish TSN of backup set            |
| RESETLOGS_TSN         | NUMBER    | Resetlogs TSN of backup set         |
| BACKUP_OPTION         | VARCHAR   | Common options used for the backup  |
| LOGFILE_BACKUP_OPTION | VARCHAR   | Options used for the logfile backup |
| PARTIAL_BACKUP_OPTION | VARCHAR   | Options used for the partial backup |
| OUTPUT_SIZE_MB        | NUMBER    | Size(MB) of backup set              |
| STORAGE               | VARCHAR   | Location of backup set              |

| Column Name | Data Type | Description        |
|-------------|-----------|--------------------|
| BACKUP_PATH | VARCHAR   | Path of backup set |

## 4.2.25. V\$BACKUP\_SET\_TABLESPACE

Displays information of backup tablespace

- Columns

| Column Name | Data Type | Description      |
|-------------|-----------|------------------|
| SET_ID      | NUMBER    | Id of backup set |
| TS#         | NUMBER    | Tablespace ID    |

## 4.2.26. V\$BG\_SESSION

Displays information on each background session.

- Columns

| Column Name | Data Type   | Description                                                                                                                                                                                                                                                                                                                                                |
|-------------|-------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| SID         | NUMBER      | Session ID                                                                                                                                                                                                                                                                                                                                                 |
| STATUS      | VARCHAR(32) | Status of session<br><ul style="list-style-type: none"> <li>- READY: The session is connected</li> <li>- RUNNING: The session is running</li> <li>- TX_RECOVERING: The transaction is being recovered</li> <li>- SESS_CLEANUP: The session resources are being cleaned up</li> <li>- ASSIGNED: Connected to database, but the session is closed</li> </ul> |
| TYPE        | VARCHAR(15) | Session type<br><ul style="list-style-type: none"> <li>- WTHR: Working thread</li> <li>- CTHR: Control thread</li> <li>- LGWR: Log writing process</li> </ul>                                                                                                                                                                                              |

| Column Name      | Data Type    | Description                                                                                                                                                                                                                                                                                                                                                      |
|------------------|--------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                  |              | <ul style="list-style-type: none"> <li>- CKPT: Checkpoint process</li> <li>- LARC: Log archive</li> <li>- AGENT: Sequence process</li> <li>- MTHR: Monitoring process</li> <li>- DBWR: Datablock writing process</li> <li>- LNW: Log network writing process</li> </ul>                                                                                          |
| STATE            | VARCHAR(25)  | State of working thread <ul style="list-style-type: none"> <li>- INVALID: Not intialized</li> <li>- NEW: Being created</li> <li>- IDLE: Ready to run</li> <li>- RUNNING: Running</li> <li>- WAITING: Waiting for internal messsage</li> <li>- RECV_WAITING: Waiting for client message</li> <li>- STOP_BY_MTHR: Stopped by MTHR</li> <li>- DEAD: Dead</li> </ul> |
| WLOCK_WAIT       | VARCHAR(100) | Type of wlock for which the session waits                                                                                                                                                                                                                                                                                                                        |
| SEQ#             | NUMBER       | Sequence of wait event                                                                                                                                                                                                                                                                                                                                           |
| WAIT_EVENT       | NUMBER       | If the session is currently waiting, then the type of wait_event for which the session is waiting. If the session is not in a wait, then the value is -1.                                                                                                                                                                                                        |
| WAIT_TIME        | NUMBER       | If the session is currently waiting, then the value is the time spent in the wait (in 0.1 seconds). If the session is currently not in a wait, then the value is 0.                                                                                                                                                                                              |
| LAST_WAIT_EVENT# | NUMBER       | If the session is currently in a wait, then the type of wait_event for which the session is waiting. If the session is not in a wait, then the type of wait_event for which the session most recently waited.                                                                                                                                                    |
| LAST_WAIT_EVENT  | VARCHAR(100) | If the session is currently in a wait, then the type name of the wait_event for which the session is waiting. If the                                                                                                                                                                                                                                             |

| Column Name                | Data Type    | Description                                                                                                                                                                                                                        |
|----------------------------|--------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                            |              | session is not in a wait, then the type name of wait_event for which the session most recently waited.                                                                                                                             |
| LAST_WAIT_TIME             | NUMBER       | If the session is currently waiting, then the value is the time spent in the wait (in milliseconds). If the session is not in a wait, then the value is the amount of time waited in the most recent wait_event (in milliseconds). |
| LAST_WAIT_TIME_MICRO       | NUMBER       | If the session is currently waiting, then the value is the time spent in the wait (in microseconds). If the session is not in a wait, then the value is the amount of time waited in the most recent wait_event (in microseconds). |
| TIME_SINCE_LAST_WAIT       | NUMBER       | If the session is currently not in a wait, then the value is the time elapsed since the end of the last wait_event (in milliseconds). If the session is currently waiting, then the value is 0.                                    |
| TIME_SINCE_LAST_WAIT_MICRO | NUMBER       | If the session is currently not in a wait, then the value is the time elapsed since the end of the last wait_event (in microseconds). If the session is currently waiting, then the value is 0.                                    |
| PGA_USED_MEM               | NUMBER       | The amount of PGA memory occupied by the session                                                                                                                                                                                   |
| PID                        | NUMBER       | Index of a process to which a session belongs                                                                                                                                                                                      |
| WTHR_ID                    | NUMBER       | Index of a working thread process to which a session belongs                                                                                                                                                                       |
| OSUSER                     | VARCHAR(128) | OS account name of the session connected                                                                                                                                                                                           |

## 4.2.27. V\$BG\_SESSTAT

Displays statistics of each background session.

- Columns

| Column Name | Data Type | Description                          |
|-------------|-----------|--------------------------------------|
| TID         | NUMBER    | Session ID                           |
| STAT#       | NUMBER    | Number of event to obtain statistics |

| Column Name | Data Type   | Description                                          |
|-------------|-------------|------------------------------------------------------|
| NAME        | VARCHAR(60) | Name of event to obtain statistics                   |
| CLASS       | NUMBER      | Class of the event                                   |
| VALUE       | NUMBER      | Value of the event (Corresponding to the event name) |

## 4.2.28. V\$BH

Displays information for all buffers.

- Columns

| Column Name | Data Type  | Description                                                                 |
|-------------|------------|-----------------------------------------------------------------------------|
| FILE#       | NUMBER     | Absolute number of the file                                                 |
| BLOCK#      | NUMBER     | Block number                                                                |
| BLKTYPE     | NUMBER     | Type of the block                                                           |
| STATUS      | VARCHAR(7) | Status of the buffer<br>- READING<br>- CURRENT<br>- CR                      |
| PIN_MODE    | VARCHAR(3) | Pin mode                                                                    |
| TEMP        | VARCHAR(1) | Whether it is a buffer of the temporary file or not                         |
| DIRTY       | VARCHAR(1) | Whether it is a modified buffer or not                                      |
| TS#         | NUMBER     | ID of the tablespace to which the block corresponding to the buffer belongs |
| OBJD        | NUMBER     | ID of the segment to which the block corresponding to the buffer belongs    |

## 4.2.29. V\$BLOCKER\_SESSION

Displays blocker sessions which are not blocked by other session.

- Columns

| Column Name | Data Type   | Description                       |
|-------------|-------------|-----------------------------------|
| SID         | NUMBER      | Session ID of the blocker session |
| TYPE        | VARCHAR(25) | Type of the lock                  |
| ID1         | NUMBER      | First ID of the lock              |
| ID2         | NUMBER      | Second ID of the lock             |
| MODE_HELD   | NUMBER      | Mode in which the lock is held    |

### 4.2.30. V\$BLOCK\_CHANGE\_TRACKING

Displays information for block change tracking.

- Columns

| Column Name | Data Type    | Description                                       |
|-------------|--------------|---------------------------------------------------|
| STATUS      | VARCHAR(9)   | Status of block change tracking                   |
| FILENAME    | VARCHAR(256) | Path name of the block change tracking file       |
| FILESIZE    | NUMBER       | Size of the block change tracking file (in bytes) |

### 4.2.31. V\$BPARAMETERS

Views the value of the parameter set in BTIP.

- Columns

| Column Name | Data Type     | Description             |
|-------------|---------------|-------------------------|
| NAME        | VARCHAR(64)   | Name of the parameter.  |
| VALUE       | VARCHAR(1024) | Value of the parameter. |

## 4.2.32. V\$BUFFER\_CACHE\_ADVICE

Display the estimate hit ratios

- Columns

| Column Name       | Data Type | Description                                       |
|-------------------|-----------|---------------------------------------------------|
| BUFFER_CACHE_SIZE | NUMBER    | The buffer cache size (in Mbytes)                 |
| HIT_RATIO         | NUMBER    | The estimate hit ratio at given buffer cache size |

## 4.2.33. V\$BUFFER\_POOL

Displays information for buffer pools.

- Columns

| Column Name    | Data Type  | Description                                        |
|----------------|------------|----------------------------------------------------|
| ID             | NUMBER     | Buffer pool ID                                     |
| NAME           | VARCHAR(7) | Name of the buffer pool                            |
| BLOCK_SIZE     | NUMBER     | Size of the block                                  |
| BUFFERS        | NUMBER     | The number of the buffers                          |
| PINNED_BUFFERS | NUMBER     | The number of the pinned buffers                   |
| WSCNT          | NUMBER     | The number of the working sets of the buffers pool |

## 4.2.34. V\$BUFFER\_POOL\_STATISTICS

Displays statistics of buffer pools.

- Columns

| Column Name | Data Type | Description                                   |
|-------------|-----------|-----------------------------------------------|
| WS#         | NUMBER    | Working set number of the buffer pool         |
| TOTAL_CNT   | NUMBER    | The total number of blocks in the working set |
| MAIN_CNT    | NUMBER    | The number of main lists                      |

| Column Name | Data Type | Description                                                 |
|-------------|-----------|-------------------------------------------------------------|
| MAIN2_CNT   | NUMBER    | The number of the second main lists                         |
| AUX_CNT     | NUMBER    | The number of the auxiliary lists                           |
| HOT_CNT     | NUMBER    | The number of the most frequently used buffers              |
| CR_CNT      | NUMBER    | The number of the buffers read in the Consistent-Read mode  |
| EXL_CNT     | NUMBER    | The number of buffers being modified                        |
| SHR_CNT     | NUMBER    | The number of buffers being scanned                         |
| DIRTY_CNT   | NUMBER    | The number of the modified buffers                          |
| SUM_GOT     | NUMBER    | The total number of buffers gotten by the working set       |
| SUM_WRITE   | NUMBER    | The total number of the buffers modified by the working set |
| SUM_READ    | NUMBER    | The total number of the buffers scanned by the working set  |
| FREE_WAIT   | NUMBER    | The number of times it waited for a free buffer             |
| WRITE_WAIT  | NUMBER    | The number of times it waited to write data to disk         |
| BUSY_WAIT   | NUMBER    | The number of times it waited for the busy buffer           |
| FREE_SCAN   | NUMBER    | The number of times a free buffer is scanned                |
| MAIN2_SCAN  | NUMBER    | The number of times the second main list is scanned         |

## 4.2.35. V\$CHECKPOINT

Displays information for checkpoints.

- Columns

| Column Name | Data Type | Description                                                                  |
|-------------|-----------|------------------------------------------------------------------------------|
| ONGOING     | NUMBER    | Whether a checkpoint is activated or not<br>- 0: Checkpoint is not activated |

| Column Name    | Data Type    | Description                                                                                                                                               |
|----------------|--------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------|
|                |              | - 1: Checkpoint is activated                                                                                                                              |
| REQ_TYPE       | VARCHAR(100) | The reason the checkpoint is activated                                                                                                                    |
| WAITERS        | NUMBER       | The number of waiters waiting for checkpoint to complete                                                                                                  |
| MIN_WAIT_TSN   | NUMBER       | The smallest TSN among the target TSNs waiting for checkpoint to complete                                                                                 |
| TARGET_TSN     | NUMBER       | The target TSN of the checkpoint that is ongoing or has recently completed (A checkpoint gets completed when the target TSN is bigger than LOW_CACHE_TSN) |
| LOW_CACHE_TSN  | NUMBER       | TSN for which the current checkpoint is valid                                                                                                             |
| MAX_TIME       | NUMBER       | The longest amount of time needed to complete check point                                                                                                 |
| MAX_BLOCKS     | NUMBER       | The maximum number of blocks used by checkpoint                                                                                                           |
| CHECKPOINT_TSN | NUMBER       | TSN of the current checkpoint                                                                                                                             |

## 4.2.36. V\$CLUSTER\_INTERCONNECTS

Displays interconnects available for use in cluster communication.

- Columns

| Column Name | Data Type   | Description                                                                                                          |
|-------------|-------------|----------------------------------------------------------------------------------------------------------------------|
| NAME        | VARCHAR(15) | Name of the interconnect                                                                                             |
| IP_ADDRESS  | VARCHAR(16) | IP address of the interconnect                                                                                       |
| IS_PUBLIC   | VARCHAR(4)  | Type of interface:public or private. Currently, it supports only private so is always set to NO.                     |
| SOURCE      | VARCHAR(31) | Indicates where this interface was picked up from. Currently, it is always set to 'the local_cluster_addr parameter' |

## 4.2.37. V\$CONTEXT

Displays context attributes in the current session.

- Columns

| Column Name | Data Type      | Description                            |
|-------------|----------------|----------------------------------------|
| NAMESPACE   | VARCHAR(128)   | The namespace of the context attribute |
| ATTRIBUTE   | VARCHAR(128)   | The name of the attribute              |
| VALUE       | VARCHAR(65532) | The value of the attribute             |

## 4.2.38. V\$CONTROLFILE

Displays information for control files.

- Columns

| Column Name | Data Type    | Description                                                    |
|-------------|--------------|----------------------------------------------------------------|
| STATUS      | NUMBER       | Status information<br><br>- 0: Normal<br><br>- Negative: Error |
| NAME        | VARCHAR(256) | Path name for the control file                                 |
| BLKSIZE     | NUMBER       | The size of the control file block                             |
| BLOCKS      | NUMBER       | The number of blocks in the control file                       |

## 4.2.39. V\$CORRUPT\_XID

Displays transactions whose recovery is suspended.

- Columns

| Column Name | Data Type      | Description                     |
|-------------|----------------|---------------------------------|
| XID         | VARCHAR(65532) | ID of the suspended transaction |

## 4.2.40. V\$DATABASE

Displays database status information.

- Columns

| Column Name            | Data Type   | Description                                                                                                                              |
|------------------------|-------------|------------------------------------------------------------------------------------------------------------------------------------------|
| DBID                   | NUMBER      | Database identifier (Unique value specified when creating a database)                                                                    |
| NAME                   | VARCHAR(40) | Name of the database (Name specified when creating a database)                                                                           |
| DB_CREATE_DATE         | DATE        | Creation date of the database                                                                                                            |
| CF_CREATE_DATE         | DATE        | Creation date of the control file                                                                                                        |
| CURRENT_TSN            | NUMBER      | The current TSN of the database                                                                                                          |
| OPEN_MODE              | VARCHAR(20) | Open mode of the database<br><br>- MOUNTED<br><br>- RECOVERY<br><br>- READ WRITE<br><br>- READ ONLY<br><br>- READ ONLY WITH APPLY        |
| PROTECTION_MODE        | VARCHAR(12) | Protection mode currently in effect for the database<br><br>- UNPROTECTED<br><br>- PROTECTION<br><br>- AVAILABILITY<br><br>- PERFORMANCE |
| RESETLOG_TSN           | NUMBER      | TSN associated with the last resetlogs                                                                                                   |
| RESETLOG_DATE          | DATE        | The last resetlogs time                                                                                                                  |
| PREV_RESET<br>LOG_TSN  | NUMBER      | TSN associated with the previous resetlogs                                                                                               |
| PREV_RESET<br>LOG_DATE | DATE        | The previous resetlogs time                                                                                                              |

| Column Name                 | Data Type    | Description                                                                                                                                                  |
|-----------------------------|--------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------|
| DATABASE_ROLE               | VARCHAR(17)  | Current role of the database<br><br>- PRIMARY<br><br>- PHYSICAL STANDBY<br><br>- LOGICAL STANDBY (TODO)<br><br>- SNAPSHOT STANDBY<br><br>- CASCADING STANDBY |
| STANDBY_BECAME_PRIMARY_TSN  | NUMBER       | TSN at which the physical standby database became primary                                                                                                    |
| STANDBY_BECAME_PRIMARY_DATE | DATE         | Time at which the physical standby database became primary<br>TODO: NORMAL 아래에서 _vt_parameter 조회 가능하게 되면 DB_UNIQUE_NAME 컬럼을 _vt_parameter에서 가져와서 추가          |
| LOG_MODE                    | VARCHAR(12)  | Archive log mode<br><br>- ARCHIVELOG<br><br>- NOARCHIVELOG                                                                                                   |
| FORCE_LOGGING               | VARCHAR(16)  | Logging mode that is currently in force<br><br>- NO<br><br>- FORCED BY STANDBY                                                                               |
| CKPT_TSN                    | NUMBER       | TSN associated with the last checkpoint                                                                                                                      |
| CKPT_DATE                   | DATE         | The last checkpoint time                                                                                                                                     |
| CPU_NAME                    | VARCHAR(32)  | The CPU type name                                                                                                                                            |
| PLATFORM_NAME               | VARCHAR(32)  | The OS platform name                                                                                                                                         |
| CPU_MODEL                   | VARCHAR(256) | The CPU model name                                                                                                                                           |
| OS_UPTIME                   | VARCHAR(256) | The OS uptime info                                                                                                                                           |
| CM_OBSERVER                 | VARCHAR(8)   | Cluster Manager Observer info<br><br>- ENABLED<br><br>- DISABLED                                                                                             |

## 4.2.41. V\$DATAFILE

Displays information for data files in the database.

- Columns

| Column Name  | Data Type    | Description                                                                                                                                                                         |
|--------------|--------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| FILE#        | NUMBER       | Datafile number                                                                                                                                                                     |
| CREATE_TSN   | NUMBER       | TSN that created the datafile                                                                                                                                                       |
| CREATE_DATE  | DATE         | Datafile created time                                                                                                                                                               |
| TS#          | NUMBER       | Tablespace number to which the datafile belongs                                                                                                                                     |
| RFILE#       | NUMBER       | Relative datafile number in the tablespace                                                                                                                                          |
| STATUS       | VARCHAR(7)   | Status information of the datafile<br><br>- OFFLINE: Datafile needs to be deleted<br><br>- ONLINE: Datafile is in the normal state<br><br>- RECOVER: Datafile needs to be recovered |
| ENABLED      | CHAR(10)     | File accessibility<br><br>- DISABLED: Unable to read/write<br><br>- READ ONLY: Readable only<br><br>- READ WRITE: Readable and writable                                             |
| CKPT_TSN     | NUMBER       | TSN associated with the last checkpoint                                                                                                                                             |
| CKPT_DATE    | DATE         | The last checkpoint time                                                                                                                                                            |
| CREATE_BYTES | NUMBER       | Size of the datafile at the time when it is created                                                                                                                                 |
| NAME         | VARCHAR(256) | Path name of the datafile                                                                                                                                                           |

## 4.2.42. V\$DATAFILE\_HEADER

Displays information of ONLINE datafiles by extracting it from file headers.

- Columns

| Column Name     | Data Type    | Description                                          |
|-----------------|--------------|------------------------------------------------------|
| FILE#           | NUMBER       | Datafile number                                      |
| DBID            | NUMBER       | ID of the database to which the datafile belongs     |
| DBNAME          | VARCHAR(40)  | Name of the database to which the datafile belongs   |
| BLKSIZE         | NUMBER       | Size of the block                                    |
| RECOVER         | VARCHAR(3)   | Media recovery necessity                             |
| FUZZY           | VARCHAR(3)   | Whether or not the datafile has been normally closed |
| CREATE_TSN      | NUMBER       | TSN that created the datafile                        |
| CREATE_DATE     | DATE         | Datafile created time                                |
| TABLESPACE_NAME | VARCHAR(128) | Name of the tablespace to which the datafile belongs |
| TS#             | NUMBER       | Tablespace number to which the datafile belongs      |
| RFILE#          | NUMBER       | Tablespace relative datafile number                  |
| RESETLOGS_TSN   | NUMBER       | TSN associated with the last reset logs              |
| RESETLOGS_DATE  | DATE         | The last resetlogs time                              |
| CKPT_TSN        | NUMBER       | TSN associated with the last checkpoint              |
| CKPT_DATE       | DATE         | The last checkpoint time                             |
| CKPT_COUNT      | NUMBER       | Checkpoint count                                     |
| FILESIZE        | NUMBER       | Size of the datafile                                 |
| NAME            | VARCHAR(256) | Path name of the datafile                            |

#### 4.2.43. V\$DBFILE

Displays datafile names in the database.

- Columns

| Column Name | Data Type    | Description               |
|-------------|--------------|---------------------------|
| FILE#       | NUMBER       | Datafile number           |
| NAME        | VARCHAR(256) | Path name of the datafile |

## 4.2.44. V\$DBLINK

Displays information for the open database links in the current session.

- Columns

| Column Name           | Data Type    | Description                                                                                                 |
|-----------------------|--------------|-------------------------------------------------------------------------------------------------------------|
| DB_LINK               | VARCHAR(128) | Name of the database link                                                                                   |
| OWNER_ID              | NUMBER       | ID of the owner of the database link                                                                        |
| OPEN_CURSORS          | NUMBER       | The number of cursors opened by the current database link                                                   |
| IN_TRANSACTION        | VARCHAR(3)   | Whether it is a database link that is accessed from the currently running transaction or other transactions |
| HETEROGENEOUS         | VARCHAR(3)   | Whether or not the remote database is RDBMS                                                                 |
| COMMIT_POINT_STRENGTH | NUMBER       | Commit point strength of the remote database                                                                |

## 4.2.45. V\$DBLINK\_STATUS

Displays gateway and remote session information for the open database links in each session.

- Columns

| Column Name | Data Type    | Description                                                         |
|-------------|--------------|---------------------------------------------------------------------|
| SESS_ID     | NUMBER       | Session ID                                                          |
| SERIAL#     | NUMBER       | A serial number of a session (Like SID, used to recognize sessions) |
| DB_LINK     | VARCHAR(128) | Name of the database link                                           |
| OWNER_ID    | NUMBER       | ID of the owner of the database link                                |

| Column Name       | Data Type    | Description                                            |
|-------------------|--------------|--------------------------------------------------------|
| GATEWAY_PID       | NUMBER       | PID of the gateway (in heterogeneous link)             |
| GATEWAY_NAME      | VARCHAR(128) | Process name of the gateway (in heterogeneous link)    |
| GATEWAY_TID       | NUMBER       | TID (thread id) of the gateway (in heterogeneous link) |
| REMOTE_SESS_INFO  | VARCHAR(32)  | Remote session info                                    |
| NLS_CHARACTER_SET | VARCHAR(32)  | NLS_CHARACTER_SET of remote session                    |

## 4.2.46. V\$DB\_PIPES

Displays DBMS\_PIPE information.

- Columns

| Column Name | Data Type     | Description                   |
|-------------|---------------|-------------------------------|
| OWNERID     | NUMBER        | Owner id of a pipe.           |
| NAME        | VARCHAR(1000) | Pipe name.                    |
| TYPE        | VARCHAR(7)    | Pipe type(private or public). |
| PIPE_SIZE   | NUMBER        | Maximum size of pipe.         |

## 4.2.47. V\$DML\_STATS

Displays dml stats that have not yet been flushed

- Columns

| Column Name | Data Type | Description                 |
|-------------|-----------|-----------------------------|
| OBJ_ID      | NUMBER    | Object ID                   |
| INSERTS     | NUMBER    | The number of rows inserted |
| DELETES     | NUMBER    | The number of rows deleted  |
| UPDATES     | NUMBER    | The number of rows updated  |

## 4.2.48. V\$ENCRYPTED\_TABLESPACES

Displays information on encrypted tablespaces.

- Columns

| Column Name   | Data Type  | Description                                             |
|---------------|------------|---------------------------------------------------------|
| TS#           | NUMBER     | A serial number of tablespace                           |
| ENCRYPTIONALG | VARCHAR(7) | An encryption algorithm                                 |
| ENCRYPTEDTS   | VARCHAR(3) | Whether or not the tablespace was encrypted (YES or NO) |

## 4.2.49. V\$ENCRYPTION\_WALLET

Displays information of the status of the wallet.

- Columns

| Column Name   | Data Type     | Description                                                   |
|---------------|---------------|---------------------------------------------------------------|
| WRL_TYPE      | VARCHAR(20)   | Type of the wallet resource locator                           |
| WRL_PARAMETER | VARCHAR(4000) | Parameter of the wallet resource locator (absolute file name) |
| STATUS        | VARCHAR(10)   | Status of the wallet                                          |

## 4.2.50. V\$ENQUEUE\_STAT

Displays statistics of locks.

- Columns

| Column Name | Data Type   | Description          |
|-------------|-------------|----------------------|
| IDX         | NUMBER      | Lock number          |
| TYPE        | VARCHAR(25) | Lock type            |
| TOTAL_REQ#  | NUMBER      | Lock requested count |
| TOTAL_WAIT# | NUMBER      | Lock wait count      |

| Column Name   | Data Type | Description                 |
|---------------|-----------|-----------------------------|
| SUCC_REQ#     | NUMBER    | Lock success count          |
| FAILED_REQ#   | NUMBER    | Lock failure count          |
| CUM_WAIT_TIME | NUMBER    | Total Lock wait time (msec) |

## 4.2.51. V\$EST\_CHECKPOINT\_TIME

Displays information for estimated checkpoint time.

- Columns

| Column Name               | Data Type | Description                                |
|---------------------------|-----------|--------------------------------------------|
| DIRTY_BLOCKS              | NUMBER    | The number of dirty blocks in buffer cache |
| DISK_WRITE_TIME           | NUMBER    | The disk write time per block(ms)          |
| ESTIMATED_CHECKPOINT_TIME | NUMBER    | the estimated checkpoint time(s)           |

## 4.2.52. V\$EVENT\_NAME

Displays event names.

- Columns

| Column Name | Data Type    | Description                               |
|-------------|--------------|-------------------------------------------|
| EVENT#      | NUMBER       | Event number                              |
| NAME        | VARCHAR(100) | Event name                                |
| CLASS       | VARCHAR(40)  | Class of the event                        |
| DESC        | VARCHAR(100) | Description of the waitevent              |
| ID1_DESC    | VARCHAR(100) | Description of the first event parameter  |
| ID2_DESC    | VARCHAR(100) | Description of the second event parameter |

## 4.2.53. V\$FILESTAT

Displays file read/write statistics.

- Columns

| Column Name    | Data Type | Description                                                |
|----------------|-----------|------------------------------------------------------------|
| FILE#          | NUMBER    | File number                                                |
| PHYRDS         | NUMBER    | The number of read operations performed                    |
| PHYWRTS        | NUMBER    | The number of write operations performed                   |
| PHYBLKRD       | NUMBER    | The number of blocks read                                  |
| PHYBLKWRT      | NUMBER    | The number of blocks written                               |
| SINGLEBLKRDS   | NUMBER    | Single-block read count                                    |
| READTIM        | NUMBER    | Time spent on read operations (msec)                       |
| WRITETIM       | NUMBER    | Time spent on write operations (msec)                      |
| SINGLEBLKRDTIM | NUMBER    | Time spent to read a single-block (msec)                   |
| AVGIOTIM       | NUMBER    | Average input/output time (msec)                           |
| LSTIOTIM       | NUMBER    | Time spent on the last input/output operation (msec)       |
| MINIOTIM       | NUMBER    | The minimum time spent on an input/output operation (msec) |
| MAXIORTM       | NUMBER    | The maximum time spent on a read operation (msec)          |
| MAXIOWTM       | NUMBER    | The maximum time spent on a write operation (msec)         |

## 4.2.54. V\$FLASHBACK\_ARCHIVE\_LOG

Displays flashback archive log information.

- Columns

| Column Name | Data Type | Description           |
|-------------|-----------|-----------------------|
| RECID       | NUMBER    | Archive log record ID |

| Column Name        | Data Type    | Description                                                                |
|--------------------|--------------|----------------------------------------------------------------------------|
| THREAD#            | NUMBER       | Log group number recorded by instances of a server                         |
| NAME               | VARCHAR(256) | Archive log file path and name                                             |
| SEQUENCE#          | NUMBER       | Sequence number of the log group                                           |
| FIRST_CHANGE#      | NUMBER       | TSN associated with the first archive log that belongs to the log group    |
| FIRST_TIME         | DATE         | Time when the first archive log that belongs to the log group was recorded |
| NEXT_CHANGE#       | NUMBER       | TSN associated with the next archive log that belongs to the log group     |
| RESET LOGS_CHANGE# | NUMBER       | Resetlogs TSN of the database when the log was written                     |
| RESETLOGS_TIME     | DATE         | Resetlogs time of the database when the log was written                    |
| FILESIZE           | NUMBER       | Size of logfile in bytes                                                   |
| BLOCKS             | NUMBER       | Number of blocks in logfile                                                |
| BLOCK_SIZE         | NUMBER       | Size of log block                                                          |

## 4.2.55. V\$FLASHBACK\_LOG

Displays information of flashback log groups.

- Columns

| Column Name | Data Type | Description                                                 |
|-------------|-----------|-------------------------------------------------------------|
| THREAD#     | NUMBER    | Log group number recorded by instances of a server          |
| GROUP#      | NUMBER    | Log group number                                            |
| SEQUENCE#   | NUMBER    | Sequence number of the log group                            |
| BYTES       | NUMBER    | Size of the log group                                       |
| MEMBERS     | NUMBER    | The number of log member files that belong to the log group |

| Column Name   | Data Type  | Description                                                                                                                                                                                                              |
|---------------|------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ARCHIVED      | VARCHAR(3) | Whether it was archived or not                                                                                                                                                                                           |
| STATUS        | VARCHAR(8) | Status information of the log group<br>- ACTIVE: File archive needs to be done<br>- UNUSED: The log file has not been used<br>- CURRENT: The log file is currently in use<br>- INACTIVE: File archive has been completed |
| FIRST_CHANGE# | NUMBER     | TSN associated with the first log that belongs to the log group                                                                                                                                                          |
| FIRST_TIME    | DATE       | Time when the first log that belongs to the log group was recorded                                                                                                                                                       |

## 4.2.56. V\$FLASHBACK\_MARKER

Displays flashback markers for flashback database.

- Columns

| Column Name  | Data Type | Description                                         |
|--------------|-----------|-----------------------------------------------------|
| RECID        | NUMBER    | Flashback marker record ID                          |
| THREAD#      | NUMBER    | Instance number which recorded the flashback marker |
| SEQUENCE     | NUMBER    | Sequence number of the flashback log when marked    |
| BLOCK_NUMBER | NUMBER    | Last block number of the flashback log when marked  |
| CHANGE#      | NUMBER    | TSN when marked                                     |
| TIME         | DATE      | Time when marked                                    |

## 4.2.57. V\$GLOBAL\_TRANSACTION

Displays information of the currently running global transactions.

- Columns

| Column Name | Data Type      | Description                  |
|-------------|----------------|------------------------------|
| GLOBAL_ID   | VARCHAR(256)   | ID of the global transaction |
| SESS_ID     | NUMBER         | ID of the current session    |
| LOCAL_ID    | VARCHAR(65532) | ID of the local transaction  |

## 4.2.58. V\$IM\_COLUMN\_LEVEL

displays specified inmemory column opts for table.

- Columns

| Column Name          | Data Type     | Description                                                   |
|----------------------|---------------|---------------------------------------------------------------|
| OWNER                | VARCHAR2(128) | owner of the table                                            |
| OBJ_NUM              | NUMBER        | object number of the table                                    |
| TABLE_NAME           | VARCHAR2(128) | name of the table                                             |
| SEGMENT_COLUMN_ID    | NUMBER        | segment column number of the column                           |
| COLUMN_NAME          | VARCHAR2(128) | name of the column                                            |
| INMEMORY_COMPRESSION | VARCHAR(14)   | displays compression options or no inmemory option for column |
| CON_ID               | NUMBER        | ID of the container. currently only displays 0                |

## 4.2.59. V\$IM\_SEGMENTS

Displays information about all the in-memory segments in the database

- Columns

| Column Name    | Data Type    | Description                                     |
|----------------|--------------|-------------------------------------------------|
| OWNER          | VARCHAR(128) | User name of the segment owner                  |
| SEGMENT_NAME   | VARCHAR(128) | Name of the segment                             |
| PARTITION_NAME | VARCHAR(128) | Partition name (NULL for non-partitioned table) |

| Column Name         | Data Type    | Description                                                                                                                                                                         |
|---------------------|--------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| SEGMENT_TYPE        | VARCHAR(128) | Type of segment<br><br>- TABLE<br><br>- TABLE PARTITION<br><br>- TABLE SUBPARTITION                                                                                                 |
| TABLESPACE_NAME     | VARCHAR(30)  | Name of the tablespace containing the segment                                                                                                                                       |
| INMEMORY_SIZE       | NUMBER       | Size of the in-memory version of the segment, in bytes                                                                                                                              |
| BYTES               | NUMBER       | Total size of the on-disk segment, in bytes                                                                                                                                         |
| BYTES_NOT_POPULATED | NUMBER       | Size of the portion of the on-disk segment that is not populated in memory, in bytes                                                                                                |
| POPULATE_STATUS     | VARCHAR2(9)  | Status of segment population<br><br>- STARTED<br><br>- COMPLETED<br><br>- FAILED                                                                                                    |
| INMEMORY_PRIORITY   | VARCHAR2(8)  | Indicates the priority for IM column store population<br><br>- LOW<br><br>- MEDIUM<br><br>- HIGH<br><br>- CRITICAL<br><br>- NONE                                                    |
| INMEMORY_DISTRIBUTE | VARCHAR2(15) | Indicates how the IM column store is distributed in an Tibero Active Clusters(TAC) environment<br><br>- AUTO<br><br>- BY ROWID RANGE<br><br>- BY PARTITION<br><br>- BY SUBPARTITION |
| INMEMORY_DUPLICATE  | VARCHAR2(13) | Indicates the duplicate setting for the IM column store in an TAC environment                                                                                                       |

| Column Name              | Data Type    | Description                                                                                 |
|--------------------------|--------------|---------------------------------------------------------------------------------------------|
|                          |              | - NO DUPLICATE<br>- DUPLICATE<br>- DUPLICATE ALL                                            |
| INMEMORY_COM<br>PRESSION | VARCHAR2(14) | Compression level for the IM column store<br>- NO MEMCOMPRESS<br>- FOR QUERY [ LOW   HIGH ] |

## 4.2.60. V\$INSTANCE

Displays instance statuses.

- Columns

| Column Name     | Data Type   | Description                                                                           |
|-----------------|-------------|---------------------------------------------------------------------------------------|
| INSTANCE_NUMBER | NUMBER      | Instance number (Used by CM)                                                          |
| INSTANCE_NAME   | VARCHAR(40) | Name of the instance                                                                  |
| DB_NAME         | VARCHAR(40) | Name of the database                                                                  |
| HOST_NAME       | VARCHAR(63) | Name of the host                                                                      |
| PARALLEL        | VARCHAR(3)  | Whether clustering is activated or not                                                |
| THREAD#         | NUMBER      | Log group number recorded by instances of a server                                    |
| VERSION         | VARCHAR(8)  | Version of RDBMS                                                                      |
| STARTUP_TIME    | DATE        | Instance startup time                                                                 |
| STATUS          | VARCHAR(16) | Current status of the server<br>- NOMOUNT<br>- MOUNT<br>- NORMAL<br>- RAISING_SVRMODE |

| Column Name      | Data Type    | Description                                  |
|------------------|--------------|----------------------------------------------|
| SHUTDOWN_PENDING | VARCHAR(4)   | Whether server shutdown was requested or not |
| TIP_FILE         | VARCHAR(256) | Pathname of the TIP file                     |

## 4.2.61. V\$INSTANCEMETRIC

Shows the load status of the current node in TAC.

- Columns

| Column Name    | Data Type | Description                                                                                |
|----------------|-----------|--------------------------------------------------------------------------------------------|
| BEGIN_TIME     | DATE      | Begin time for load calculations                                                           |
| END_TIME       | DATE      | End time for load calculations                                                             |
| INTSIZE_CSEC   | NUMBER    | Interval length for load calculations (Unit: 0.01 seconds)                                 |
| ELAPSEDPERCALL | NUMBER    | Average processing time per user request within the interval                               |
| CPUPERCALL     | NUMBER    | Average CPU processing time per user request within the interval (currently not supported) |
| DBTIMEPERCALL  | NUMBER    | Average DB processing time per user request within the interval (currently not supported)  |
| CALLSPERSEC    | NUMBER    | Number of user requests per second within the interval                                     |
| DBTIMEPERSEC   | NUMBER    | The time consumed in DB per second within the interval (currently not supported)           |
| GOODNESS       | NUMBER    | Load level of the interval (number of sessions or average processing time used)            |
| DELTA          | NUMBER    | Load fluctuation predicated for additional sessions                                        |

## 4.2.62. V\$INSTANCEMETRIC\_HISTORY

Shows the load status of the current node in TAC.

- Columns

| Column Name    | Data Type | Description                                                                                |
|----------------|-----------|--------------------------------------------------------------------------------------------|
| BEGIN_TIME     | DATE      | Begin time for load calculations                                                           |
| END_TIME       | DATE      | End time for load calculations                                                             |
| INTSIZE_CSEC   | NUMBER    | Interval size for load calculations (Unit: 0.01 Seconds)                                   |
| ELAPSEDPERCALL | NUMBER    | Average processing time per user request within the interval                               |
| CPUPERCALL     | NUMBER    | Average CPU processing time per user request within the interval (currently not supported) |
| DBTIMEPERCALL  | NUMBER    | Average DB processing time per user request within the interval (currently not supported)  |
| CALLSPERSEC    | NUMBER    | Number of user requests per second within the interval                                     |
| DBTIMEPERSEC   | NUMBER    | The time consumed in DB per second within the interval (currently not supported)           |
| GOODNESS       | NUMBER    | Load level of the interval (number of sessions or average processing time used)            |
| DELTA          | NUMBER    | Load fluctuation predicated for additional sessions                                        |

## 4.2.63. V\$INTERCONNECTION

Displays informations of interconnections in cluster

- Columns

| Column Name   | Data Type | Description                |
|---------------|-----------|----------------------------|
| CONNECTION_ID | NUMBER    | Id of interconnection      |
| CONNECTED_NID | NUMBER    | Remote instance id         |
| MSG_TO_SEND   | NUMBER    | Number of messages to send |

## 4.2.64. V\$INTERCONNECT\_LATENCY

Displays interconnect latency in cluster

- Columns

| Column Name    | Data Type | Description              |
|----------------|-----------|--------------------------|
| REMOTE_INST_ID | NUMBER    | Remote instance id       |
| MIN_TIME       | NUMBER    | min time latency(us)     |
| MAX_TIME       | NUMBER    | max time latency(us)     |
| AVG_TIME       | NUMBER    | average time latency(us) |
| CUR_TIME       | NUMBER    | current time latency(us) |

## 4.2.65. V\$IPARAM\_DESC

Displays initialization parameters.

- Columns

| Column Name | Data Type      | Description                                 |
|-------------|----------------|---------------------------------------------|
| NAME        | VARCHAR(60)    | Name of the initialization parameter        |
| DESCRIPTION | VARCHAR(65532) | Description of the initialization parameter |

## 4.2.66. V\$LATCH

Displays spinlock information.

- Columns

| Column Name | Data Type   | Description           |
|-------------|-------------|-----------------------|
| LATCH#      | NUMBER      | Spinlock number       |
| LEVEL#      | NUMBER      | Level of the spinlock |
| CLASS#      | NUMBER      | Class of the spinlock |
| NAME        | VARCHAR(30) | Name of the spinlock  |

| Column Name      | Data Type | Description                                                                                    |
|------------------|-----------|------------------------------------------------------------------------------------------------|
| GETS             | NUMBER    | The number of times a spinlock was acquired                                                    |
| MISSES           | NUMBER    | The number of times the requester had to wait for a spinlock                                   |
| SPIN_GETS        | NUMBER    | The number of times the requester failed to acquire a spinlock but succeeded in gaining a spin |
| SLEEPS           | NUMBER    | The number of times the requester was put to sleep while waiting for a spinlock                |
| IMMEDIATE_GETS   | NUMBER    | The number of times a trylock was acquired                                                     |
| IMMEDIATE_MISSES | NUMBER    | The number of times the requester failed to acquire a trylock                                  |
| WAIT_TIME        | NUMBER    | The total sleep time of spinlock                                                               |
| MAX_WAIT         | NUMBER    | The maximum wait time of spinlock                                                              |
| SLEEPS_CNT       | NUMBER    | The number of actual sleep count while waiting for a spinlock                                  |

## 4.2.67. V\$LIBRARYCACHE

Displays information of PP cache and DD cache.

- Columns

| Column Name | Data Type   | Description                                                                                        |
|-------------|-------------|----------------------------------------------------------------------------------------------------|
| NAMESPACE   | VARCHAR(16) | Name of the table<br>- PP cache: set to "SQL AREA"<br>- DD cache: set to the name of each DD table |
| GETS        | NUMBER      | Cache total access count                                                                           |
| GETHITS     | NUMBER      | Cache hit count                                                                                    |
| GETHITRATIO | NUMBER      | Cache hit ratio (%)                                                                                |
| PINS        | NUMBER      | Number of times a PIN was requested for objects                                                    |

| Column Name | Data Type | Description                        |
|-------------|-----------|------------------------------------|
| PINHITS     | NUMBER    | Number of times objects was pinned |
| PINHITRATIO | NUMBER    | Ratio of PINHITS to PINS           |

## 4.2.68. V\$LICENSE

Displays license information.

- Columns

| Column Name     | Data Type    | Description                                                        |
|-----------------|--------------|--------------------------------------------------------------------|
| LICENSE_VERSION | VARCHAR(50)  | Version of the license                                             |
| LICENSEE        | VARCHAR(100) | User of the license                                                |
| LICENSE_TYPE    | VARCHAR(30)  | Type of the license<br>- DEMO<br>- REAL                            |
| PRODUCT_NAME    | VARCHAR(100) | Name of the product                                                |
| PRODUCT_VERSION | VARCHAR(50)  | Version of the product                                             |
| EDITION         | VARCHAR(30)  | Type of product<br>- BASE<br>- STANDARD<br>- ENTERPRISE            |
| ISSUE_DATE      | VARCHAR(10)  | Issue date of the license                                          |
| EXPIRE_DATE     | VARCHAR(10)  | Expiration date of the license (Only for the DEMO type of license) |
| LIMIT_USER      | VARCHAR(10)  | Limit for the number of users                                      |
| LIMIT_CPU       | VARCHAR(10)  | Limit for the number of CPUs                                       |
| HOST            | VARCHAR(100) | Host (IP or host name)                                             |

## 4.2.69. V\$LISTENER\_PORT

Displays listener port information

- Columns

| Column Name | Data Type   | Description      |
|-------------|-------------|------------------|
| IP          | VARCHAR(46) | LISTENER IP      |
| PORT        | NUMBER      | Port number      |
| IP_TYPE     | VARCHAR(20) | IP description   |
| PORT_TYPE   | VARCHAR(20) | Port description |

## 4.2.70. V\$LOCK

Displays locks that are currently in use.

- Columns

| Column Name | Data Type   | Description                                                                                                                                                                                                                              |
|-------------|-------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| TYPE        | VARCHAR(25) | Type of the Lock                                                                                                                                                                                                                         |
| ID1         | NUMBER      | The first ID of the Lock                                                                                                                                                                                                                 |
| ID2         | NUMBER      | The second ID of the Lock                                                                                                                                                                                                                |
| LMODE       | NUMBER      | Mode that the Lock owner owns.<br><br>- 0: None<br><br>- 1: Row-Shared(RS)<br><br>- 2: Row-Exclusive(RX)<br><br>- 3: Shared(S)<br><br>- 4: Shared-Row-Exclusive(SRX)<br><br>- 5: Exclusive(X)<br><br>- 6: Pin (Special mode for TX-wait) |
| REQUESTED   | NUMBER      | Mode that the Lock waiter requests                                                                                                                                                                                                       |
| SESS_ID     | NUMBER      | ID of the thread that owns the Lock(the same as SID)                                                                                                                                                                                     |

| Column Name | Data Type | Description                                             |
|-------------|-----------|---------------------------------------------------------|
| CTIME       | NUMBER    | The amount of elapsed time (in seconds) in current mode |

## 4.2.71. V\$LOCKED\_OBJECT

Displays session, transaction, object, lock info that are currently locked.

- Columns

| Column Name  | Data Type    | Description                                                                                                                                                                                                                              |
|--------------|--------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| XIDUSN       | NUMBER       | Undo segment number                                                                                                                                                                                                                      |
| XIDSLOT      | NUMBER       | Slot number of transaction entry                                                                                                                                                                                                         |
| XIDSQN       | NUMBER       | The number of times of reusing a transaction slot                                                                                                                                                                                        |
| OBJECT_ID    | NUMBER       | Object ID being locked                                                                                                                                                                                                                   |
| SESS_ID      | NUMBER       | Session ID                                                                                                                                                                                                                               |
| USERNAME     | VARCHAR(128) | The current user's name                                                                                                                                                                                                                  |
| OS_USER_NAME | VARCHAR(128) | OS account name of the session connected                                                                                                                                                                                                 |
| PROCESS      | NUMBER       | Client process id of the session connected                                                                                                                                                                                               |
| LOCKED_MODE  | NUMBER       | Mode that the Lock owner owns.<br><br>- 0: None<br><br>- 1: Row-Shared(RS)<br><br>- 2: Row-Exclusive(RX)<br><br>- 3: Shared(S)<br><br>- 4: Shared-Row-Exclusive(SRX)<br><br>- 5: Exclusive(X)<br><br>- 6: Pin (Special mode for TX-wait) |

## 4.2.72. V\$LOG

Displays information of log groups.

- Columns

| Column Name   | Data Type  | Description                                                                                                                                                                                                                              |
|---------------|------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| THREAD#       | NUMBER     | Log group number recorded by instances of a server                                                                                                                                                                                       |
| GROUP#        | NUMBER     | Log group number                                                                                                                                                                                                                         |
| SEQUENCE#     | NUMBER     | Sequence number of the log group                                                                                                                                                                                                         |
| BYTES         | NUMBER     | Size of the log group                                                                                                                                                                                                                    |
| MEMBERS       | NUMBER     | The number of log member files that belong to the log group                                                                                                                                                                              |
| ARCHIVED      | VARCHAR(3) | Whether it was archived or not                                                                                                                                                                                                           |
| STATUS        | VARCHAR(8) | Status information of the log group<br><br>- ACTIVE: File archive needs to be done<br><br>- UNUSED: The log file has not been used<br><br>- CURRENT: The log file is currently in use<br><br>- INACTIVE: File archive has been completed |
| FIRST_CHANGE# | NUMBER     | TSN associated with the first log that belongs to the log group                                                                                                                                                                          |
| FIRST_TIME    | DATE       | Time when the first log that belongs to the log group was recorded                                                                                                                                                                       |

## 4.2.73. V\$LOGFILE

Displays information of log member files.

- Columns

| Column Name | Data Type | Description                                            |
|-------------|-----------|--------------------------------------------------------|
| GROUP#      | NUMBER    | Log group number to which the log member file belongs. |

| Column Name | Data Type    | Description                                                                                                                      |
|-------------|--------------|----------------------------------------------------------------------------------------------------------------------------------|
| STATUS      | VARCHAR(7)   | Status information of the log member file (Whether or not the log member file was deleted. Currently, it is always set to null.) |
| TYPE        | CHAR(6)      | Always ONLINE                                                                                                                    |
| MEMBER      | VARCHAR(256) | Path name of the log member file                                                                                                 |

## 4.2.74. V\$LOG\_HISTORY

Displays log history information.

- Columns

| Column Name        | Data Type  | Description                                                        |
|--------------------|------------|--------------------------------------------------------------------|
| THREAD#            | NUMBER     | Log group number recorded by instances of a server                 |
| SEQUENCE#          | NUMBER     | Sequence number of the log group                                   |
| FIRST_CHANGE#      | NUMBER     | TSN associated with the first log that belongs to the log group    |
| FIRST_TIME         | DATE       | Time when the first log that belongs to the log group was recorded |
| NEXT_CHANGE#       | NUMBER     | TSN associated with the next log that belongs to the log group     |
| NEXT_TIME          | DATE       | Time when the next log that belongs to the log group was recorded  |
| RESET LOGS_CHANGE# | NUMBER     | Resetlogs TSN of the database when the log was written             |
| RESETLOGS_TIME     | DATE       | Resetlogs time of the database when the log was written            |
| FILESIZE           | NUMBER     | Size of the logfile at the time of log switch                      |
| STANDBY_LOG        | VARCHAR(3) | Whether it was from standby log or not                             |

## 4.2.75. V\$MEDIA\_RECOVERY\_PROGRESS

Displays the progress of media recovery.

- Columns

| Column Name        | Data Type   | Description                                                                           |
|--------------------|-------------|---------------------------------------------------------------------------------------|
| THREAD#            | NUMBER      | Log group number recorded by instances of a server                                    |
| START_TIME         | VARCHAR(19) | The time when media recovery began                                                    |
| CURRENT_TIME       | VARCHAR(19) | Current time                                                                          |
| ESTIMATE_TIME      | VARCHAR(19) | Estimated finish time of the media recovery                                           |
| BEGIN_SEQUENCE#    | NUMBER      | Sequence number of the log group where media recovery started                         |
| RECOVERY_SEQUENCE# | NUMBER      | Sequence number of the log group that is currently being replayed                     |
| END_SEQUENCE#      | NUMBER      | Sequence number of the log group where media recovery would end for complete recovery |

## 4.2.76. V\$MEMORY\_TABLESPACE

Displays information for memory tablespace data files in the database.

- Columns

| Column Name | Data Type  | Description                                                                   |
|-------------|------------|-------------------------------------------------------------------------------|
| FILE#       | NUMBER     | Datafile number                                                               |
| CREATE_TSN  | NUMBER     | TSN that created the datafile                                                 |
| CREATE_DATE | DATE       | Datafile created time                                                         |
| TS#         | NUMBER     | Tablespace number to which the datafile belongs                               |
| RFILE#      | NUMBER     | Relative datafile number in the tablespace                                    |
| STATUS      | VARCHAR(7) | Status information of the datafile<br>- OFFLINE: Datafile needs to be deleted |

| Column Name  | Data Type    | Description                                                                                                                 |
|--------------|--------------|-----------------------------------------------------------------------------------------------------------------------------|
|              |              | - ONLINE: Datafile is in the normal state<br>- RECOVER: Datafile needs to be recovered                                      |
| ENABLED      | CHAR(10)     | File accessibility<br>- DISABLED: Unable to read/write<br>- READ ONLY: Readable only<br>- READ WRITE: Readable and writable |
| CKPT_TSN     | NUMBER       | TSN associated with the last checkpoint                                                                                     |
| CKPT_DATE    | DATE         | The last checkpoint time                                                                                                    |
| CREATE_BYTES | NUMBER       | Size of the datafile at the time when it is created                                                                         |
| NAME         | VARCHAR(256) | Path name of the datafile                                                                                                   |

## 4.2.77. V\$MEM\_MGR

Memory manager information.

- Columns

| Column Name | Data Type   | Description                          |
|-------------|-------------|--------------------------------------|
| PID         | VARCHAR(16) | PID                                  |
| NAME        | VARCHAR(10) | Programe                             |
| TOTAL       | NUMBER      | Total bound size of the proc_mem_mgr |
| USED        | NUMBER      | Total used size of the proc_mem_mgr  |

## 4.2.78. V\$METRIC

Displays information on the metric history.

- Columns

| Column Name | Data Type   | Description                     |
|-------------|-------------|---------------------------------|
| SAMPLE_ID   | NUMBER      | Sample ID                       |
| THREAD#     | NUMBER      | Thread#                         |
| BEGIN_TIME  | DATE        | Begin Time of the Sample        |
| END_TIME    | DATE        | End Time of the Sample          |
| METRIC_ID   | NUMBER      | Metric ID                       |
| METRIC_NAME | VARCHAR(64) | Metric Name                     |
| METRIC_UNIT | VARCHAR(64) | Descriptions of the Metric Unit |
| VALUE       | NUMBER      | Value of the Metric             |
| GROUP_ID    | NUMBER      | Group ID                        |
| GROUP_NAME  | VARCHAR(64) | Group Name                      |
| INTERVAL    | NUMBER      | Interval of the Collection(Sec) |

## 4.2.79. V\$METRICNAME

Displays specifications of the metrics.

- Columns

| Column Name | Data Type   | Description                        |
|-------------|-------------|------------------------------------|
| METRIC_ID   | NUMBER      | Metric ID                          |
| METRIC_NAME | VARCHAR(64) | Metric Name                        |
| METRIC_UNIT | VARCHAR(64) | Descriptions of the Metric Unit    |
| GROUP_ID    | NUMBER      | Group ID                           |
| GROUP_NAME  | VARCHAR(64) | Group Name                         |
| INTERVAL    | NUMBER      | Interval of Metric Collection(Sec) |

## 4.2.80. V\$MODIFIED\_PARAMETERS

Displays values of modified initialization parameters.

- Columns

| Column Name | Data Type     | Description                                                               |
|-------------|---------------|---------------------------------------------------------------------------|
| NAME        | VARCHAR(64)   | Name of an initialization parameter                                       |
| VALUE       | VARCHAR(1024) | Value of an initialization parameter currently in effect for the instance |
| DFLT_VALUE  | VARCHAR(1024) | Default Value of an initialization parameter                              |
| STATUS      | VARCHAR(16)   | Current Status of an initialization parameter                             |

## 4.2.81. V\$MYSTAT

Displays statistics of the current session.

- Columns

| Column Name | Data Type   | Description                                 |
|-------------|-------------|---------------------------------------------|
| SID         | NUMBER      | ID of the session                           |
| STAT#       | NUMBER      | Event number used to obtain statistics      |
| NAME        | VARCHAR(60) | Name of the event used to obtain statistics |
| CLASS       | NUMBER      | Class of the event                          |
| VALUE       | NUMBER      | The number of times that the event was used |

## 4.2.82. V\$NLS\_PARAMETERS

Displays nls parameters of current session.

- Columns

| Column Name | Data Type   | Description              |
|-------------|-------------|--------------------------|
| PARAMETER   | VARCHAR(64) | Name of an nls parameter |

| Column Name | Data Type   | Description               |
|-------------|-------------|---------------------------|
| VALUE       | VARCHAR(64) | Value of an nls parameter |

### 4.2.83. V\$OBJECT\_USAGE

Displays statistics about index usage gathered from the database for the indexes owned by the current user.

- Columns

| Column Name      | Data Type    | Description                                                 |
|------------------|--------------|-------------------------------------------------------------|
| OWNER_NAME       | VARCHAR(128) | Name of user who has the object                             |
| INDEX_NAME       | VARCHAR(128) | The name of the index that is monitored                     |
| TABLE_NAME       | VARCHAR(128) | The name of the base table                                  |
| MONITORING       | VARCHAR(1)   | Y or N                                                      |
| USED             | VARCHAR(1)   | Y or N                                                      |
| START_MONITORING | DATE         | Start monitoring time in sys._object_usage.start_monitoring |
| END_MONITORING   | DATE         | End monitoring time in sys._object_usage.end_monitoring     |

### 4.2.84. V\$OPEN\_CURSOR

Displays information for all open cursors for each session

- Columns

| Column Name | Data Type    | Description                     |
|-------------|--------------|---------------------------------|
| SID         | NUMBER       | Session ID                      |
| CURSOR_ID   | NUMBER       | cursor ID                       |
| USER_NAME   | VARCHAR(128) | Name of the current user        |
| HASH_VALUE  | NUMBER       | Hash value of the SQL statement |

| Column Name     | Data Type   | Description                                |
|-----------------|-------------|--------------------------------------------|
| PLAN_HASH_VALUE | NUMBER      | Hash value of the physical plan            |
| SQL_ID          | VARCHAR(13) | SQL identifier                             |
| SQL_TEXT        | VARCHAR(64) | SQL text of the open cursor                |
| EXEC_TIME       | NUMBER      | Total working time performed on the cursor |

## 4.2.85. V\$OSSTAT

Displays various OS statistics.

- Columns

| Column Name | Data Type   | Description             |
|-------------|-------------|-------------------------|
| PID         | NUMBER      | Pid of process          |
| NAME        | VARCHAR(64) | Name of process         |
| CPUTIME     | NUMBER      | CPU user time           |
| CPUUSAGE    | NUMBER      | CPU usage in percentage |

## 4.2.86. V\$OSSTAT2

Displays various OS statistics.

- Columns

| Column Name | Data Type    | Description                                        |
|-------------|--------------|----------------------------------------------------|
| ID          | NUMBER       | Id of OS statistic                                 |
| STAT_NAME   | VARCHAR(64)  | Name of OS statistic                               |
| COMMENTS    | VARCHAR(512) | OS statistic comments                              |
| VALUE       | NUMBER       | OS statistic value                                 |
| CUMULATIVE  | VARCHAR(3)   | Whether the statistic is cumulative('YES' or 'NO') |

## 4.2.87. V\$OUTLINE

Displays outline hints in PPC

- Columns

| Column Name     | Data Type    | Description                                                 |
|-----------------|--------------|-------------------------------------------------------------|
| SQL_ID          | VARCHAR(13)  | SQL identifier of the parent statement in the library cache |
| PLAN_HASH_VALUE | NUMBER       | The hash value of the physical plan                         |
| CHILD_NUMBER    | NUMBER       | The number of this child physical plan                      |
| NO              | NUMBER       | The number of relative hint order                           |
| HINT            | VARCHAR(100) | Outline hint in string                                      |

## 4.2.88. V\$PARAMETERS

Displays values of initialization parameters currently in effect for the session.

- Columns

| Column Name | Data Type     | Description                                  |
|-------------|---------------|----------------------------------------------|
| NAME        | VARCHAR(64)   | Name of an initialization parameter          |
| VALUE       | VARCHAR(1024) | Value of an initialization parameter         |
| DFLT_VALUE  | VARCHAR(1024) | Default Value of an initialization parameter |

## 4.2.89. V\$PATCH

Displays list of patch files.

- Columns

| Column Name | Data Type   | Description        |
|-------------|-------------|--------------------|
| NAME        | VARCHAR(32) | Name of patch file |

## 4.2.90. V\$PE\_PESPLAN

Displays elapsed time information of PES in the format of SQL execution plan.

- Columns

| Column Name    | Data Type      | Description                                |
|----------------|----------------|--------------------------------------------|
| Execution_Plan | VARCHAR(65532) | Execution plan and embedded PES statistics |

## 4.2.91. V\$PE\_PESSTAT

Displays statistics of each PES for parallel processing.

- Columns

| Column Name  | Data Type   | Description                                 |
|--------------|-------------|---------------------------------------------|
| SQL_ID       | VARCHAR(13) | SQL identifier                              |
| PEM_ID       | NUMBER      | PEM node ID                                 |
| PROD_PPN_ID  | NUMBER      | PPN ID associated with the producer PES     |
| PROD_PES_ID  | NUMBER      | Producer PES ID                             |
| CONS_PPN_ID  | NUMBER      | PPN ID associated with the consumer PES     |
| CONS_PES_ID  | NUMBER      | Consumer PES ID                             |
| ELAPSED_TIME | NUMBER      | Total elapsed time                          |
| SAMPLE_TIME  | NUMBER      | Time taken for sampling with the range mode |

## 4.2.92. V\$PE\_PROCESS

Displays information of each parallel execution process currently allocated by QC session.

- Columns

| Column Name | Data Type | Description                   |
|-------------|-----------|-------------------------------|
| PEP_ID      | NUMBER    | PE process ID                 |
| OS_PID      | NUMBER    | PE process ID generated by OS |

| Column Name | Data Type   | Description                                                                                                |
|-------------|-------------|------------------------------------------------------------------------------------------------------------|
| STATUS      | VARCHAR(16) | The status of PE process                                                                                   |
| CTHR_TID    | NUMBER      | The control thread ID of PE process                                                                        |
| QCSID       | NUMBER      | ID of a session working as a parallel coordinator                                                          |
| QCSERIAL#   | NUMBER      | A serial number of a session working as a parallel coordinator (Like QCSID, used to recognize the session) |
| SQL_CNT     | NUMBER      | The count of processed SQLs plus the current sql after this process has been assigned                      |
| PENDING_MSG | VARCHAR(5)  | A flag which indicates whether this process has a pending message from QC                                  |

### 4.2.93. V\$PE\_SESSION

Displays sessions in parallel processing.

- Columns

| Column Name | Data Type | Description                                                                                                |
|-------------|-----------|------------------------------------------------------------------------------------------------------------|
| SID         | NUMBER    | Session ID                                                                                                 |
| SERIAL#     | NUMBER    | A serial number of a session (Like SID, used to recognize the session)                                     |
| PEP_ID      | NUMBER    | ID of parallel execution process which this session belongs to                                             |
| QCSID       | NUMBER    | ID of a session working as a parallel coordinator                                                          |
| QCSERIAL#   | NUMBER    | A serial number of a session working as a parallel coordinator (Like QCSID, used to recognize the session) |
| DEGREE      | NUMBER    | Parallel processing degree specified by server                                                             |
| REQ_DEGREE  | NUMBER    | Parallel processing degree requested by user                                                               |

## 4.2.94. V\$PE\_SYSSTAT

Displays statistics of parallel execution.

- Columns

| Column Name | Data Type   | Description                                                                                                                                                                                                                                                                                                                                                                                                               |
|-------------|-------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| NAME        | VARCHAR(64) | Name of a PE statistic<br><br>- PEP allocation failure count (current) : The number of failure to allocate PEP for current parallel coordinators. This value is incremented at most once for each session.<br><br>- PEP allocation failure count (accumulated) : The number of failure to allocate PEP for all parallel coordinators including nonexistent ones. This value is incremented at most once for each session. |
| VALUE       | NUMBER      | Value of the PE statistic                                                                                                                                                                                                                                                                                                                                                                                                 |

## 4.2.95. V\$PE\_TQSTAT

Displays statistics of table queues for parallel processing.

- Columns

| Column Name | Data Type   | Description                                                                                                                                                                                                                   |
|-------------|-------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| PEM_ID      | NUMBER      | PEM node ID                                                                                                                                                                                                                   |
| TQ_ID       | NUMBER      | Table queue ID                                                                                                                                                                                                                |
| SERVER_TYPE | VARCHAR(10) | The role of server thread<br><br>- PRODUCER: Transfers a result row to another thread<br><br>- CONSUMER: Receives a result row from another thread<br><br>- RANGER: Determines the range in which a result row is transferred |
| NUM_ROWS    | NUMBER      | The number of processed rows                                                                                                                                                                                                  |
| BYTES       | NUMBER      | The byte number of processed data                                                                                                                                                                                             |

| Column Name | Data Type   | Description                    |
|-------------|-------------|--------------------------------|
| WAITS       | NUMBER      | The number of times of waiting |
| THREAD      | VARCHAR(30) | Thread name                    |

## 4.2.96. V\$PGASTAT

Displays statistics of PGA.

- Columns

| Column Name | Data Type   | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|-------------|-------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| NAME        | VARCHAR(64) | <p>Name of a PGA statistic</p> <ul style="list-style-type: none"> <li>- Ex memory soft limit: The maximum amount of PGA memory that can be allocated for the entire working thread running for queries. The memory tuner within the RDBMS monitors the PGA memory usage of the entire thread and adjusts the value accordingly to ensure the value does not exceed the maximum limit.</li> <li>- Ex memory operation limit: The maximum amount of memory that can be allocated for a single working thread to use memory (Sort, Hash-Join, and so on) during query executions. The memory tuner within the RDBMS adjusts this value properly when distributing the available memory among a number of working threads simultaneously request memory. The sort and hash-join operation references this value to check whether the operation is possible with its allocated memory. And if not possible, it uses a temporary table space and executes the file I/O operation instead.</li> <li>- Total ex memory allocated: The total amount of PGA memory currently occupied by the entire working thread of RDBMS.</li> <li>- Total ex memory allocated (max): The highest total PGA memory usage it has ever reached after RDBMS boots. In turn, the system keeps the highest value of the 'Total ex memory allocated' item.</li> </ul> |

| Column Name | Data Type | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|-------------|-----------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|             |           | <p>- FIXED pga memory: The fixed amount of memory that the malloc function directly allocates from the system when booted.</p> <p>- ALLOCATED pga memory: The amount of memory allocated using the malloc function and managed by the allocator with the RDBMS. This may increase during the operation.</p> <p>- USED pga memory (from ALLOCATED): The amount of 'ALLOCATED pga memory' currently occupied by server processes. This value must be either less than or equal to the value set in 'ALLOCATED pga memory'. And this value is equal to the sum of each 'pga_used_mem' item of the processes listed via v\$process.</p> <p>- Total workarea memory used: The total amount of memory allocated by the server for query processing.</p> <p>- Total workarea temporary segment used: The size of the temporary segments used for query processing.</p> |
| VALUE       | NUMBER    | Value of the PGA statistic                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |

## 4.2.97. V\$PROCESS

Displays processes and threads used in RDBMS.

- Columns

| Column Name | Data Type   | Description                             |
|-------------|-------------|-----------------------------------------|
| PID         | NUMBER      | Process ID assigned by DBMS             |
| TID         | NUMBER      | Thread ID assigned by DBMS              |
| SPID        | NUMBER      | Process ID assigned by operating system |
| NAME        | VARCHAR(8)  | Thread name                             |
| LATCHWAIT   | VARCHAR(18) | Spinlock waiting time of thread (msec)  |

## 4.2.98. V\$PROC\_CONN\_INFO

Displays status of session connection for each foreground working processes.

- Columns

| Column Name | Data Type  | Description                                |
|-------------|------------|--------------------------------------------|
| PROCESS     | VARCHAR(8) | Name of process                            |
| PID         | NUMBER     | Pid of process                             |
| ENABLED     | VARCHAR(3) | Indicates whether the process is available |
| CONN_OPENED | NUMBER     | Number of sessions opened so far           |
| CONN_CLOSED | NUMBER     | Number of sessions closed so far           |

## 4.2.99. V\$PROXY\_SQLTEXT

Displays texts of SQL statements that are remotely executed through the database link.

- Columns

| Column Name     | Data Type    | Description                                        |
|-----------------|--------------|----------------------------------------------------|
| HASH_VALUE      | NUMBER       | Hash value of the SQL statement                    |
| PLAN_HASH_VALUE | NUMBER       | Hash value of the physical plan                    |
| SQL_ID          | VARCHAR(13)  | SQL identifier                                     |
| ID              | NUMBER       | Number given to each job of a physical plan        |
| LINK_NAME       | VARCHAR(128) | Database link name                                 |
| PIECE           | NUMBER       | Number given to each text of a split SQL statement |
| SQL_TEXT        | VARCHAR(64)  | Text of an SQL statement split into 64-byte texts  |

## 4.2.100. V\$PROXY\_SQL\_PARAM

Displays bind parameters of SQL statements that are remotely executed through the database link.

- Columns

| Column Name     | Data Type   | Description                                 |
|-----------------|-------------|---------------------------------------------|
| HASH_VALUE      | NUMBER      | Hash value of the SQL statement             |
| PLAN_HASH_VALUE | NUMBER      | Hash value of the physical plan             |
| SQL_ID          | VARCHAR(13) | SQL identifier                              |
| ID              | NUMBER      | Number given to each job of a physical plan |
| BIND_NO         | NUMBER      | The order of binding in a proxy node        |
| CLIENT_BIND_NO  | NUMBER      | The order of binding to all plans           |

## 4.2.101. V\$RECOVERY\_FILE\_STATUS

Displays the status of a file that is being recovered.

- Columns

| Column Name | Data Type    | Description                                                     |
|-------------|--------------|-----------------------------------------------------------------|
| FILE#       | NUMBER       | A serial number of the file                                     |
| NAME        | VARCHAR(256) | The pathname of the file                                        |
| STATUS      | CHAR(11)     | The status of the file ("IN RECOVERY" signifies a fixed status) |

## 4.2.102. V\$RECOVERY\_PROGRESS

Displays recovery progress.

- Columns

| Column Name | Data Type    | Description                      |
|-------------|--------------|----------------------------------|
| START_TIME  | TIMESTAMP(6) | Start time of the recovery       |
| END_TIME    | TIMESTAMP(6) | End time of the recovery         |
| TYPE        | CHAR(20)     | Recovery type                    |
| ITEM        | VARCHAR(32)  | Name of the recovery information |

| Column Name | Data Type   | Description                                  |
|-------------|-------------|----------------------------------------------|
|             |             | - UNDO BLOCK COUNT<br>- RECOVERY TIME REMAIN |
| UNITS       | VARCHAR(32) | Unit of the value specified in ITEM column   |
| AMOUNT      | NUMBER      | The amount currently recovered               |
| TOTAL       | NUMBER      | Total amount to be recovered                 |

### 4.2.103. V\$RECOVERY\_STATUS

Displays the status of media recovery progress.

- Columns

| Column Name    | Data Type  | Description                                                                                                                                |
|----------------|------------|--------------------------------------------------------------------------------------------------------------------------------------------|
| RECOVERY_FILES | NUMBER     | Number of datafiles in media recovery                                                                                                      |
| NEXT_START_TSN | NUMBER     | TSN of the next redo log                                                                                                                   |
| MIN_CANCEL_TSN | NUMBER     | The minimum number of TSNs to be the end point of incomplete media recovery                                                                |
| RECO_TSN       | NUMBER     | TSN of the current redo log                                                                                                                |
| REDO_THREADS   | NUMBER     | Number of redo threads needed in media recovery                                                                                            |
| INCOMPLETE     | VARCHAR(1) | Whether or not the current media recovery mode is in complete<br><br>- True: Incomplete media recovery<br>- False: Complete media recovery |
| CANCEL_TSN     | NUMBER     | TSN specified to be the end-point of incomplete media recovery                                                                             |
| CANCEL_DATE    | DATE       | End-point time of the incomplete media recovery                                                                                            |

## 4.2.104. V\$RECOVER\_FILE

Displays files which require media recovery.

- Columns

| Column Name | Data Type    | Description                              |
|-------------|--------------|------------------------------------------|
| FILE#       | NUMBER       | A serial number of a file                |
| ONLINE      | VARCHAR(7)   | Online status of a file                  |
| ERROR       | VARCHAR(200) | Error details of a file to be recovered  |
| CHANGE#     | NUMBER       | TSN for which to start recovery          |
| TIME        | DATE         | Time that the recovery should be started |

## 4.2.105. V\$RESERVED\_WORDS

Displays a list of all SQL keywords.

- Columns

| Column Name | Data Type   | Description                                                |
|-------------|-------------|------------------------------------------------------------|
| KEYWORD     | VARCHAR(30) | Name of the keyword                                        |
| LENGTH      | NUMBER      | Length of the keyword                                      |
| RESERVED    | VARCHAR(1)  | Whether or not the keyword cannot be used as an identifier |

## 4.2.106. V\$RESULT\_CACHE\_DEPENDENCY

Displays the depends-on relationship between cached result and dependencies.

- Columns

| Column Name | Data Type | Description                 |
|-------------|-----------|-----------------------------|
| RESULT_ID   | NUMBER    | ID of the cached result     |
| DEPEND_ID   | NUMBER    | ID of the dependency object |

| Column Name | Data Type | Description                                       |
|-------------|-----------|---------------------------------------------------|
| OBJECT_NO   | NUMBER    | Dictionary object number of the dependency object |

## 4.2.107. V\$RESULT\_CACHE\_OBJECTS

Displays Result Cache Objects and their attributes.

- Columns

| Column Name        | Data Type    | Description                                                                                                                                                       |
|--------------------|--------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ID                 | NUMBER       | ID of the cached object                                                                                                                                           |
| TYPE               | VARCHAR(10)  | Type of the cached object<br>- Result<br>- Dependency                                                                                                             |
| STATUS             | VARCHAR(9)   | status of the object<br>- New: Result is still under construction<br>- Published: Result is available for use<br>- Invalid: Result is no longer available for use |
| BUCKET_NO          | NUMBER       | Internal hash bucket of the object                                                                                                                                |
| HASH               | NUMBER       | Hash value of the object                                                                                                                                          |
| NAME               | VARCHAR(512) | SQL prefix                                                                                                                                                        |
| NAMESPACE          | VARCHAR(3)   | Namespace (SQL or PSM)                                                                                                                                            |
| CREATION_TIMESTAMP | DATE         | Time when the object was created                                                                                                                                  |
| CREATOR_UID        | NUMBER       | ID of the schema that created the object                                                                                                                          |
| DEPEND_COUNT       | NUMBER       | Number of dependencies or dependents                                                                                                                              |
| TSN                | NUMBER       | Build TSN or invalidation TSN                                                                                                                                     |
| COLUMN_COUNT       | NUMBER       | Number of columns in the cached result                                                                                                                            |
| ROW_COUNT          | NUMBER       | Total number of rows in the cached result                                                                                                                         |

| Column Name   | Data Type    | Description                                                        |
|---------------|--------------|--------------------------------------------------------------------|
| PIN_COUNT     | NUMBER       | Number of active scans on this result                              |
| BUILD_TIME    | NUMBER       | Amount of time (in millisecond) it took to build the cached result |
| OBJECT_NO     | NUMBER       | Dictionary object number of the dependency object                  |
| INVALIDATIONS | NUMBER       | Number of times the object has invalidated its dependents          |
| SPACE_USED    | NUMBER       | Used space (in bytes) for the result                               |
| CACHE_ID      | VARCHAR(512) | Cache ID for the result                                            |

## 4.2.108. V\$RESULT\_CACHE\_OBJECT\_STATISTICS

Displays Result Cache Object statistics.

- Columns

| Column Name  | Data Type    | Description                                   |
|--------------|--------------|-----------------------------------------------|
| ID           | NUMBER       | ID of the cached object                       |
| NAME         | VARCHAR(512) | SQL prefix                                    |
| NAMESPACE    | VARCHAR(3)   | Namespace (SQL or UDF)                        |
| SEARCH_COUNT | NUMBER       | Total number of searches in the cached result |
| HIT_COUNT    | NUMBER       | Total number of hits in the cached result     |
| HIT_RATIO    | NUMBER       | Ratio of hits in the cached result            |

## 4.2.109. V\$RESULT\_CACHE\_STATISTICS

Displays various Result Cache settings and usage statistics.

- Columns

| Column Name | Data Type   | Description            |
|-------------|-------------|------------------------|
| ID          | NUMBER      | Statistic number       |
| NAME        | VARCHAR(40) | Name of the statistic  |
| VALUE       | NUMBER      | Value of the statistic |

## 4.2.110. V\$ROLLSTAT

Displays statistics of Undo usage.

- Columns

| Column Name | Data Type | Description                                                               |
|-------------|-----------|---------------------------------------------------------------------------|
| USN         | NUMBER    | Undo segment number                                                       |
| EXTENTS     | NUMBER    | The number of extents                                                     |
| RSSIZE      | NUMBER    | The size of Undo segment                                                  |
| SEQNO       | NUMBER    | Sequence number of undo segment                                           |
| INCNO       | NUMBER    | The number of undo segment incarnation                                    |
| XACTS       | NUMBER    | The number of transactions which are performing the assigned Undo segment |
| HWMSIZE     | NUMBER    | The number of Undo blocks which were already used                         |
| SHRINKS     | NUMBER    | The number of times of shrinking Undo space                               |
| WRAPS       | NUMBER    | The number of times of reusing Undo space                                 |
| EXTENDS     | NUMBER    | The number of times of extending Undo space                               |
| SHRINK_SIZE | NUMBER    | The number of Undo blocks shrinking Undo space                            |
| ACTIVE_SIZE | NUMBER    | The number of Undo blocks currently in use                                |
| STATUS      | NUMBER    | Status information ("ONLINE" signifies a fixed status)                    |
| CUREXT      | NUMBER    | Extent that is currently in use                                           |
| CURSIZE     | NUMBER    | The size of extent currently in use                                       |

| Column Name     | Data Type | Description                                       |
|-----------------|-----------|---------------------------------------------------|
| CURBLK          | NUMBER    | Block number in use by the current extent         |
| MAX_ACTIVE_SIZE | NUMBER    | The maximum number of blocks used by Undo segment |

## 4.2.111. V\$ROWCACHE

Displays details of status information on DD cache.

- Columns

| Column Name  | Data Type   | Description                                                           |
|--------------|-------------|-----------------------------------------------------------------------|
| NAME         | VARCHAR(16) | DD table name                                                         |
| SEARCH_KEY   | NUMBER      | Search key value in DD cache                                          |
| BUCKETSET_ID | NUMBER      | Bucketset ID in the search key                                        |
| ENTRY_CNT    | NUMBER      | The number of entries cached in the bucketset                         |
| HIT_CNT      | NUMBER      | The number of times that DD cache succeeded                           |
| MISS_CNT     | NUMBER      | The number of times that DD cache failed                              |
| CACHE_SIZE   | NUMBER      | The size of the bucketset (in bytes)                                  |
| PINNED_CNT   | NUMBER      | The number of entries pinned in the bucketset                         |
| INVALID_CNT  | NUMBER      | The number of invalid entries in the bucketset                        |
| NE_CNT       | NUMBER      | The number of entries for objects which do not exist in the bucketset |

## 4.2.112. V\$RSBSTAT

Displays Stats of RSBs

- Columns

| Column Name | Data Type | Description        |
|-------------|-----------|--------------------|
| CTX         | CHAR(3)   | Context (CCC, CWS) |

| Column Name    | Data Type | Description                                                |
|----------------|-----------|------------------------------------------------------------|
| MST_RSB        | NUMBER    | Count of Master RSBs                                       |
| SHD_RSB        | NUMBER    | Count of Shadow RSBs                                       |
| RECLAIMED      | NUMBER    | Accumulated Count of reclaimed RSBs                        |
| FAILED_BH      | NUMBER    | Accumulated Count of RSB reclamation failures (cached bh)  |
| FAILED_USING   | NUMBER    | Accumulated Count of RSB reclamation failures (in use)     |
| FAILED_GLOBAL  | NUMBER    | Accumulated Count of RSB reclamation failures (global rsb) |
| FAILED_PIN     | NUMBER    | Accumulated Count of RSB reclamation failures (pinned)     |
| FAILED_INVALID | NUMBER    | Accumulated Count of RSB reclamation failures (on invalid) |

#### 4.2.113. V\$RSRC\_CONSUMER\_GROUP

Displays data related to currently active resource consumer groups

- Columns

| Column Name          | Data Type   | Description                                                                                                                     |
|----------------------|-------------|---------------------------------------------------------------------------------------------------------------------------------|
| NAME                 | VARCHAR(32) | Name of the consumer group                                                                                                      |
| ACTIVE_SESSIONS      | NUMBER      | Number of currently active sessions in the consumer group                                                                       |
| CONSUMED_CPU_TIME    | NUMBER      | Cumulative amount of CPU time consumed by all sessions in the consumer group                                                    |
| YIELDS               | NUMBER      | Cumulative number of times that sessions in the consumer group had to yield CPU to other sessions because of quantum expiration |
| SMALL_READ_MEGABYTES | NUMBER      | Number of single block megabytes read                                                                                           |
| LARGE_READ_MEGABYTES | NUMBER      | Number of multiblock megabytes read                                                                                             |

| Column Name             | Data Type | Description                          |
|-------------------------|-----------|--------------------------------------|
| SMALL_READ_RE<br>QUESTS | NUMBER    | Number of single block read requests |
| LARGE_READ_RE<br>QUESTS | NUMBER    | Number of multiblock read requests   |

#### 4.2.114. V\$RSRC\_PLAN

Displays the names of all currently active resource plans

- Columns

| Column Name | Data Type   | Description                                                 |
|-------------|-------------|-------------------------------------------------------------|
| NAME        | VARCHAR(32) | Name of the resource plan                                   |
| IS_TOP_PLAN | VARCHAR(5)  | Indicates whether the resource plan is the current top plan |

#### 4.2.115. V\$RSRC\_SESSION\_INFO

Displays resource manager statistics per session

- Columns

| Column Name                | Data Type   | Description                                                                                         |
|----------------------------|-------------|-----------------------------------------------------------------------------------------------------|
| SID                        | NUMBER      | Session identifier                                                                                  |
| CURRENT_CON<br>SUMER_GROUP | VARCHAR(32) | The name of the consumer group in which the session currently belongs                               |
| STATE                      | VARCHAR(32) | Current state of the session                                                                        |
| CON<br>SUMED_CPU_TIME      | NUMBER      | Cumulative amount of CPU time consumed by the session                                               |
| YIELDS                     | NUMBER      | Cumulative number of times the session had to yield CPU to other sessions due to quantum expiration |
| SMALL_READ_MEGABYTES       | NUMBER      | Total number of single block megabytes read by the session                                          |

| Column Name          | Data Type | Description                                               |
|----------------------|-----------|-----------------------------------------------------------|
| LARGE_READ_MEGABYTES | NUMBER    | Total number of multiblock megabytes read by the session  |
| SMALL_READ_REQUESTS  | NUMBER    | Total number of single block read requests by the session |
| LARGE_READ_REQUESTS  | NUMBER    | Total number of multiblock read requests by the session   |

#### 4.2.116. V\$RUNNING\_JOBS

Displays information on running jobs.

- Columns

| Column Name | Data Type | Description                                          |
|-------------|-----------|------------------------------------------------------|
| SID         | NUMBER    | Session ID of the currently running job              |
| JOB_ID      | NUMBER    | JOB ID of the currently running job                  |
| START_TIME  | DATE      | Start time of the job                                |
| FAILURE     | NUMBER    | Job failure count                                    |
| INSTANCE    | NUMBER    | Instance name that started the currently running job |

#### 4.2.117. V\$RUNNING\_JOBS\_WITH\_NAME

Displays information on running jobs with name.

- Columns

| Column Name | Data Type    | Description                             |
|-------------|--------------|-----------------------------------------|
| SID         | NUMBER       | Session ID of the currently running job |
| JOB_ID      | NUMBER       | JOB ID of the currently running job     |
| JOB_NAME    | VARCHAR(100) | JOB name of the currently running job   |
| START_TIME  | DATE         | Start time of the job                   |

| Column Name | Data Type | Description                                          |
|-------------|-----------|------------------------------------------------------|
| FAILURE     | NUMBER    | Job failure count                                    |
| INSTANCE    | NUMBER    | Instance name that started the currently running job |

## 4.2.118. V\$SEGMENT\_STATISTICS

!!! No documentation yet !!!

- Columns

| Column Name     | Data Type    | Description                     |
|-----------------|--------------|---------------------------------|
| ID              | NUMBER       | Object ID                       |
| NAME            | VARCHAR(128) | Object name                     |
| TYPE            | VARCHAR(20)  | Object type                     |
| STATISTICS#     | NUMBER       | Segment Statistics Type         |
| STATISTICS_NAME | VARCHAR(64)  | Name of the Segment Statistics  |
| VALUE           | NUMBER       | Value of the Segment Statistics |

## 4.2.119. V\$SERVER\_EVENT\_STATUS

Displays information about the server event status

- Columns

| Column Name | Data Type  | Description                            |
|-------------|------------|----------------------------------------|
| IS_READY    | VARCHAR(5) | Status if or not can use server event. |

## 4.2.120. V\$SERVER\_EVENT\_SUBTYPE

Displays information about the server event subtype

- Columns

| Column Name | Data Type    | Description                                  |
|-------------|--------------|----------------------------------------------|
| NAME        | VARCHAR(32)  | Name of server event subtype.                |
| VALUE       | NUMBER       | Value of server event subtype.               |
| DESC        | VARCHAR(300) | Description of server event subtype.         |
| PARENT_NAME | VARCHAR(32)  | Name of parent type of server event subtype. |

#### 4.2.121. V\$SERVER\_EVENT\_TYPE

Displays information about the server event type

- Columns

| Column Name | Data Type    | Description                       |
|-------------|--------------|-----------------------------------|
| NAME        | VARCHAR(32)  | Name of server event type.        |
| VALUE       | NUMBER       | Value of server event type.       |
| DESC        | VARCHAR(300) | Description of server event type. |

#### 4.2.122. V\$SESSION

Displays information on each session.

- Columns

| Column Name | Data Type    | Description                                                         |
|-------------|--------------|---------------------------------------------------------------------|
| SID         | NUMBER       | Session ID                                                          |
| SERIAL#     | NUMBER       | A serial number of a session (Like SID, used to recognize sessions) |
| AUDSID      | NUMBER       | A secondary serial number of a session                              |
| USER#       | NUMBER       | The current user's ID                                               |
| USERNAME    | VARCHAR(128) | The current user's name                                             |
| IPADDR      | VARCHAR(21)  | IP to which the user connects                                       |

| Column Name | Data Type    | Description                                                                                                                                                                                                                                                                                                                                                                                                    |
|-------------|--------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| COMMAND     | NUMBER       | Currently performing SQL type<br>- 1: SELECT<br>- 2: INSERT<br>- 3: UPDATE<br>- 4: DELETE<br>- 5: MERGE<br>- 6: CALL                                                                                                                                                                                                                                                                                           |
| STATUS      | VARCHAR(32)  | Status of session<br>- READY: The session is ready<br>- RUNNING: The session is running<br>- TX_RECOVERING: The transaction is being recovered<br>- SESS_CLEANUP: The session resources are being cleaned up<br>- ASSIGNED: The session has an assigned thread, but is not ready yet<br>- CLOSING: The session is being closed<br>- ROLLING_BACK: The statement-level transaction of PE slave is rolling back. |
| SCHEMA#     | NUMBER       | The current schema ID                                                                                                                                                                                                                                                                                                                                                                                          |
| SCHEMANAME  | VARCHAR(128) | The current schema name                                                                                                                                                                                                                                                                                                                                                                                        |
| TYPE        | VARCHAR(15)  | Session type<br>- WTHR: Working thread<br>- CTHR: Control thread<br>- LGWR: Log writing process<br>- CKPT: Checkpoint process<br>- LARC: Log archive<br>- AGENT: Sequence process<br>- MTHR: Monitoring process                                                                                                                                                                                                |

| Column Name       | Data Type    | Description                                                                                                                                                                                                                                                                                                                                                     |
|-------------------|--------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                   |              | <ul style="list-style-type: none"> <li>- DBWR: Datablock writing process</li> <li>- LNW: Log network writing process</li> </ul>                                                                                                                                                                                                                                 |
| SQL_ID            | VARCHAR(13)  | ID of the currently running SQL                                                                                                                                                                                                                                                                                                                                 |
| SQL_CHILD_NUMBER  | NUMBER       | Child number of the currently running SQL                                                                                                                                                                                                                                                                                                                       |
| SUB_SQL_ID        | VARCHAR(13)  | Sub SQL ID of currently running SQL                                                                                                                                                                                                                                                                                                                             |
| SUB_CHILD_NUMBER  | NUMBER       | Child number of the current running Sub SQL                                                                                                                                                                                                                                                                                                                     |
| PREV_SQL_ID       | VARCHAR(13)  | ID of the last executed SQL                                                                                                                                                                                                                                                                                                                                     |
| PREV_CHILD_NUMBER | NUMBER       | Child number of the last executed SQL                                                                                                                                                                                                                                                                                                                           |
| SQL_ET            | NUMBER       | Elapsed time (ms) of SQL that is being performed                                                                                                                                                                                                                                                                                                                |
| LOGON_TIME        | DATE         | Logon time                                                                                                                                                                                                                                                                                                                                                      |
| STATE             | VARCHAR(25)  | State of working thread <ul style="list-style-type: none"> <li>- INVALID: Not intialized</li> <li>- NEW: Being created</li> <li>- IDLE: Ready to run</li> <li>- RUNNING: Running</li> <li>- WAITING: Waiting for internal message</li> <li>- RECV_WAITING: Waiting for client message</li> <li>- STOP_BY_MTHR: Stopped by MTHR</li> <li>- DEAD: Dead</li> </ul> |
| WLOCK_WAIT        | VARCHAR(100) | Type of wlock for which the session waits                                                                                                                                                                                                                                                                                                                       |
| SEQ#              | NUMBER       | Sequence of wait event                                                                                                                                                                                                                                                                                                                                          |
| WAIT_EVENT        | NUMBER       | If the session is currently waiting, then the type of wait_event for which the session is waiting. If the session is not in a wait, then the value is -1.                                                                                                                                                                                                       |

| Column Name                | Data Type    | Description                                                                                                                                                                                                                        |
|----------------------------|--------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| WAIT_TIME                  | NUMBER       | If the session is currently waiting, then the value is the time spent in the wait (in 0.1 seconds). If the session is currently not in a wait, then the value is 0.                                                                |
| LAST_WAIT_EVENT#           | NUMBER       | If the session is currently in a wait, then the type of wait_event for which the session is waiting. If the session is not in a wait, then the type of wait_event for which the session most recently waited.                      |
| LAST_WAIT_EVENT            | VARCHAR(100) | If the session is currently in a wait, then the type name of the wait_event for which the session is waiting. If the session is not in a wait, then the type name of wait_event for which the session most recently waited.        |
| LAST_WAIT_TIME             | NUMBER       | If the session is currently waiting, then the value is the time spent in the wait (in milliseconds). If the session is not in a wait, then the value is the amount of time waited in the most recent wait_event (in milliseconds). |
| LAST_WAIT_TIME_MICRO       | NUMBER       | If the session is currently waiting, then the value is the time spent in the wait (in microseconds). If the session is not in a wait, then the value is the amount of time waited in the most recent wait_event (in microseconds). |
| TIME_SINCE_LAST_WAIT       | NUMBER       | If the session is currently not in a wait, then the value is the time elapsed since the end of the last wait_event (in milliseconds). If the session is currently waiting, then the value is 0.                                    |
| TIME_SINCE_LAST_WAIT_MICRO | NUMBER       | If the session is currently not in a wait, then the value is the time elapsed since the end of the last wait_event (in microseconds). If the session is currently waiting, then the value is 0.                                    |
| LATEST_STAT#               | NUMBER       | Number of stat that the session most recently updated                                                                                                                                                                              |
| PGA_USED_MEM               | NUMBER       | The amount of PGA memory occupied by the session                                                                                                                                                                                   |
| SQL_TRACE                  | VARCHAR(8)   | Whether a session uses SQL trace                                                                                                                                                                                                   |
| PROG_NAME                  | VARCHAR(30)  | Client program name                                                                                                                                                                                                                |
| CLIENT_PID                 | NUMBER       | Client PID                                                                                                                                                                                                                         |

| Column Name       | Data Type    | Description                                                             |
|-------------------|--------------|-------------------------------------------------------------------------|
| PID               | NUMBER       | Identifier of a process to which a session belongs                      |
| WTHR_ID           | NUMBER       | Index of a worker thread to which a session belongs                     |
| OS_THR_ID         | NUMBER       | Identifier of thread generated from OS to which a session belongs       |
| OSUSER            | VARCHAR(128) | OS account name of the session connected                                |
| MACHINE           | VARCHAR(64)  | Host name of the session connected                                      |
| TERMINAL          | VARCHAR(16)  | Terminal(tty) information of the session connected                      |
| MODULE            | VARCHAR(64)  | Name of a module specified by dbms_application_info.set_module          |
| ACTION            | VARCHAR(64)  | Name of an action specified by dbms_application_info.set_module/action  |
| CLIENT_INFO       | VARCHAR(64)  | Name of client_info specified by dbms_application_info.set_client_info  |
| CLIENT_IDENTIFIER | VARCHAR(64)  | Name of client ID specified by dbms_session.set_identifier              |
| PDML_ENABLED      | VARCHAR(3)   | Parallel dml enabled, YES or NO                                         |
| PDML_STATUS       | VARCHAR(8)   | Parallel dml status, FORCE, ENABLED or DISABLED                         |
| PDDL_STATUS       | VARCHAR(8)   | Parallel ddl status, FORCE, ENABLED or DISABLED                         |
| PQ_STATUS         | VARCHAR(8)   | Parallel query status, FORCE, ENABLED or DISABLED                       |
| ROW_WAIT_OBJ_ID   | NUMBER       | Object ID of the row that causes a lock when waiting for a row lock.    |
| ROW_WAIT_FILE_NO  | NUMBER       | File number of the row that causes a lock when waiting for a row lock.  |
| ROW_WAIT_BLOCK_NO | NUMBER       | Block number of the row that causes a lock when waiting for a row lock. |
| ROW_WAIT_ROW_NO   | NUMBER       | Number of row that causes a lock when waiting for a row lock.           |

| Column Name       | Data Type   | Description                                                           |
|-------------------|-------------|-----------------------------------------------------------------------|
| CONSUMER_GROUP    | VARCHAR(32) | The name of the consumer group in which the session currently belongs |
| CONSUMED_CPU_TIME | NUMBER      | Cumulative amount of CPU time consumed by the session                 |

### 4.2.123. V\$SESSION\_EVENT

Displays events that are performed in each session.

- Columns

| Column Name    | Data Type    | Description                                              |
|----------------|--------------|----------------------------------------------------------|
| TID            | NUMBER       | Session ID                                               |
| IDX            | NUMBER       | Event number                                             |
| NAME           | VARCHAR(100) | Name of the event                                        |
| CLASS          | VARCHAR(40)  | Class of the event                                       |
| DESC           | VARCHAR(100) | Description of the event                                 |
| TOTAL_WAITS    | NUMBER       | Total number of event calls                              |
| TOTAL_TIMEOUTS | NUMBER       | The number of times that a time-out occurred in an event |
| TIME_WAITED    | NUMBER       | The amount of time waited for this event (msec)          |
| AVERAGE_WAIT   | NUMBER       | The average amount of time waited for this event (msec)  |
| MAX_WAIT       | NUMBER       | The maximum amount of time waited for this event (msec)  |

### 4.2.124. V\$SESSION\_IO

Displays input/output which occurred in each session.

- Columns

| Column Name        | Data Type | Description                                                           |
|--------------------|-----------|-----------------------------------------------------------------------|
| SID                | NUMBER    | Session ID                                                            |
| BLOCK_GETS         | NUMBER    | The number of times of reading blocks in Current mode                 |
| CONSISTENT_GETS    | NUMBER    | The number of times of reading blocks in Consistent Read (CR) mode    |
| PHYSICAL_READS     | NUMBER    | The number of blocks read in the disk                                 |
| BLOCK_CHANGES      | NUMBER    | The number of times of changing a buffer read in Current mode         |
| CONSISTENT_CHANGES | NUMBER    | The number of times of changing a buffer read in Consistent Read mode |

#### 4.2.125. V\$SESSION\_LATEST\_STAT

Displays the most recently updated stat information of each session.

- Columns

| Column Name | Data Type   | Description     |
|-------------|-------------|-----------------|
| SID         | NUMBER      | Session ID      |
| STAT#       | NUMBER      | Number of event |
| NAME        | VARCHAR(40) | Name of event   |

#### 4.2.126. V\$SESSION\_LONGOPS

Displays status of sessions taking longer time than specified in LONGOPS\_THRESHOLD\_SEC seconds.

- Columns

| Column Name | Data Type | Description                                                         |
|-------------|-----------|---------------------------------------------------------------------|
| SID         | NUMBER    | Session ID                                                          |
| SERIAL#     | NUMBER    | A serial number of a session (Like SID, used to recognize sessions) |

| Column Name  | Data Type    | Description                                                                |
|--------------|--------------|----------------------------------------------------------------------------|
| USER_NAME    | VARCHAR(128) | Name of the current user                                                   |
| WTHR_ID      | NUMBER       | Index of a working thread process to which a session belongs               |
| SQL_ID       | VARCHAR(13)  | SQL identifier                                                             |
| SQL_PLAN_ID  | NUMBER       | SQL physical plan identifier                                               |
| OPNAME       | VARCHAR(128) | Name of the operation                                                      |
| TARGET       | VARCHAR(128) | Name of the target object                                                  |
| SOFAR        | NUMBER       | The work throughput done so far                                            |
| TOTALWORK    | NUMBER       | The overall work throughput                                                |
| UNITS        | VARCHAR(128) | Unit of the work throughput                                                |
| START_TIME   | DATE         | Start time of the session                                                  |
| ELAPSED_TIME | NUMBER       | Total time for which a particular job has been performed (in milliseconds) |
| END_TIME     | DATE         | End time of the session                                                    |

## 4.2.127. V\$SESSION\_WAIT

Displays the waiting status of each session.

- Columns

| Column Name | Data Type    | Description            |
|-------------|--------------|------------------------|
| SID         | NUMBER       | Session ID             |
| NAME        | VARCHAR(100) | Name of waiting event  |
| ID1         | NUMBER       | ID1 of waiting target  |
| ID2         | NUMBER       | ID2 of waiting target  |
| SEQ#        | NUMBER       | Sequence of wait event |

| Column Name | Data Type | Description                                |
|-------------|-----------|--------------------------------------------|
| TIME_WAITED | NUMBER    | Time for which an event was waiting (msec) |
| TIMEOUT     | NUMBER    | Time-out value (msec)                      |

## 4.2.128. V\$SESSTAT

Displays statistics of each session.

- Columns

| Column Name | Data Type   | Description                                          |
|-------------|-------------|------------------------------------------------------|
| SID         | NUMBER      | Session ID                                           |
| STAT#       | NUMBER      | Number of event to obtain statistics                 |
| NAME        | VARCHAR(60) | Name of event to obtain statistics                   |
| CLASS       | NUMBER      | Class of the event                                   |
| VALUE       | NUMBER      | Value of the event (Corresponding to the event name) |

## 4.2.129. V\$SESS\_TIME\_MODEL

Displays the session-level time statistics.

- Columns

| Column Name | Data Type   | Description                      |
|-------------|-------------|----------------------------------|
| SID         | NUMBER      | Session ID                       |
| STAT_ID     | NUMBER      | Identifier of a time statistics  |
| STAT_NAME   | VARCHAR(64) | Name of a time statistics        |
| VALUE       | NUMBER      | Amount of time (in microseconds) |

## 4.2.130. V\$SGA

Displays information on shared memory used in the database.

- Columns

| Column Name | Data Type   | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|-------------|-------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| NAME        | VARCHAR(40) | <p>Name of shared memory space</p> <p>- SHARED MEMORY: Total shared memory size allocated during instance startup according to the value of TOTAL_SHM_SIZE.</p> <p>- FIXED MEMORY: Total fixed memory area size fixed for database buffers, redo buffers, etc.</p> <p>- SHARED POOL MEMORY: Total/used shared pool size for various purposes. Memory is allocated on demand using a shared pool allocator specific to the purpose. The size of the allocators can be extended as long as the pool is available.</p> <p>- SHARED POOL ALLOCATORS (LC): Total/used memory size of the allocators for library cache memory.</p> <p>- SHARED POOL ALLOCATORS (DD): Total/used memory size of the allocators for data dictionary cache memory.</p> <p>- SHARED POOL ALLOCATORS (SLAB): Total/used memory size of the allocators for slab memory.</p> <p>- SHARED POOL ALLOCATORS (MISC): Total/used memory size of the allocators for miscellaneous memory.</p> <p>- SHARED POOL ALLOCATORS (LONG-TERM): Total/used memory size of the allocators for long-term memory.</p> <p>- SHARED POOL ALLOCATORS (Total): Total/used memory size of the allocators for all purposes. This size equals the sum of all allocator sizes.</p> <p>- Database Buffers: Total database buffer size allocated since instance startup according to the value of DB_CACHE_SIZE.</p> |

| Column Name | Data Type | Description                                                                                                    |
|-------------|-----------|----------------------------------------------------------------------------------------------------------------|
|             |           | - Redo Buffers: Total redo buffer size allocated during instance startup according to the value of LOG_BUFFER. |
| TOTAL       | NUMBER    | Size of shared memory space                                                                                    |
| USED        | NUMBER    | Usage of shared memory space                                                                                   |

### 4.2.131. V\$SGASTAT

Displays information on the shared pool for a database, such as its purpose and how much of the shared pool is used.

- Columns

| Column Name | Data Type   | Description                           |
|-------------|-------------|---------------------------------------|
| NAME        | VARCHAR(40) | Purpose for which shared pool is used |
| SIZE        | NUMBER      | Usage of shared pool                  |

### 4.2.132. V\$SLOG

Displays slog from the current system log file. Up to SLOG\_VIEW\_SIZE bytes are stored in the view.

- Columns

| Column Name | Data Type     | Description  |
|-------------|---------------|--------------|
| TIME        | TIMESTAMP(6)  | TIME         |
| MODULE      | VARCHAR(3)    | Module id    |
| SESSION_ID  | NUMBER        | Session id   |
| LOG_LEVEL   | VARCHAR(1)    | Log level    |
| MESSAGE     | VARCHAR(4000) | Message text |

### 4.2.133. V\$SPINLOCK HOLDER

Displays spinlock information.

- Columns

| Column Name | Data Type   | Description                                 |
|-------------|-------------|---------------------------------------------|
| PID         | NUMBER      | Process ID                                  |
| SID         | NUMBER      | ID of the session that owns the spinlock    |
| NAME        | VARCHAR(30) | Spinlock Name                               |
| ADDR        | VARCHAR(16) | Spinlock address                            |
| GETS        | NUMBER      | The number of times a spinlock was acquired |

### 4.2.134. V\$SPINUSE\_STAT

Displays statistics of Spinlock.

- Columns

| Column Name | Data Type   | Description                                                                                |
|-------------|-------------|--------------------------------------------------------------------------------------------|
| LOCATION    | VARCHAR(60) | Location of Spinlock                                                                       |
| SLEEPS_CNT  | NUMBER      | The number of cases that a LOCK is possessed in LOCATION while sleeping to obtain Spinlock |

### 4.2.135. V\$SQL

Displays details of a SQL child cursor execution.

- Columns

| Column Name  | Data Type   | Description                            |
|--------------|-------------|----------------------------------------|
| SQL_ID       | VARCHAR(13) | SQL identifier                         |
| CHILD_NUMBER | NUMBER      | The number of this child physical plan |
| HASH_VALUE   | NUMBER      | Hash value of the SQL statement        |

| Column Name           | Data Type      | Description                                                                          |
|-----------------------|----------------|--------------------------------------------------------------------------------------|
| PLAN_HASH_VALUE       | NUMBER         | Hash value of the physical plan                                                      |
| FIRST_LOAD_TIME       | DATE           | Time that an SQL statement was first loaded                                          |
| LAST_ACTIVE_TIME      | DATE           | Last time that statistics were collected                                             |
| PARSING_USER_ID       | NUMBER         | Identifier of user that parsed an SQL statement for the first time                   |
| PARSING_SCHEMA_ID     | NUMBER         | Identifier of schema that parsed an SQL statement for the first time                 |
| MODULE                | VARCHAR(64)    | Name of module that had been set when an SQL statement was parsed for the first time |
| ACTION                | VARCHAR(64)    | Name of action that had been set when an SQL statement was first parsed              |
| SQL_TEXT              | VARCHAR(30000) | SQL text of the current cursor                                                       |
| IS_SHAREABLE          | VARCHAR(1)     | Indicates whether this child cursor can be shared                                    |
| CPU_TIME              | NUMBER         | CPU time (microseconds)                                                              |
| ELAPSED_TIME          | NUMBER         | Total execution time                                                                 |
| PARSE_CALLS           | NUMBER         | The number of parsing requests                                                       |
| ROWS_PROCESSED        | NUMBER         | Total number of processed rows                                                       |
| FETCHES               | NUMBER         | The number of fetching times                                                         |
| EXECUTIONS            | NUMBER         | The number of times executing an SQL statement                                       |
| END_OF_FETCH_COUNT    | NUMBER         | The number of times this SQL statement was fully executed                            |
| PX_SERVERS_EXECUTIONS | NUMBER         | The number of times executing parallel execution slaves                              |
| APPLICATION_WAIT_TIME | NUMBER         | Total amount of application wait time                                                |
| CONCURRENCY_WAIT_TIME | NUMBER         | Total amount of concurrency wait time                                                |

| Column Name             | Data Type   | Description                                          |
|-------------------------|-------------|------------------------------------------------------|
| CLUSTER_WAIT_TIME       | NUMBER      | Total amount of cluster wait time                    |
| USER_IO_WAIT_TIME       | NUMBER      | Total amount of I/O wait time (by user)              |
| DISK_READS              | NUMBER      | Disk read count                                      |
| DIRECT_WRITES           | NUMBER      | Direct write count                                   |
| BUFFER_GETS             | NUMBER      | Total number of buffers read in Consistent-Read mode |
| PHYSICAL_READ_REQUESTS  | NUMBER      | Disk read request count                              |
| PHYSICAL_READ_BYTES     | NUMBER      | Total number of bytes read from disks (in bytes)     |
| PHYSICAL_WRITE_REQUESTS | NUMBER      | Disk write request count                             |
| PHYSICAL_WRITE_BYTES    | NUMBER      | Total number of bytes write to disks (in bytes)      |
| SORTS                   | NUMBER      | The number of sorts performed                        |
| OBJECT_STATUS           | VARCHAR(14) | Status of the cursor                                 |
| PROGRAM_ID              | NUMBER      | Program identifier                                   |
| PROGRAM_LINE#           | NUMBER      | Program line number                                  |

## 4.2.136. V\$SQLAREA

Displays details of SQL statement execution.

- Columns

| Column Name      | Data Type   | Description                              |
|------------------|-------------|------------------------------------------|
| SQL_ID           | VARCHAR(13) | SQL identifier                           |
| HASH_VALUE       | NUMBER      | Hash value of the SQL statement          |
| PLAN_HASH_VALUE  | NUMBER      | Hash value of the physical plan          |
| LAST_ACTIVE_TIME | DATE        | Last time that statistics were collected |

| Column Name           | Data Type      | Description                                                                          |
|-----------------------|----------------|--------------------------------------------------------------------------------------|
| FIRST_LOAD_TIME       | DATE           | Time that an SQL statement was first loaded                                          |
| PARSING_USER_ID       | NUMBER         | Identifier of user that parsed an SQL statement for the first time                   |
| PARSING_SCHEMA_ID     | NUMBER         | Identifier of schema that parsed an SQL statement for the first time                 |
| MODULE                | VARCHAR(64)    | Name of module that had been set when an SQL statement was parsed for the first time |
| ACTION                | VARCHAR(64)    | Name of action that had been set when an SQL statement was first parsed              |
| SQL_TEXT              | VARCHAR(65532) | SQL text of the current cursor                                                       |
| CPU_TIME              | NUMBER         | CPU time (microseconds)                                                              |
| ELAPSED_TIME          | NUMBER         | Total execution time                                                                 |
| PARSE_CALLS           | NUMBER         | The number of parsing requests                                                       |
| ROWS_PROCESSED        | NUMBER         | Total number of processed rows                                                       |
| FETCHES               | NUMBER         | The number of fetching times                                                         |
| EXECUTIONS            | NUMBER         | The number of times executing an SQL statement                                       |
| END_OF_FETCH_COUNT    | NUMBER         | The number of times this SQL statement was fully executed                            |
| PX_SERVERS_EXECUTIONS | NUMBER         | The number of times executing parallel execution slaves                              |
| APPLICATION_WAIT_TIME | NUMBER         | Total amount of application wait time                                                |
| CONCURRENCY_WAIT_TIME | NUMBER         | Total amount of concurrency wait time                                                |
| CLUSTER_WAIT_TIME     | NUMBER         | Total amount of cluster wait time                                                    |
| USER_IO_WAIT_TIME     | NUMBER         | Total amount of I/O wait time (by user)                                              |
| DISK_READS            | NUMBER         | Disk read count                                                                      |

| Column Name             | Data Type | Description                                          |
|-------------------------|-----------|------------------------------------------------------|
| DIRECT_WRITES           | NUMBER    | Direct write count                                   |
| BUFFER_GETS             | NUMBER    | Total number of buffers read in Consistent-Read mode |
| PHYSICAL_READ_REQUESTS  | NUMBER    | Disk read request count                              |
| PHYSICAL_READ_BYTES     | NUMBER    | Total number of bytes read from disks (in bytes)     |
| PHYSICAL_WRITE_REQUESTS | NUMBER    | Disk write request count                             |
| PHYSICAL_WRITE_BYTES    | NUMBER    | Total number of bytes write to disks (in bytes)      |
| SORTS                   | NUMBER    | The number of sorts performed                        |
| PROGRAM_ID              | NUMBER    | Program identifier                                   |
| PROGRAM_LINE#           | NUMBER    | Program line number                                  |

## 4.2.137. V\$SQLSTATS

Displays information on SQL statement execution.

- Columns

| Column Name               | Data Type     | Description                                                           |
|---------------------------|---------------|-----------------------------------------------------------------------|
| SQL_TEXT                  | VARCHAR(1000) | First thousand characters of the SQL text                             |
| SQL_ID                    | VARCHAR(13)   | SQL identifier                                                        |
| HASH_VALUE                | NUMBER        | Hash value of the SQL statement                                       |
| LAST_ACTIVE_TIME          | DATE          | Last time that statistics were collected                              |
| LAST_ACTIVE_CHILD_ADDRESS | RAW(8)        | Address of the child physical plan when the statistics were collected |
| PLAN_HASH_VALUE           | NUMBER        | Hash value of the physical plan                                       |

| Column Name             | Data Type | Description                                                  |
|-------------------------|-----------|--------------------------------------------------------------|
| SHAREABLE_MEM           | NUMBER    | Total shared memory (in bytes) occupied by all child cursors |
| CPU_TIME                | NUMBER    | CPU time (microseconds)                                      |
| ELAPSED_TIME            | NUMBER    | Total execution time (in microseconds)                       |
| PARSE_CALLS             | NUMBER    | The number of parsing requests                               |
| ROWS_PROCESSED          | NUMBER    | Total number of processed rows                               |
| FETCHES                 | NUMBER    | The number of fetching times                                 |
| EXECUTIONS              | NUMBER    | The number of times executing an SQL statement               |
| END_OF_FETCH_COUNT      | NUMBER    | The number of times this SQL statement was fully executed    |
| PX_SERVERS_EXECUTIONS   | NUMBER    | The number of times executing parallel execution slaves      |
| APPLICATION_WAIT_TIME   | NUMBER    | Total amount of application wait time                        |
| CONCURRENCY_WAIT_TIME   | NUMBER    | Total amount of concurrency wait time                        |
| CLUSTER_WAIT_TIME       | NUMBER    | Total amount of cluster wait time                            |
| USER_IO_WAIT_TIME       | NUMBER    | Total amount of I/O wait time (by user)                      |
| DISK_READS              | NUMBER    | Disk read count                                              |
| DIRECT_WRITES           | NUMBER    | Direct write count                                           |
| BUFFER_GETS             | NUMBER    | Total number of buffers read in Consistent-Read mode         |
| PHYSICAL_READ_REQUESTS  | NUMBER    | Disk read request count                                      |
| PHYSICAL_READ_BYTES     | NUMBER    | Total number of bytes read from disks (in bytes)             |
| PHYSICAL_WRITE_REQUESTS | NUMBER    | Disk write request count                                     |

| Column Name             | Data Type | Description                                     |
|-------------------------|-----------|-------------------------------------------------|
| PHYSICAL_WRITE_BYTES    | NUMBER    | Total number of bytes write to disks (in bytes) |
| SORTS                   | NUMBER    | The number of sorts performed                   |
| DISK_READ_TIME          | NUMBER    | Total disk read time                            |
| TEMP_SEGMENT_READ_TIME  | NUMBER    | Total temp segment read time                    |
| TEMP_SEGMENT_WRITE_TIME | NUMBER    | Total temp segment write time                   |

## 4.2.138. V\$SQLTEXT

Displays texts of SQL statements belonging to the cursor.

- Columns

| Column Name     | Data Type   | Description                                        |
|-----------------|-------------|----------------------------------------------------|
| HASH_VALUE      | NUMBER      | Hash value of SQL statement                        |
| PLAN_HASH_VALUE | NUMBER      | Hash value of the physical plan                    |
| SQL_ID          | VARCHAR(13) | SQL identifier                                     |
| CHILD_NUMBER    | NUMBER      | The number of this child physical plan             |
| COMMAND_TYPE    | NUMBER      | Number code of SQL statement type                  |
| PIECE           | NUMBER      | Number given to each text of a split SQL statement |
| SQL_TEXT        | VARCHAR(64) | Text of an SQL statement split into 64-byte texts  |

- See also

[V\\$SQLTEXT2](#)

[V\\$SQLTEXT\\_WITH\\_NEWLINES](#)

[V\\$SQLTEXT\\_WITH\\_NEWLINES2](#)

## 4.2.139. V\$SQLTEXT2

Displays texts of SQL statements belonging to the cursor.

- Columns

| Column Name     | Data Type      | Description                                          |
|-----------------|----------------|------------------------------------------------------|
| HASH_VALUE      | NUMBER         | Hash value of SQL statement                          |
| PLAN_HASH_VALUE | NUMBER         | Hash value of the physical plan                      |
| SQL_ID          | VARCHAR(13)    | SQL identifier                                       |
| CHILD_NUMBER    | NUMBER         | The number of this child physical plan               |
| COMMAND_TYPE    | NUMBER         | Number code of SQL statement type                    |
| PIECE           | NUMBER         | Number given to each text of a split SQL statement   |
| SQL_TEXT        | VARCHAR(65532) | Text of an SQL statement split into 65532-byte texts |

- See also

[V\\$SQLTEXT](#)

[V\\$SQLTEXT\\_WITH\\_NEWLINES](#)

[V\\$SQLTEXT\\_WITH\\_NEWLINES2](#)

## 4.2.140. V\$SQLTEXT\_WITH\_NEWLINES

Displays texts of SQL statements belonging to the cursor, without removing newline characters or blank characters.

- Columns

| Column Name     | Data Type   | Description                            |
|-----------------|-------------|----------------------------------------|
| HASH_VALUE      | NUMBER      | Hash value of SQL statement            |
| PLAN_HASH_VALUE | NUMBER      | Hash value of the physical plan        |
| SQL_ID          | VARCHAR(13) | SQL identifier                         |
| CHILD_NUMBER    | NUMBER      | The number of this child physical plan |

| Column Name  | Data Type   | Description                                         |
|--------------|-------------|-----------------------------------------------------|
| COMMAND_TYPE | NUMBER      | Number code of SQL statement type                   |
| PIECE        | NUMBER      | Number given to each text of a split SQL statement. |
| SQL_TEXT     | VARCHAR(64) | Text of an SQL statement split into 64-byte texts   |

- See also

[V\\$SQLTEXT](#)

[V\\$SQLTEXT2](#)

[V\\$SQLTEXT\\_WITH\\_NEWLINES2](#)

## 4.2.141. V\$SQLTEXT\_WITH\_NEWLINES2

Displays texts of SQL statements belonging to the cursor, without removing newline characters or blank characters.

- Columns

| Column Name     | Data Type      | Description                                          |
|-----------------|----------------|------------------------------------------------------|
| HASH_VALUE      | NUMBER         | Hash value of SQL statement                          |
| PLAN_HASH_VALUE | NUMBER         | Hash value of the physical plan                      |
| SQL_ID          | VARCHAR(13)    | SQL identifier                                       |
| CHILD_NUMBER    | NUMBER         | The number of this child physical plan               |
| COMMAND_TYPE    | NUMBER         | Number code of SQL statement type                    |
| PIECE           | NUMBER         | Number given to each text of a split SQL statement.  |
| SQL_TEXT        | VARCHAR(65532) | Text of an SQL statement split into 65532-byte texts |

- See also

[V\\$SQLTEXT](#)

[V\\$SQLTEXT\\_WITH\\_NEWLINES](#)

[V\\$SQLTEXT2](#)

## 4.2.142. V\$SQL\_BIND\_CAPTURE

Displays the values of the captured bind variables from the current session

- Columns

| Column Name     | Data Type      | Description                                  |
|-----------------|----------------|----------------------------------------------|
| SQL_HASH_VALUE  | NUMBER         | The Hash value of the sql                    |
| PLAN_HASH_VALUE | NUMBER         | The Hash value of the physical plan          |
| SQL_ID          | VARCHAR(13)    | The SQL identifier                           |
| CHILD_NUMBER    | NUMBER         | The number of this child physical plan       |
| USER_ID         | NUMBER         | The user id that performed the SQL           |
| NAME            | VARCHAR(128)   | The name of the bind variable                |
| POSITION        | NUMBER         | The position of the bind variable in the SQL |
| TYPE            | NUMBER         | The type number of the bind variable         |
| TYPE_NAME       | VARCHAR(128)   | The type of the bind variable                |
| PRECISION       | NUMBER         | The precision of the bind variable           |
| SCALE           | NUMBER         | The scale of the bind variable               |
| LENGTH          | NUMBER         | The length of the bind variable              |
| TIME_CAPTURED   | DATE           | The time when the bind variable was captured |
| VALUE           | VARCHAR(65532) | The value of the bind variable               |

## 4.2.143. V\$SQL\_BIND\_CAPTURE\_ALL

Displays the values of the captured bind variables from all sessions

- Columns

| Column Name     | Data Type | Description                         |
|-----------------|-----------|-------------------------------------|
| SQL_HASH_VALUE  | NUMBER    | The Hash value of the sql           |
| PLAN_HASH_VALUE | NUMBER    | The Hash value of the physical plan |

| Column Name   | Data Type      | Description                                  |
|---------------|----------------|----------------------------------------------|
| SQL_ID        | VARCHAR(13)    | The SQL identifier                           |
| CHILD_NUMBER  | NUMBER         | The number of this child physical plan       |
| USER_ID       | NUMBER         | The user id that performed the SQL           |
| NAME          | VARCHAR(128)   | The name of the bind variable                |
| POSITION      | NUMBER         | The position of the bind variable in the SQL |
| TYPE          | NUMBER         | The type number of the bind variable         |
| TYPE_NAME     | VARCHAR(128)   | The type of the bind variable                |
| PRECISION     | NUMBER         | The precision of the bind variable           |
| SCALE         | NUMBER         | The scale of the bind variable               |
| LENGTH        | NUMBER         | The length of the bind variable              |
| TIME_CAPTURED | DATE           | The time when the bind variable was captured |
| VALUE         | VARCHAR(65532) | The value of the bind variable               |

#### 4.2.144. V\$SQL\_EXPC\_INFO

Displays information about expression execution of a physical plan

- Columns

| Column Name    | Data Type      | Description               |
|----------------|----------------|---------------------------|
| SQL_ID         | VARCHAR(13)    | SQL identifier            |
| BIND_PARAM_CNT | NUMBER         | Number of bind parameters |
| BIND_TYPES     | VARCHAR(65532) | Type of bind parameters   |
| ALLOC_SIZE     | NUMBER         | Size of memory usage      |

## 4.2.145. V\$SQL\_MONITOR

Shows the details of SQL statements that are currently being monitored or those that have been monitored through the real time SQL monitoring function.

- Columns

| Column Name          | Data Type   | Description                                                                                                                                                                                                                                                                                                                                                                                                                  |
|----------------------|-------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| STATUS               | VARCHAR(19) | Shows the current execution state of the SQL statement.<br><br>- EXECUTING: The SQL statement is currently being executed.<br><br>- DONE(ERROR): The SQL statement generated a runtime error and was closed.<br><br>- DONE(FIRST N ROWS): Execution completed before all rows were fetched.<br><br>- DONE(ALL ROWS): Execution completed after all rows were fetched.<br><br>- DONE: Completed execution of parallel slaves. |
| SQL_ID               | VARCHAR(13) | SQL identifier                                                                                                                                                                                                                                                                                                                                                                                                               |
| SQL_EXEC_START       | DATE        | The time when the execution started.                                                                                                                                                                                                                                                                                                                                                                                         |
| SQL_EXEC_ID          | NUMBER      | Execution identifier. Identifies multiple, simultaneous SQL executions. To specify a single SQL execution, three columns; SQL_ID, SQL_EXEC_START, and SQL_EXEC_ID must be defined.                                                                                                                                                                                                                                           |
| SID                  | NUMBER      | Session identifier                                                                                                                                                                                                                                                                                                                                                                                                           |
| SESSION_SERIAL#      | NUMBER      | Session serial number                                                                                                                                                                                                                                                                                                                                                                                                        |
| SQL_PLAN_HASH_VALUE  | NUMBER      | Hash value of the execution plan                                                                                                                                                                                                                                                                                                                                                                                             |
| PX_QCSID             | NUMBER      | For parallel execution slaves, this indicates the identifier of the query coordinator session.                                                                                                                                                                                                                                                                                                                               |
| PX_MAXDOP            | NUMBER      | For parallel execution coordinators, this indicates the maximum allocated DDP value.                                                                                                                                                                                                                                                                                                                                         |
| PX_SERVERS_REQUESTED | NUMBER      | For parallel execution coordinators, this indicates the number of requested parallel slaves.                                                                                                                                                                                                                                                                                                                                 |

| Column Name              | Data Type     | Description                                                                                                                      |
|--------------------------|---------------|----------------------------------------------------------------------------------------------------------------------------------|
| PX_SERVERS_ALLO<br>CATED | NUMBER        | For parallel execution coordinators, this indicates the number of allocated parallel slaves.                                     |
| PX_SERVER#               | NUMBER        | For parallel execution slaves, this indicates a logical number of slaves within a single parallel execution set.                 |
| PX_SERVER_SET            | NUMBER        | Whether the parallel execution slave belongs to the producer set or the consumer set. (either 1 or 2)                            |
| PX_SERVER_GROUP          | NUMBER        | For parallel execution slaves, this indicates the logical group number when queries are performed by multiple query coordinators |
| MODULE                   | VARCHAR(48)   | Name of a module specified by dbms_application_in<br>fo.set_module                                                               |
| ACTION                   | VARCHAR(32)   | Name of an action specified by dbms_application_in<br>fo.set_module/action                                                       |
| CLIENT_INFO              | VARCHAR(64)   | Name of client_info specified by dbms_application_in<br>fo.set_client_info                                                       |
| CLIENT_IDENTIFIER        | VARCHAR(64)   | Name of client identifier specified by dbms_ses<br>sion.set_identifier                                                           |
| PROGRAM                  | VARCHAR(48)   | Client program name                                                                                                              |
| WTHR_ID                  | NUMBER        | Working thread identifier                                                                                                        |
| USER#                    | NUMBER        | The current user's identifier                                                                                                    |
| USERNAME                 | VARCHAR(30)   | The current user's name                                                                                                          |
| SQL_TEXT                 | VARCHAR(2000) | Executed SQL statements (A maximum of 2000 charac<br>ters)                                                                       |
| IS_FULL_SQLTEXT          | VARCHAR(1)    | Whether the value saved in the SQL_TEXT column is<br>a full or partial SQL statement.                                            |
| ERROR_NUMBER             | NUMBER        | Error number when a runtime error occurs.                                                                                        |
| ERROR_MESSAGE            | VARCHAR(256)  | Error messages when a runtime error occurs.                                                                                      |
| PSM_ENTRY_OB<br>JECT_ID  | NUMBER        | Object ID of the top-most PSM procedure or function on<br>the stack.                                                             |

| Column Name                     | Data Type      | Description                                                                                               |
|---------------------------------|----------------|-----------------------------------------------------------------------------------------------------------|
| PSM_ENTRY_SUB<br>PROGRAM_ID     | NUMBER         | The subprogram identifier of the procedure or function that was called upon entry into the PSM execution. |
| PSM_OBJECT_ID                   | NUMBER         | The object identifier of the current executed PSM                                                         |
| PSM_SUBPRO<br>GRAM_ID           | NUMBER         | The subprogram identifier of the current executed PSM                                                     |
| CPU_TIME                        | NUMBER         | CPU time (microseconds)                                                                                   |
| ELAPSED_TIME                    | NUMBER         | SQL execution time (microseconds)                                                                         |
| FETCHES                         | NUMBER         | Number of fetches                                                                                         |
| APPLICA<br>TION_WAIT_TIME       | NUMBER         | Total amount of application wait time                                                                     |
| CONCURREN<br>CY_WAIT_TIME       | NUMBER         | Total amount of concurrency wait time                                                                     |
| CLUS<br>TER_WAIT_TIME           | NUMBER         | Total amount of cluster wait time                                                                         |
| USER_IO_WAIT_TIME               | NUMBER         | Total amount of I/O wait time (by user)                                                                   |
| DISK_READS                      | NUMBER         | Disk read count                                                                                           |
| DIRECT_WRITES                   | NUMBER         | Direct disk write cnt without through buffer cache                                                        |
| BUFFER_GETS                     | NUMBER         | Total number of buffers read in Consistent-Read mode                                                      |
| PHYSICAL_READ_RE<br>QUESTS      | NUMBER         | Disk read request count                                                                                   |
| PHYSI<br>CAL_READ_BYTES         | NUMBER         | Total number of bytes read from disks (in bytes)                                                          |
| PHYSI<br>CAL_WRITE_RE<br>QUESTS | NUMBER         | Disk write request count                                                                                  |
| PHYSI<br>CAL_WRITE_BYTES        | NUMBER         | Total number of bytes write to disks (in bytes)                                                           |
| BINDS                           | VARCHAR(32413) | Bind variables used in sql                                                                                |
| BIND_CNT                        | NUMBER         | Total number of bind variables used in sql                                                                |

## 4.2.146. V\$SQL\_PLAN

Displays information on physical plans to execute SQL statements.

- Columns

| Column Name     | Data Type    | Description                                                                                        |
|-----------------|--------------|----------------------------------------------------------------------------------------------------|
| HASH_VALUE      | NUMBER       | Hash value of the SQL statement                                                                    |
| PLAN_HASH_VALUE | NUMBER       | Hash value of the physical plan                                                                    |
| SQL_ID          | VARCHAR(13)  | SQL identifier                                                                                     |
| CHILD_NUMBER    | NUMBER       | The number of this child physical plan                                                             |
| OPERATION       | VARCHAR(128) | Name of operation job                                                                              |
| OBJECT#         | NUMBER       | Identifier of object accessed by the job                                                           |
| OBJECT_OWNER    | VARCHAR(128) | Name of user who has the object                                                                    |
| OBJECT_NAME     | VARCHAR(128) | Object name                                                                                        |
| OBJECT_TYPE     | VARCHAR(20)  | Object type                                                                                        |
| ID              | NUMBER       | Number given to each job of the physical plan                                                      |
| PARENT_ID       | NUMBER       | ID of the next execution step that operates on the output of the current step in the physical plan |
| DEPTH           | NUMBER       | Tree level of the physical plan                                                                    |
| POSITION        | NUMBER       | Positions between all the jobs with the same PARENT_ID                                             |
| SEARCH_COLUMNS  | NUMBER       | The number of keys used in index search                                                            |
| COST            | NUMBER       | Job cost estimated by query optimizer                                                              |
| CPU_COST        | NUMBER       | CPU cost of the operation as estimated by optimizer                                                |
| IO_COST         | NUMBER       | I/O cost of the operation as estimated by optimizer                                                |
| CARDINALITY     | NUMBER       | The number of output results estimated by query optimizer                                          |
| PSTART          | VARCHAR(38)  | In a partitioned table, a start partition for access                                               |

| Column Name       | Data Type     | Description                                                                            |
|-------------------|---------------|----------------------------------------------------------------------------------------|
| PEND              | VARCHAR(38)   | In a partitioned table, an end partition for access                                    |
| OTHERS            | VARCHAR(4000) | Other information specific to the execution step that users may find useful            |
| ACCESS_PREDICATES | VARCHAR(4000) | Predicate information to access indexes or to handle joins                             |
| FILTER_PREDICATES | VARCHAR(4000) | Predicate information to handle filters                                                |
| ADVISED_COLNO     | VARCHAR(4000) | Column number(s) recommended as indexes by the advisor for the plan's TSCAN FULL node. |

## 4.2.147. V\$SQL\_PLAN\_MONITOR

Displays plan level information about each SQL execution registered in V\$SQL\_MONITOR. Each row corresponds to a single plan operation of one SQL execution.

- Columns

| Column Name    | Data Type   | Description                                                                                                                                                                                                                                                                                                                                                                                                                  |
|----------------|-------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| STATUS         | VARCHAR(19) | Shows the current execution state of the SQL statement.<br><br>- EXECUTING: The SQL statement is currently being executed.<br><br>- DONE(ERROR): The SQL statement generated a runtime error and was closed.<br><br>- DONE(FIRST N ROWS): Execution completed before all rows were fetched.<br><br>- DONE(ALL ROWS): Execution completed after all rows were fetched.<br><br>- DONE: Completed execution of parallel slaves. |
| SQL_ID         | VARCHAR(13) | SQL identifier                                                                                                                                                                                                                                                                                                                                                                                                               |
| SQL_EXEC_START | DATE        | The time when the execution started.                                                                                                                                                                                                                                                                                                                                                                                         |
| SQL_EXEC_ID    | NUMBER      | Execution identifier. It is necessary to identify multiple simultaneous SQL executions. To identify a single SQL execution, three columns; SQL_ID, SQL_EXEC_START,                                                                                                                                                                                                                                                           |

| Column Name          | Data Type   | Description                                                                                                               |
|----------------------|-------------|---------------------------------------------------------------------------------------------------------------------------|
|                      |             | and SQL_EXEC_ID must be defined. The three column values are used to match with corresponding rows in the V\$SQL_MONITOR. |
| SID                  | NUMBER      | Session identifier                                                                                                        |
| SQL_PLAN_HASH_VALUE  | NUMBER      | Hash value of the execution plan                                                                                          |
| PLAN_PARENT_ID       | NUMBER      | Identifier of the parent plan operation of this operation                                                                 |
| PLAN_LINE_ID         | NUMBER      | Identifier of the plan operation                                                                                          |
| PLAN_OPERATION       | VARCHAR(30) | Name of the plan operation                                                                                                |
| PLAN_DEPTH           | NUMBER      | Level of the plan operation within the execution plan tree                                                                |
| PLAN_POSITION        | NUMBER      | Logical number among siblings within the plan execution tree                                                              |
| PLAN_COST            | NUMBER      | Cost of the plan operation estimated by the optimizer                                                                     |
| PLAN_CPU_COST        | NUMBER      | CPU cost of the plan operation estimated by the optimizer                                                                 |
| PLAN_IO_COST         | NUMBER      | I/O cost of the plan operation estimated by the optimizer                                                                 |
| PLAN_CARDINALITY     | NUMBER      | The number of rows displayed in a plan operation estimated by the optimizer                                               |
| PLAN_OBJECT_OWNER    | VARCHAR(30) | Owner of the table or index used by this operation                                                                        |
| PLAN_OBJECT_NAME     | VARCHAR(30) | Name of the table or index used by this operation                                                                         |
| PLAN_OBJECT_TYPE     | VARCHAR(30) | Type of object used by this operation                                                                                     |
| PLAN_PARTITION_START | VARCHAR(64) | Start partition when accessing part of a partitioned object                                                               |
| PLAN_PARTITION_STOP  | VARCHAR(64) | End partition when accessing part of a partitioned object                                                                 |
| FIRST_CHANGE_TIME    | DATE        | First time the operation was changed to an upper level operation                                                          |

| Column Name             | Data Type | Description                                                                   |
|-------------------------|-----------|-------------------------------------------------------------------------------|
| LAST_CHANGE_TIME        | DATE      | Last time the operation was changed to an upper level operation               |
| STARTS                  | NUMBER    | Number of times the operation was executed. (Repeated index accesses, etc.)   |
| OUTPUT_ROWS             | NUMBER    | Number of rows produced by this operation                                     |
| PHYSICAL_READ_REQUESTS  | NUMBER    | Disk read count requested by this operation                                   |
| PHYSICAL_READ_BYTES     | NUMBER    | Total number of bytes read from disks by this operation (in bytes)            |
| PHYSICAL_WRITE_REQUESTS | NUMBER    | Disk write count requested by this operation                                  |
| PHYSICAL_WRITE_BYTES    | NUMBER    | Total number of bytes write to disks by this operation (in bytes)             |
| WORKAREA_MEM            | NUMBER    | Size of private memory space occupied by this operation (in bytes)            |
| WORKAREA_MAX_MEM        | NUMBER    | Maximum value for WORKAREA_MEM                                                |
| WORKAREA_TEMPSEG        | NUMBER    | Sum of temporary segment size currently occupied by this operation (in bytes) |
| WORKAREA_MAX_TEMPSEG    | NUMBER    | Maximum value for WORKAREA_TEMPSEG                                            |

## 4.2.148. V\$SQL\_PLAN\_STATISTICS

Displays statistics of results of each physical plan job.

- Columns

| Column Name     | Data Type   | Description                     |
|-----------------|-------------|---------------------------------|
| HASH_VALUE      | NUMBER      | Hash value of the SQL statement |
| PLAN_HASH_VALUE | NUMBER      | Hash value of the physical plan |
| SQL_ID          | VARCHAR(13) | SQL identifier                  |

| Column Name               | Data Type | Description                                                                                              |
|---------------------------|-----------|----------------------------------------------------------------------------------------------------------|
| CHILD_NUMBER              | NUMBER    | The number of this child physical plan                                                                   |
| ID                        | NUMBER    | Number given to each job of the physical plan                                                            |
| EXECUTIONS                | NUMBER    | The number of times that a particular job has been performed                                             |
| LAST_STARTS               | NUMBER    | The number of times that a particular job was started during the last execution                          |
| STARTS                    | NUMBER    | Total number of times that a particular job has been started                                             |
| LAST_OUT<br>PUT_ROWS      | NUMBER    | The number of rows created in a particular job during the last execution                                 |
| OUTPUT_ROWS               | NUMBER    | Total number of rows created in a particular job                                                         |
| LAST_LEFT_IN<br>PUT_ROWS  | NUMBER    | The number of rows which a particular job accepted from a left-side lower job during the last execution  |
| LEFT_INPUT_ROWS           | NUMBER    | Total number of rows which a particular job has accepted from a left-side lower job                      |
| LAST_RIGHT_IN<br>PUT_ROWS | NUMBER    | The number of rows which a particular job accepted from a right-side lower job during the last execution |
| RIGHT_INPUT_ROWS          | NUMBER    | Total number of rows which a particular job has accepted from a right-side lower job                     |
| LAST_CR_BUFFER_GETS       | NUMBER    | The number of buffers which a particular job read in Consistent-Read mode during the last execution      |
| CR_BUFFER_GETS            | NUMBER    | Total number of buffers which a particular job has read in Consistent-Read mode                          |
| LAST_ELAPSED_TIME         | NUMBER    | Time for which a particular job was performed in the last execution (in milliseconds)                    |
| ELAPSED_TIME              | NUMBER    | Total time for which a particular job has been performed (in milliseconds)                               |
| LAST_MEM_USAGE            | NUMBER    | Memory usage for a particular operation during the last execution.                                       |

| Column Name                 | Data Type | Description                                                                                        |
|-----------------------------|-----------|----------------------------------------------------------------------------------------------------|
| MEM_USAGE                   | NUMBER    | Total memory usage for which a particular job has been performed                                   |
| LAST_TEMP_SEGMENT_READ_CNT  | NUMBER    | The number of temporary segment read which a particular job was performed in the last execution    |
| TEMP_SEGMENT_READ_CNT       | NUMBER    | Total number of temporary segment read which a particular job was performed in the last execution  |
| LAST_TEMP_SEGMENT_WRITE_CNT | NUMBER    | The number of temporary segment write which a particular job was performed in the last execution   |
| TEMP_SEGMENT_WRITE_CNT      | NUMBER    | Total number of temporary segment write which a particular job was performed in the last execution |
| LAST_CU_BUFFER_GETS         | NUMBER    | The number of buffers which a particular job read in Current mode during the last execution        |
| CU_BUFFER_GETS              | NUMBER    | Total number of buffers which a particular job has read in Current mode                            |
| LAST_DISK_READS             | NUMBER    | The number of physical disk reads performed by the operation, during the last execution            |
| DISK_READS                  | NUMBER    | The number of physical disk reads performed by the operation, accumulated over the past executions |

## 4.2.149. V\$SQL\_WORKAREA

Display information about work areas used by SQL cursors.

- Columns

| Column Name      | Data Type   | Description                                                 |
|------------------|-------------|-------------------------------------------------------------|
| ADDRESS          | RAW(8)      | Address of the parent statement                             |
| HASH_VALUE       | NUMBER      | Hash value of the parent statement in the library cache     |
| SQL_ID           | VARCHAR(13) | SQL identifier of the parent statement in the library cache |
| WORKAREA_ADDRESS | RAW(8)      | Address of the work area                                    |

| Column Name            | Data Type   | Description                                                                                                                                       |
|------------------------|-------------|---------------------------------------------------------------------------------------------------------------------------------------------------|
| OPERATION_TYPE         | VARCHAR(20) | Operation type that uses the work area                                                                                                            |
| OPERATION_ID           | NUMBER      | Identifier used to distinguish this operation from the execution plan                                                                             |
| POLICY                 | VARCHAR(10) | Policy that determines the work area size (MANUAL or AUTO)                                                                                        |
| ESTIMATED_OPTIMAL_SIZE | NUMBER      | The minimum size (in bytes) for the work area to be executed in memory only                                                                       |
| ESTIMATED_ONEPASS_SIZE | NUMBER      | The minimum size (in bytes) of the work area, to be executed by one-time disk use                                                                 |
| LAST_MEMORY_USED       | NUMBER      | The memory size (in bytes) that was used by the work area during the last execution                                                               |
| LAST_EXECUTION         | VARCHAR(10) | The Process that the work area used during the last execution<br><br>- OPTIMAL: optimal mode execution<br><br>- N PASS: multi pass mode execution |
| LAST_DEGREE            | NUMBER      | Parallelization degree of the work area, during the last execution                                                                                |
| TOTAL_EXECUTIONS       | NUMBER      | Count of the number of times the work area was executed                                                                                           |
| OPTIMAL_EXECUTIONS     | NUMBER      | Count of the number of times that the work area was executed in optimal mode                                                                      |
| ONEPASS_EXECUTIONS     | NUMBER      | Count of the number of times that the work area was executed in one-pass mode                                                                     |
| MULTIPASSES_EXECUTIONS | NUMBER      | Count of the number of times that the work area was executed in multi-pass mode                                                                   |
| ACTIVE_TIME            | NUMBER      | Average time this work area is active                                                                                                             |
| MAX_TEMPSEG_SIZE       | NUMBER      | The maximum size (in bytes) of a temporary segment, that the work area uses                                                                       |

| Column Name       | Data Type | Description                                                                                 |
|-------------------|-----------|---------------------------------------------------------------------------------------------|
| LAST_TEMPSEG_SIZE | NUMBER    | The temporary segment size (in bytes), that was used by the work area in the last execution |

## 4.2.150. V\$SQL\_WORKAREA\_HISTOGRAM

Display the cumulative work area execution statistics

- Columns

| Column Name            | Data Type | Description                                                                                                                                      |
|------------------------|-----------|--------------------------------------------------------------------------------------------------------------------------------------------------|
| LOW_OPTIMAL_SIZE       | NUMBER    | The lower bounding value (in bytes) for the optimal size                                                                                         |
| HIGH_OPTIMAL_SIZE      | NUMBER    | The higher bounding value (in bytes) for the optimal size                                                                                        |
| OPTIMAL_EXECUTIONS     | NUMBER    | Number of work areas with an optimal memory requirement between LOW_OPTIMAL_SIZE and HIGH_OPTIMAL_SIZE which have been executed in optimal mode. |
| ONEPASS_EXECUTIONS     | NUMBER    | The number of work areas that were executed in one-pass mode, within each LOW_OPTIMAL_SIZE and HIGH_OPTIMAL_SIZE bound                           |
| MULTIPASSES_EXECUTIONS | NUMBER    | The number of work areas that were executed in multi-pass mode, within each LOW_OPTIMAL_SIZE and HIGH_OPTIMAL_SIZE bound                         |
| TOTAL_EXECUTIONS       | NUMBER    | The total number of work areas that were executed within each LOW_OPTIMAL_SIZE and HIGH_OPTIMAL_SIZE bound                                       |

## 4.2.151. V\$SSVR\_CLIENT

Displays information of valid connection to the storage server instance.

- Columns

| Column Name | Data Type   | Description         |
|-------------|-------------|---------------------|
| ADDRESS     | VARCHAR(20) | ITHR address number |

| Column Name   | Data Type    | Description              |
|---------------|--------------|--------------------------|
| PORT          | NUMBER       | ITHR port number         |
| NAME          | VARCHAR(128) | DB name                  |
| THREAD_NUMBER | NUMBER       | Connection Thread number |

## 4.2.152. V\$SSVR\_FLASHCACHE

Displays information of flashcaches used by the storage server instance.

- Columns

| Column Name       | Data Type    | Description            |
|-------------------|--------------|------------------------|
| FLASHCACHE_NUMBER | NUMBER       | Flashcache number      |
| NAME              | VARCHAR(32)  | Flashcache name        |
| PATH              | VARCHAR(256) | Flashcache path (full) |
| OS_BYTES          | NUMBER       | Flashcache size        |

## 4.2.153. V\$SSVR\_GRID\_DISK

Displays information of grid disks.

- Columns

| Column Name         | Data Type    | Description         |
|---------------------|--------------|---------------------|
| GRID_DISK_NUMBER    | NUMBER       | Grid disk number    |
| NAME                | VARCHAR(128) | Grid disk name      |
| STORAGE_DISK_NUMBER | NUMBER       | Storage disk number |
| STORAGE_DISK_OFFSET | NUMBER       | Storage disk offset |
| TOTAL_BYTES         | NUMBER       | Grid disk size      |

## 4.2.154. V\$SSVR\_IOLOAD

IO load of storage server

- Columns

| Column Name            | Data Type   | Description                  |
|------------------------|-------------|------------------------------|
| DISK_TYPE              | VARCHAR(32) | !!! No documentation yet !!! |
| DISK_NAME              | VARCHAR(32) | !!! No documentation yet !!! |
| READ_CNT               | NUMBER      | !!! No documentation yet !!! |
| READ_CNT_INTER<br>NAL  | NUMBER      | !!! No documentation yet !!! |
| READ_MB                | NUMBER      | !!! No documentation yet !!! |
| READ_INTERNAL_MB       | NUMBER      | !!! No documentation yet !!! |
| WRITE_CNT              | NUMBER      | !!! No documentation yet !!! |
| WRITE_CNT_INTER<br>NAL | NUMBER      | !!! No documentation yet !!! |
| WRITE_MB               | NUMBER      | !!! No documentation yet !!! |
| WRITE_INTER<br>NAL_MB  | NUMBER      | !!! No documentation yet !!! |

## 4.2.155. V\$SSVR\_IORM\_CATEGORY\_PLAN

Category Plan List used by the storage server instance

- Columns

| Column Name  | Data Type    | Description                  |
|--------------|--------------|------------------------------|
| NAME         | VARCHAR(128) | !!! No documentation yet !!! |
| SHARE_NUMBER | NUMBER       | !!! No documentation yet !!! |
| LIMIT        | NUMBER       | !!! No documentation yet !!! |
| STATUS       | VARCHAR(8)   | !!! No documentation yet !!! |

## 4.2.156. V\$SSVR\_IORM\_DB\_PLAN

DB Plan List used by the storage server instance

- Columns

| Column Name  | Data Type    | Description                  |
|--------------|--------------|------------------------------|
| NAME         | VARCHAR(128) | !!! No documentation yet !!! |
| SHARE_NUMBER | NUMBER       | !!! No documentation yet !!! |
| LIMIT        | NUMBER       | !!! No documentation yet !!! |
| STATUS       | VARCHAR(8)   | !!! No documentation yet !!! |

## 4.2.157. V\$SSVR\_IOSTAT

IO stat per sec of storage server

- Columns

| Column Name               | Data Type   | Description                  |
|---------------------------|-------------|------------------------------|
| DISK_TYPE                 | VARCHAR(32) | !!! No documentation yet !!! |
| DISK_NAME                 | VARCHAR(32) | !!! No documentation yet !!! |
| TPS_READ                  | NUMBER      | !!! No documentation yet !!! |
| TPS_READ_INTER<br>NAL     | NUMBER      | !!! No documentation yet !!! |
| READ_MB_SEC               | NUMBER      | !!! No documentation yet !!! |
| READ_INTER<br>NAL_MB_SEC  | NUMBER      | !!! No documentation yet !!! |
| TPS_WRITE                 | NUMBER      | !!! No documentation yet !!! |
| TPS_WRITE_INTER<br>NAL    | NUMBER      | !!! No documentation yet !!! |
| WRITE_MB_SEC              | NUMBER      | !!! No documentation yet !!! |
| WRITE_INTER<br>NAL_MB_SEC | NUMBER      | !!! No documentation yet !!! |

## 4.2.158. V\$SSVR\_MEMSTAT

Memory stat of storage server

- Columns

| Column Name            | Data Type | Description                  |
|------------------------|-----------|------------------------------|
| TOTAL_PGA_MEMORY_MB    | NUMBER    | !!! No documentation yet !!! |
| FIXED_PGA_MEMORY_MB    | NUMBER    | !!! No documentation yet !!! |
| USED_PGA_MEMORY_MB     | NUMBER    | !!! No documentation yet !!! |
| TOTAL_SHARED_MEMORY_MB | NUMBER    | !!! No documentation yet !!! |
| FIXED_SHARED_MEMORY_MB | NUMBER    | !!! No documentation yet !!! |
| USED_SHARED_MEMORY_MB  | NUMBER    | !!! No documentation yet !!! |

## 4.2.159. V\$SSVR\_SLAB\_STAT

used buffer slab information of storage server

- Columns

| Column Name     | Data Type | Description                 |
|-----------------|-----------|-----------------------------|
| SLAB_SIZE       | NUMBER    | buf_L1_pool slab size       |
| SLAB_GET_CNT    | NUMBER    | buf_L1_pool slab get count  |
| TOTAL_CHUNK_CNT | NUMBER    | buf_L1_pool chunk count     |
| MAX_CHUNK_CNT   | NUMBER    | buf_L1_pool max chunk count |

## 4.2.160. V\$SSVR\_STORAGE\_DISK

Displays information of storage disks used by the storage server instance.

- Columns

| Column Name         | Data Type    | Description              |
|---------------------|--------------|--------------------------|
| STORAGE_DISK_NUMBER | NUMBER       | Storage disk number      |
| NAME                | VARCHAR(128) | Storage disk name        |
| PATH                | VARCHAR(256) | Storage disk path (full) |
| OS_BYTES            | NUMBER       | Storage disk size        |

## 4.2.161. V\$STANDBY

Displays information on the database duplication in the RDBMS Standby Cluster.

- Columns

| Column Name     | Data Type   | Description                                                                       |
|-----------------|-------------|-----------------------------------------------------------------------------------|
| PRIMARY_ADDR    | VARCHAR(80) | Host name and port number of the connection from the primary DB                   |
| STATUS          | VARCHAR(40) | Status of a thread related to standby                                             |
| PRIMARY_TSN     | NUMBER      | TSN updated from primary DB                                                       |
| RECV_SEQ        | NUMBER      | Log sequence number received from primary DB                                      |
| RECV_BLKNO      | NUMBER      | Log block number received from primary DB                                         |
| RECV_TSN        | NUMBER      | TSN of last received log block                                                    |
| RECV_DATE       | DATE        | Time of last received log block                                                   |
| ACK_SEQ         | NUMBER      | Sequence number of the log which the standby DB reported was successfully flushed |
| ACK_BLKNO       | NUMBER      | Block number of the log which the standby DB reported was successfully flushed    |
| RECOVERED_SEQ   | NUMBER      | Log sequence number recovered by SMR of the standby DB                            |
| RECOVERED_BLKNO | NUMBER      | Log block number recovered by SMR of the standby DB                               |

| Column Name    | Data Type | Description                                                         |
|----------------|-----------|---------------------------------------------------------------------|
| RECOVERED_TSN  | NUMBER    | TSN recovered by SMR of the standby DB                              |
| RECOVERED_DATE | DATE      | Time recovered by SMR of the standby DB                             |
| OPEN_SEQ       | NUMBER    | Log sequence number that cannot be used to access the DB as standby |
| OPEN_BLKNO     | NUMBER    | Log block number that cannot be used to access the DB as standby    |

## 4.2.162. V\$STANDBY\_DEST

Displays information on the standby DB in the RDBMS Standby Cluster.

- Columns

| Column Name  | Data Type   | Description                                                                              |
|--------------|-------------|------------------------------------------------------------------------------------------|
| STANDBY_ADDR | VARCHAR(80) | Host name of standby DB and port number                                                  |
| TYPE         | VARCHAR(16) | Standby type<br>- LGWR SYNC<br>- LGWR ASYNC<br>- ARCH ASYNC<br>- LGWR PROXY              |
| THREAD#      | NUMBER      | Redo thread#                                                                             |
| FLAGS        | VARCHAR(32) | Other information of standby<br>- CONNECTED: connected<br>- NOT CONNECTED: not connected |
| SENT_SEQ     | NUMBER      | Sequence number of transmitted log                                                       |
| SENT_BLKNO   | NUMBER      | Block number of transmitted log                                                          |
| ACKED_SEQ    | NUMBER      | Log sequence number reporting that the standby DB was successfully flushed               |

| Column Name | Data Type | Description                                                             |
|-------------|-----------|-------------------------------------------------------------------------|
| ACKED_BLKNO | NUMBER    | Log block number reporting that the standby DB was successfully flushed |

## 4.2.163. V\$STANDBY\_LOG

Displays information of log groups.

- Columns

| Column Name   | Data Type  | Description                                                                                                                                                                                                              |
|---------------|------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| THREAD#       | NUMBER     | Log group number recorded by instances of a server                                                                                                                                                                       |
| GROUP#        | NUMBER     | Log group number                                                                                                                                                                                                         |
| SEQUENCE#     | NUMBER     | Sequence number of the log group                                                                                                                                                                                         |
| BYTES         | NUMBER     | Size of the log group                                                                                                                                                                                                    |
| MEMBERS       | NUMBER     | The number of log member files that belong to the log group                                                                                                                                                              |
| ARCHIVED      | VARCHAR(3) | Whether it was archived or not                                                                                                                                                                                           |
| STATUS        | VARCHAR(8) | Status information of the log group<br>- ACTIVE: File archive needs to be done<br>- UNUSED: The log file has not been used<br>- CURRENT: The log file is currently in use<br>- INACTIVE: File archive has been completed |
| FIRST_CHANGE# | NUMBER     | TSN associated with the first log that belongs to the log group                                                                                                                                                          |
| FIRST_TIME    | DATE       | Time when the first log that belongs to the log group was recorded                                                                                                                                                       |

## 4.2.164. V\$STANDBY\_LOGFILE

Displays information of log member files.

- Columns

| Column Name | Data Type    | Description                                                                                                                      |
|-------------|--------------|----------------------------------------------------------------------------------------------------------------------------------|
| GROUP#      | NUMBER       | Log group number to which the log member file belongs.                                                                           |
| STATUS      | VARCHAR(7)   | Status information of the log member file (Whether or not the log member file was deleted. Currently, it is always set to null.) |
| TYPE        | CHAR(6)      | Always ONLINE                                                                                                                    |
| MEMBER      | VARCHAR(256) | Path name of the log member file                                                                                                 |

## 4.2.165. V\$STANDBY\_STATUS

Displays information on log duplication of the primary DB in the RDBMS Standby Cluster.

- Columns

| Column Name   | Data Type   | Description                                                                 |
|---------------|-------------|-----------------------------------------------------------------------------|
| STANDBY_MODE  | VARCHAR(16) | Data protection level                                                       |
| FLUSHED_SEQ   | NUMBER      | Sequence number of log flushed by the standby DB working in LGWR SYNC mode  |
| FLUSHED_BLKNO | NUMBER      | Block number of the log flushed by the standby DB working in LGWR SYNC mode |
| SYNC          | VARCHAR(8)  | Whether or not LGWR participates in the actions of standby                  |
| WAITING       | VARCHAR(8)  | Event for which LGWR is waiting                                             |
| DETAILS       | VARCHAR(40) | Details of the event for which LGWR is waiting                              |

## 4.2.166. V\$STATNAME

Displays the detailed information on the statistics event.

- Columns

| Column Name | Data Type   | Description                       |
|-------------|-------------|-----------------------------------|
| STAT#       | NUMBER      | Event number to obtain statistics |
| NAME        | VARCHAR(60) | Event name to obtain statistics   |
| CLASS       | NUMBER      | Class number of the event         |
| CLASS_NAME  | VARCHAR(40) | Class name of the event           |

## 4.2.167. V\$SYSSTAT

Displays statistics of a system.

- Columns

| Column Name | Data Type   | Description                                          |
|-------------|-------------|------------------------------------------------------|
| STAT#       | NUMBER      | Event number to obtain statistics                    |
| NAME        | VARCHAR(60) | Event name to obtain statistics                      |
| CLASS       | NUMBER      | Class number of the event                            |
| VALUE       | NUMBER      | Value of the event (Corresponding to the event name) |

## 4.2.168. V\$SYSSTAT\_VALUE

Displays values for getting statistics of a system.

- Columns

| Column Name | Data Type   | Description                       |
|-------------|-------------|-----------------------------------|
| STAT#       | NUMBER      | Event number to obtain statistics |
| NAME        | VARCHAR(60) | Event name to obtain statistics   |
| VALUE       | NUMBER      | Value of the event                |

## 4.2.169. V\$SYSTEM\_EVENT

Displays events performing in the system.

- Columns

| Column Name    | Data Type    | Description                                              |
|----------------|--------------|----------------------------------------------------------|
| IDX            | NUMBER       | Event number                                             |
| NAME           | VARCHAR(100) | Name of the event                                        |
| CLASS          | VARCHAR(40)  | Class of the event                                       |
| DESC           | VARCHAR(100) | Description of the event                                 |
| TOTAL_WAITS    | NUMBER       | Total number of event calls                              |
| TOTAL_TIMEOUTS | NUMBER       | The number of times that a time-out occurred in an event |
| TIME_WAITED    | NUMBER       | The amount of time waited for this event (msec)          |
| AVERAGE_WAIT   | NUMBER       | The average amount of time waited for this event (msec)  |
| MAX_WAIT       | NUMBER       | The maximum amount of time waited for this event (msec)  |

## 4.2.170. V\$SYSTEM\_PARAMETERS

Displays values of initialization parameters currently in effect for the instance.

- Columns

| Column Name | Data Type     | Description                                  |
|-------------|---------------|----------------------------------------------|
| NAME        | VARCHAR(64)   | Name of an initialization parameter          |
| VALUE       | VARCHAR(1024) | Value of an initialization parameter         |
| DFLT_VALUE  | VARCHAR(1024) | Default Value of an initialization parameter |

## 4.2.171. V\$SYS\_TIME\_MODEL

Displays the system-level time statistics.

- Columns

| Column Name | Data Type   | Description                      |
|-------------|-------------|----------------------------------|
| STAT_ID     | NUMBER      | Identifier of a time statistics  |
| STAT_NAME   | VARCHAR(64) | Name of a time statistics        |
| VALUE       | NUMBER      | Amount of time (in microseconds) |

## 4.2.172. V\$TABLESPACE

Displays information on tablespaces.

- Columns

| Column Name  | Data Type    | Description                                   |
|--------------|--------------|-----------------------------------------------|
| TS#          | NUMBER       | A serial number of tablespace                 |
| NAME         | VARCHAR(128) | Tablespace name                               |
| TYPE         | VARCHAR(4)   | Tablespace type<br>- DATA<br>- UNDO<br>- TEMP |
| BIGFILE      | CHAR(2)      | NO in all cases                               |
| FLASHBACK_ON | CHAR(2)      | NO in all cases                               |

## 4.2.173. V\$TACSTAT

Displays statistics of a TAC.

- Columns

| Column Name | Data Type   | Description                       |
|-------------|-------------|-----------------------------------|
| STAT#       | NUMBER      | Event number to obtain statistics |
| NAME        | VARCHAR(60) | Event name to obtain statistics   |

| Column Name | Data Type | Description                                          |
|-------------|-----------|------------------------------------------------------|
| VALUE       | NUMBER    | Value of the event (Corresponding to the event name) |

## 4.2.174. V\$TEMPFILE

Displays temporary files in the database.

- Columns

| Column Name | Data Type    | Description                                           |
|-------------|--------------|-------------------------------------------------------|
| FILE#       | NUMBER       | Temporary file number                                 |
| CREATE_TSN  | NUMBER       | TSN that created the temporary file                   |
| CREATE_DATE | DATE         | Time that the temporary file was created              |
| TS#         | NUMBER       | Number of a tablespace including the temporary file   |
| RFILE#      | NUMBER       | Relative number of the temporary file in a tablespace |
| ENABLED     | CHAR(3)      | Availability (YES in all cases)                       |
| BLOCKS      | NUMBER       | Size of temporary file (the number of blocks)         |
| CREATE_SIZE | NUMBER       | Temporary file size when the file was created         |
| NAME        | VARCHAR(256) | Path name of temporary file                           |

## 4.2.175. V\$TEMPSEG\_OP\_USAGE

Displays temporary segments usage in the server.

- Columns

| Column Name | Data Type   | Description                                              |
|-------------|-------------|----------------------------------------------------------|
| SEGTYPE     | VARCHAR(32) | Temporary segment type<br>- SORT<br>- HASH<br>- ANALYTIC |

| Column Name        | Data Type | Description                                                 |
|--------------------|-----------|-------------------------------------------------------------|
|                    |           | - INDEX<br>- DATA<br>- LOBDATA                              |
| CURRENT_EXTENTS    | NUMBER    | The number of extents currently allocated                   |
| PEEK_EXTENTS       | NUMBER    | The peak number of extents allocated after instance boot    |
| MAX_ONE_OP_EXTENTS | NUMBER    | The maximum number of extents allocated in single operation |

## 4.2.176. V\$TEMPSEG\_USAGE

Displays temporary segments in the server.

- Columns

| Column Name | Data Type    | Description                                                                                |
|-------------|--------------|--------------------------------------------------------------------------------------------|
| USERNAME    | VARCHAR(32)  | Name of user using the temporary segment                                                   |
| SESSION_NUM | NUMBER       | Number of sessions using the temporary segment                                             |
| SQL_ID      | NUMBER       | ID of PP cache used by the temporary segment                                               |
| TABLESPACE  | VARCHAR(128) | Name of tablespace containing the temporary segment                                        |
| SEGTYPE     | VARCHAR(32)  | Temporary segment type<br>- SORT<br>- HASH<br>- ANALYTIC<br>- INDEX<br>- DATA<br>- LOBDATA |
| SEGFILE#    | NUMBER       | Absolute number of a file containing the header block of the temporary segment             |

| Column Name | Data Type | Description                                                                    |
|-------------|-----------|--------------------------------------------------------------------------------|
| SEGBLK#     | NUMBER    | Block number of the header block of the temporary segment                      |
| EXTENTS     | NUMBER    | The number of extents in the temporary segment                                 |
| BLOCKS      | NUMBER    | The number of blocks in the temporary segment                                  |
| SEGRFNO#    | NUMBER    | Relative number of a file containing the header block of the temporary segment |

## 4.2.177. V\$TEMPSTAT

Displays statistics of file I/O.

- Columns

| Column Name    | Data Type | Description                                   |
|----------------|-----------|-----------------------------------------------|
| FILE#          | NUMBER    | File number                                   |
| PHYRDS         | NUMBER    | The number of reading times                   |
| PHYWRTS        | NUMBER    | The number of writing times                   |
| PHYBLKRD       | NUMBER    | The number of read blocks                     |
| PHYBLKWRT      | NUMBER    | The number of written blocks                  |
| SINGLEBLKRDS   | NUMBER    | The number of times of reading a single block |
| READTIM        | NUMBER    | Reading time (msec)                           |
| WRITETIM       | NUMBER    | Writing time (msec)                           |
| SINGLEBLKRDTIM | NUMBER    | Time for which a single block was read (msec) |
| AVGIOTIM       | NUMBER    | Average I/O time (msec)                       |
| LSTIOTIM       | NUMBER    | Last I/O time (msec)                          |
| MINIOTIM       | NUMBER    | Shortest I/O time (msec)                      |
| MAXIORTM       | NUMBER    | Longest reading time (msec)                   |

| Column Name | Data Type | Description                 |
|-------------|-----------|-----------------------------|
| MAXIOWTM    | NUMBER    | Longest writing time (msec) |

## 4.2.178. V\$THRINFO

Information on thr\_info

- Columns

| Column Name      | Data Type   | Description                            |
|------------------|-------------|----------------------------------------|
| THR_ID           | NUMBER      | Index of thread                        |
| THR_TYPE         | VARCHAR(15) | Type of thread                         |
| THR_NAME         | VARCHAR(8)  | Name of thread                         |
| THR_STATUS       | VARCHAR(25) | Status of thread                       |
| PID              | NUMBER      | Process id                             |
| PROC_IDX         | NUMBER      | Process index                          |
| PHY_TID          | NUMBER      | Thread index inside a process          |
| PTHREAD_ID       | NUMBER      | Pthread id (for pthread_kill etc.)     |
| BITQ_DATA0       | NUMBER      | Bitmask data                           |
| BITQ_DATA1       | NUMBER      | Bitmask data                           |
| BITQ_DATA2       | NUMBER      | Bitmask data                           |
| BITQ_SIGFLAG     | NUMBER      | Used by async. signal handler          |
| FLAG             | NUMBER      | Thread flag (for stop / shutdown etc.) |
| SPINLOCK_ACQ_CNT | NUMBER      | Count of acquired spinlock             |
| SPIN_WAIT        | VARCHAR(18) | Spinlock address                       |
| TIMEOUT          | NUMBER      | Bitq_read timeout                      |

## 4.2.179. V\$THR\_ACT\_INFO

Displays information on activities of the threads.

- Columns

| Column Name       | Data Type   | Description                                                         |
|-------------------|-------------|---------------------------------------------------------------------|
| THR_ID            | NUMBER      | Thread index                                                        |
| THR_TYPE          | VARCHAR(15) | Thread type                                                         |
| THR_NAME          | VARCHAR(8)  | Thread name                                                         |
| THR_STATUS        | VARCHAR(25) | Thread status                                                       |
| PID               | NUMBER      | Process id                                                          |
| PROC_IDX          | NUMBER      | Process index                                                       |
| PHY_TID           | NUMBER      | Thread index within the process                                     |
| PTHREAD_ID        | NUMBER      | Pthread id (for pthread_kill etc.)                                  |
| CUR_ACT_CNT       | NUMBER      | Current activity counter                                            |
| NO_ACT_TIME       | NUMBER      | Elapsed time without explicit thread activity                       |
| NO_ACT_TIME_LIMIT | NUMBER      | Limit of the elapsed time without explicit thread activity          |
| CUR_ACT_INTERVAL  | NUMBER      | Limit of the elapsed time without explicit thread activity(current) |
| ACT_REPORT        | VARCHAR(3)  | Whether or not to report thread activity (on/off)                   |

## 4.2.180. V\$THR\_EVENT

Displays events that are performed in each thread.

- Columns

| Column Name | Data Type | Description        |
|-------------|-----------|--------------------|
| TID         | NUMBER    | Thread ID          |
| IDX         | NUMBER    | Index of the event |

| Column Name    | Data Type    | Description                                              |
|----------------|--------------|----------------------------------------------------------|
| NAME           | VARCHAR(100) | Name of the event                                        |
| CLASS          | VARCHAR(40)  | Class of the event                                       |
| DESC           | VARCHAR(100) | Description of the event                                 |
| TOTAL_WAITS    | NUMBER       | Total number of event calls                              |
| TOTAL_TIMEOUTS | NUMBER       | The number of times that a time-out occurred in an event |
| TIME_WAITED    | NUMBER       | The amount of time waited for this event (msec)          |
| AVERAGE_WAIT   | NUMBER       | The average amount of time waited for this event (msec)  |
| MAX_WAIT       | NUMBER       | The maximum amount of time waited for this event (msec)  |

## 4.2.181. V\$TIMER

Displays information on the system timer.

- Columns

| Column Name | Data Type | Description                           |
|-------------|-----------|---------------------------------------|
| TICK        | NUMBER    | Elapsed time since the system started |

## 4.2.182. V\$TIMEZONE\_NAMES

Displays timezone names.

- Columns

| Column Name | Data Type   | Description                  |
|-------------|-------------|------------------------------|
| TZNAME      | VARCHAR(40) | Name of the timezone         |
| TZABBREV    | VARCHAR(8)  | Abbreviation of the timezone |

## 4.2.183. V\$TRANSACTION

Displays ongoing transactions.

- Columns

| Column Name | Data Type | Description                                                              |
|-------------|-----------|--------------------------------------------------------------------------|
| SESS_ID     | NUMBER    | ID of session operation transactions                                     |
| USN         | NUMBER    | Number of Undo segment assigned in the transaction                       |
| SLOT        | NUMBER    | Slot number of transaction in use                                        |
| WRAP        | NUMBER    | The number of times of reusing a transaction slot                        |
| UEA_FILE    | NUMBER    | Number of a file containing Undo which a transaction used most recently  |
| UEA_BLK     | NUMBER    | Number of a block containing Undo which a transaction used most recently |
| UEA_SEQ     | NUMBER    | A serial number of Undo which a transaction used most recently           |
| UEA_ROW     | NUMBER    | A row number of Undo which a transaction used most recently              |
| STATE       | NUMBER    | Status of transaction                                                    |
| START_TIME  | DATE      | Transaction start time                                                   |
| START_BASE  | NUMBER    | Child value of transaction start TSN                                     |
| START_WRAP  | NUMBER    | Parent value of transaction start TSN                                    |
| START_UEXT  | NUMBER    | Extent number of Undo space when the transaction starts                  |
| START_FILE  | NUMBER    | File number of Undo space when the transaction starts                    |
| START_BLK   | NUMBER    | Block number of Undo space when the transaction starts                   |
| START_SEQ   | NUMBER    | A serial number of Undo space when the transaction starts                |
| START_ROW   | NUMBER    | Row number of Undo space when the transaction starts                     |

| Column Name | Data Type    | Description                                                                        |
|-------------|--------------|------------------------------------------------------------------------------------|
| START_INCNO | NUMBER       | Incarnation number of Undo space when the transaction starts                       |
| FLAG        | NUMBER       | Flags                                                                              |
| SPACE       | NUMBER       | Whether or not the transaction is a space management transaction                   |
| NAME        | VARCHAR(256) | Transaction name                                                                   |
| PTX_USN     | NUMBER       | Number of Undo segment assigned to the parent transaction                          |
| PTX_SLOT    | NUMBER       | Number of transaction slot in use by the parent transaction                        |
| PTX_WRAP    | NUMBER       | The number of times of reusing a transaction slot in use by the parent transaction |
| LOG_IO      | NUMBER       | The number of logical io by transactions                                           |
| PHY_IO      | NUMBER       | The number of physical io used by transactions                                     |
| CR_GET      | NUMBER       | The number of Consistent block gets by transactions                                |
| RECOVERY    | VARCHAR(8)   | Recovery status<br>- RECOVERY<br>- SUSPEND                                         |
| USED_BLK    | NUMBER       | The number of Undo blocks used by transactions                                     |
| USED_REC    | NUMBER       | The number of Undo records used by transactions                                    |
| EXP_RB_TIME | NUMBER       | Expectation value of rollback time                                                 |
| TX_TYPE     | VARCHAR(20)  | Transaction type<br>- LOCAL<br>- XA<br>- XA_RECOVERY<br>- DBLINK                   |

## 4.2.184. V\$TSN\_TIME

Displays the mapping table of TSN and time.

- Columns

| Column Name | Data Type    | Description |
|-------------|--------------|-------------|
| TSN         | NUMBER       | TSN         |
| TIME        | TIMESTAMP(6) | TIMESTAMP   |

## 4.2.185. V\$UNDOSTAT

Displays statistics of Undo storage.

- Columns

| Column Name      | Data Type | Description                                               |
|------------------|-----------|-----------------------------------------------------------|
| BEGIN_TIME       | DATE      | Start time of statistics                                  |
| END_TIME         | DATE      | End time of statistics                                    |
| UNDO_TSNO        | NUMBER    | Tablespace number                                         |
| UNDOBLKS         | NUMBER    | The number of used blocks                                 |
| TXCOUNT          | NUMBER    | Total number of executed transactions                     |
| MAXCONCURRENCY   | NUMBER    | The largest number of concurrently executed transactions  |
| SNAP_TOO_OLD_CNT | NUMBER    | The number of errors caused by failing to read old data   |
| NOSPACEERRCNT    | NUMBER    | The number of errors caused by lack of Undo storage space |
| ACTIVEBLKS       | NUMBER    | The number of blocks in use                               |
| UNEXPIREDBLKS    | NUMBER    | The number of non reusable blocks                         |
| EXPIREDBLKS      | NUMBER    | The number of reusable blocks                             |

## 4.2.186. V\$UNDO\_FREE\_SPACE

Displays free space in block count for each UNDO tablespace.

- Columns

| Column Name  | Data Type    | Description                                       |
|--------------|--------------|---------------------------------------------------|
| TS_ID        | NUMBER       | ID of the UNDO tablespace                         |
| TS_NAME      | VARCHAR(128) | Name of the UNDO tablespace                       |
| TOTAL_BLOCKS | NUMBER       | The total number of blocks in the UNDO tablespace |
| FREE_BLOCKS  | NUMBER       | The number of free blocks in the UNDO tablespace  |
| TOTAL_SIZE   | NUMBER       | The total size of the UNDO tablespace in bytes    |
| FREE_SIZE    | NUMBER       | The free size of the UNDO tablespace in bytes     |

## 4.2.187. V\$USGMT\_TXENTRY

Displays undo segment transaction entry information.

- Columns

| Column Name | Data Type   | Description                                                              |
|-------------|-------------|--------------------------------------------------------------------------|
| USN         | NUMBER      | Undo segment number                                                      |
| SLOT        | NUMBER      | Slot number of transaction entry                                         |
| WRAP        | NUMBER      | The number of times of reusing a transaction slot                        |
| STATUS      | VARCHAR(32) | Status information                                                       |
| FLAGS       | VARCHAR(32) | Other information of transaction entry                                   |
| UEA_FILE    | NUMBER      | Number of a file containing undo which a transaction used most recently  |
| UEA_BLOCK   | NUMBER      | Number of a block containing undo which a transaction used most recently |
| UEA_DBA     | NUMBER      | Starting DBA of a transaction                                            |
| UBLKCNT     | NUMBER      | Used undo block count                                                    |

## 4.2.188. V\$VERSION

Displays RDBMS version information.

- Columns

| Column Name | Data Type    | Description                       |
|-------------|--------------|-----------------------------------|
| NAME        | VARCHAR(32)  | Name of the version information   |
| VALUE       | VARCHAR(32)  | Value of the version information  |
| VALUE       | VARHCAR(256) | Banner of the version information |

## 4.2.189. V\$WAITER\_SESSION

Displays information on the sessions that wait for a lock acquisition

- Columns

| Column Name | Data Type   | Description                                                     |
|-------------|-------------|-----------------------------------------------------------------|
| WAIT_SID    | NUMBER      | Session ID that waits for a lock acquisition                    |
| TYPE        | VARCHAR(25) | Type of the lock                                                |
| ID1         | NUMBER      | First ID of the lock                                            |
| ID2         | NUMBER      | Second ID of the lock                                           |
| MODE_REQ    | NUMBER      | Lock mode by which is requested                                 |
| HOLD_SID    | NUMBER      | Session ID that owns the requested lock                         |
| MODE_HELD   | NUMBER      | Lock mode by which is owned                                     |
| WAIT_TIME   | NUMBER      | The time (in seconds) that the session waited to acquire a lock |

## 4.2.190. V\$WAITSTAT

Displays information on block wait events for each block type and class

- Columns

| Column Name | Data Type   | Description                                     |
|-------------|-------------|-------------------------------------------------|
| CLASS       | VARCHAR(20) | Block class (Block type)                        |
| COUNT       | NUMBER      | The number of times waiting for the block class |
| TIME        | NUMBER      | Total time spent waiting for the block class    |

## 4.2.191. V\$WAITUSE\_STAT

Displays information on block wait events

- Columns

| Column Name | Data Type   | Description                                                   |
|-------------|-------------|---------------------------------------------------------------|
| CLASS       | VARCHAR(20) | Block class (Block type)                                      |
| PIN_USE     | VARCHAR(15) | The pin usage of the block pin owners                         |
| WAIT_USE    | VARCHAR(15) | The pin usage of the block pin waiters                        |
| COUNT       | NUMBER      | The number of times waiting for the block class and pin usage |
| TIME        | NUMBER      | Total time spent waiting for the block class and pin usage    |

## 4.2.192. V\$WAITUSE\_THR\_STAT

Displays information on block wait events for each block type, class, and pin usage

- Columns

| Column Name | Data Type   | Description                                                   |
|-------------|-------------|---------------------------------------------------------------|
| CLASS       | VARCHAR(20) | Block class (Block type)                                      |
| PIN_USE     | VARCHAR(15) | The pin usage of the buffer pin owners                        |
| WAIT_USE    | VARCHAR(15) | The pin usage of the buffer pin waiters                       |
| COUNT       | NUMBER      | The number of times waiting for the block class and pin usage |

| Column Name | Data Type | Description                                                    |
|-------------|-----------|----------------------------------------------------------------|
| TIME        | NUMBER    | Total time spent waiting for the block class and pin us<br>age |

# 제5장 Database Limits

## 5.1. Overview

Tibero 데이터베이스 기능 및 개체와 관련된 값의 제한을 기술하고 있다. 일반적으로 데이터베이스에는 초과할 수 없는 하드 코딩된 제한이 데이터베이스의 여러 수준에 존재합니다. 또한 이 값은 운영 체제(OS)에 따라 추가로 제한될 수 있습니다.

## 5.2. List of Datatype Limits

이 섹션은 Tibero의 데이터 타입과 관련된 값의 제한을 기술한다.

| 데이터 타입    | 제한                                                                      |
|-----------|-------------------------------------------------------------------------|
| BLOB      | 4 GB                                                                    |
| CHAR      | 2,000 bytes 또는 2,000 글자                                                 |
| CLOB      | 4 GB                                                                    |
| JSON      | 4 GB                                                                    |
| LONG      | 2 GB                                                                    |
| NCHAR     | 2,000 글자                                                                |
| NCLOB     | 4 GB                                                                    |
| NUMBER    | 최대값 999...(38개의 9) <sup>125</sup><br>최소값 -999...(38개의 9) <sup>125</sup> |
| NVARCHAR2 | 65,532 bytes                                                            |
| RAW       | 2,000 bytes                                                             |
| VARCHAR2  | 65,532 bytes                                                            |

## 5.3. List of Physical Database Limits

이 섹션은 Tibero의 물리적인 데이터베이스와 관련된 값의 제한을 기술한다.

| 항목                | 제한 종류 | 제한값        |
|-------------------|-------|------------|
| Database Block 크기 | 최소값   | 2048 bytes |

| 항목                                  | 제한 종류                          | 제한값                                                                                                                   |
|-------------------------------------|--------------------------------|-----------------------------------------------------------------------------------------------------------------------|
| Database Block 크기                   | 최댓값                            | 32 KB                                                                                                                 |
| Database Blocks                     | 세그먼트의 초기 extent에서의 최솟값         | 16                                                                                                                    |
| Database Blocks                     | datafile 당 최댓값                 | $2^{22}$                                                                                                              |
| Controlfiles                        | 개수                             | 최소 1 개; 최대 $2^{31}$ 개까지 다중화 가능                                                                                        |
| Controlfiles                        | 블록 크기                          | 16 KB 고정                                                                                                              |
| Controlfiles                        | 파일 당 크기                        | CREATE DATABASE/CONTROLFILE 구문에서 기입하는 섹션별 제한 개수에 의존; 최대 약 478 MB                                                      |
| Database files                      | 테이블스페이스 당 최대 개수                | 1024; DIRECT DFID(기본값 128)를 모두 사용하면 896개                                                                              |
| Database files                      | 데이터베이스 당 최대 개수                 | 65533; MAXLOGFILES, MAXBACKUPSET 등 다른 값에 의해서 제한될 수 있음                                                                 |
| Database extents                    | dictionary 관리 tablespace 당 최댓값 | datafile 전체를 extent로 할당할 때 최대가 되므로 $1024 * 2^{22} * \text{physical block size} = 2^{32} * \text{physical block size}$ |
| Database extents                    | 로컬로 관리되는 (균일한) tablespace당 최대  | datafile 전체를 extent로 할당할 때 최대가 되므로 $1024 * 2^{22} * \text{physical block size} = 2^{32} * \text{physical block size}$ |
| MAXEXTENTS                          | 기본값                            | STORAGE 구문에서 기입하는 MAXEXTENT에 의존; 기입되지 않았다면 무제한                                                                        |
| MAXEXTENTS                          | 최댓값                            | 무제한                                                                                                                   |
| Redo Log 파일 개수                      | 로그 그룹 최소 개수                    | 2                                                                                                                     |
| Redo log files                      | 로그 그룹 최대 개수                    | CREATE DATABASE/CONTROLFILE 구문에서 기입하는 MAXLOGFILES에 의존; 최댓값 255                                                        |
| Redo log files                      | 로그 그룹 당 멤버 최대 개수               | 8                                                                                                                     |
| Redo log files                      | 최소 크기                          | 512 KB                                                                                                                |
| Redo log files                      | 최대 크기                          | 2 TB                                                                                                                  |
| Tablespaces                         | 데이터베이스 당 최대 개수                 | $2^{16} - 9$                                                                                                          |
| Bigfile Tablespaces                 | 블록 개수                          | $2^{32}$                                                                                                              |
| Smallfile (traditional) Tablespaces | 블록 개수                          | $2^{22}$                                                                                                              |

| 항목                   | 제한 종류 | 제한값                                          |
|----------------------|-------|----------------------------------------------|
| External Tables file | 최대 크기 | 운영 체제에 따라 달라질 수 있으며 외부 테이블은 여러 파일로 구성될 수 있음. |

## 5.4. List of Logical Database Limits

이 섹션은 Tibero의 논리적 데이터베이스와 관련된 값의 제한을 기술한다.

| 항목                                  | 제한 종류                                  | 제한값          |
|-------------------------------------|----------------------------------------|--------------|
| Columns                             | 테이블 당 최대 값                             | 1500         |
| Columns                             | Maximum per index (or clustered index) | 32           |
| Columns                             | Maximum per bitmapped index            | 32           |
| Constraints                         | 컬럼 당 최대 개수                             | 무제한          |
| Constraints                         | 데이터베이스 당 최대 개수                         | 무제한          |
| Database users                      | 데이터베이스 당 최대 개수                         | 무제한          |
| Dictionary-managed database objects | 데이터베이스 당 최대 개수                         | 무제한          |
| Indexes                             | 테이블 당 최대 개수                            | 무제한          |
| Partitions                          | 파티셔닝된 키에서의 최대 컬럼 수                     | 32           |
| Partitions                          | 테이블 또는 인덱스에 생성할 수 있는 파티션 최대 개수         | $2^{31} - 1$ |
| Rows                                | 테이블당 최대 개수                             | 무제한          |
| Subpartitions                       | 복합 파티션 테이블에 생성할 수 있는 서브 파티션 최대 개수      | $2^{31} - 1$ |
| Subqueries                          | SQL 문장 하나에서 사용 가능한 최대 개수               | 무제한          |
| Tables                              | 데이터베이스 당 최대 개수                         | 무제한          |
| Trigger Cascade Limit               | 최대 개수                                  | 무제한          |
| Users and Roles                     | 최대 개수                                  | 무제한          |

## 5.5. List of Process and Runtime Limits

이 섹션은 Tibero의 프로세스와 런타임과 관련된 값의 제한을 기술한다.

| 항목                        | 제한 종류                     | 제한값                                                                 |
|---------------------------|---------------------------|---------------------------------------------------------------------|
| Database 당 instance 개수    | TAC에서의 최댓값                | 1024                                                                |
| Locks                     | Row-level                 | 무제한                                                                 |
| Locks                     | Cluster Wait Lock Service | Shared Memory 크기 내에서 무제한                                            |
| Shared Memory size        | 최댓값                       | 운영 체제에 따라 다름. 일반적으로 32비트 운영 체제의 경우 4 GB, 64비트 운영 체제의 경우 4 GB 이상     |
| Job Sessions              | 인스턴스 당 최댓값                | Initialization Parameter (MAX_SESSION_COUNT)로 설정된 제한 내에서 무제한        |
| Sessions                  | 인스턴스 당 최댓값                | $2^{31} - 1$ . Initialization Parameter(MAX_SESSION_COUNT)로 설정      |
| Parallel Execution Slaves | 인스턴스 당 최댓값                | $10^8$ . Initialization Parameter(WTHR_PER_PEP * PEP_PROC_CNT)들로 설정 |

# Appendix A. 초기화 파라미터의 설정 예제

본 장에서는 Tibero를 설치하고 나서 자동으로 설정되는 초기화 파라미터를 설명한다. 초기화 파라미터에 대한 자세한 내용은 “제1장 Introduction to Initialization Parameters”를 참고한다.

다음은 초기화 파라미터를 설정한 예이다.

---

## 참고

TOTAL\_SHM\_SIZE 및 MEMORY\_TARGET 설정 시, Megabyte, Gigabyte 단위로 설정하는 경우 M, G 단위로 입력하며 byte의 경우 숫자만을 입력해준다.

---

---

## 참고

CONTROL\_FILES 와 같이 초기화 파라미터에 절대 경로를 입력하는 경우, WINDOWS 플랫폼에서는 경로 구분자로 '/'(슬래시) 또는 '\\'(백슬래시 두 개)를 사용해야한다.

---

<\$TB\_SID.tip>

```
DB_NAME=$TB_SID
LISTENER_PORT=8629
CONTROL_FILES="$TB_HOME/database/tiberoc1.ctl"

MAX_SESSION_COUNT=20

TOTAL_SHM_SIZE=1G
MEMORY_TARGET=2G
```

다음은 위 예에서 설정한 각 파라미터에 대한 설명이다.

| 파라미터              | 설명                                                                                                          |
|-------------------|-------------------------------------------------------------------------------------------------------------|
| DB_NAME           | 데이터베이스 이름을 설정한다. (기본값: tiberoc)                                                                             |
| LISTENER_PORT     | 리스너가 사용할 포트 번호를 설정한다. (기본값: 8629)<br><br>SHMKEY와 더불어 한 서버에서 두 개 이상의 데이터베이스 인스턴스를 수행할 때에는 반드시 다르게 설정해 줘야 한다. |
| CONTROL_FILES     | 컨트롤 파일이 존재하는 위치로서 절대 경로로 설정한다.                                                                              |
| MAX_SESSION_COUNT | 사용자들이 접속해서 동시에 사용할 수 있는 세션의 최대 개수를 설정한다.                                                                    |

| 파라미터                           | 설명                                                                                                                                                                                                                  |
|--------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <a href="#">TOTAL_SHM_SIZE</a> | <p>데이터베이스의 인스턴스 내에서 사용할 전체 공유 메모리의 크기를 설정한다.</p> <p>전체 공유 메모리의 크기는 <math>DB\_CACHE\_SIZE + LOG\_BUFFER + 20M(\text{초기 기동 시 할당 영역}) + (WTHR\_PROC\_CNT * WTHR\_PER\_PROC) * 1M</math>를 계산한 값보다 좀 더 초과하여 설정하면 된다.</p> |
| <a href="#">MEMORY_TARGET</a>  | <p>데이터베이스의 인스턴스 내에서 사용할 전체 메모리의 크기를 설정한다.</p> <p>이는 전체 공유 메모리와 프로세스 각각 사용할 메모리 전체를 합친 크기로 <math>TOTAL\_SHM\_SIZE + (\text{프로세스에서 사용할 메모리 전체})</math>를 계산한 값으로 설정하면 된다.</p>                                          |

## 참고

초기화 파라미터는 시스템의 사용 환경에 따라 사용하는 파라미터가 다를 수 있으며, 추가로 설정할 수 있다.

# Appendix B. COMMAND

본 부록에서는 COMMAND의 종류에 대하여 설명한다.

COMMAND\_TYPES TABLE을 조회하면 같은 내용을 볼 수 있다.

| type_no | name            |
|---------|-----------------|
| 0       | NONE            |
| 1       | SELECT          |
| 2       | INSERT          |
| 3       | UPDATE          |
| 4       | DELETE          |
| 5       | MERGE           |
| 6       | CALL            |
| 7       | CREATE DATABASE |
| 8       | ALTER DATABASE  |
| 9       | CREATE TABLE    |
| 10      | ALTER TABLE     |
| 11      | DROP TABLE      |
| 12      | CREATE INDEX    |
| 13      | ALTER INDEX     |
| 14      | DROP INDEX      |
| 15      | CREATE VIEW     |
| 16      | ALTER VIEW      |
| 17      | DROP VIEW       |
| 18      | CREATE MVIEW    |
| 19      | ALTER MVIEW     |
| 20      | DROP MVIEW      |
| 21      | CREATE MLOG     |
| 22      | ALTER MLOG      |
| 23      | DROP MLOG       |
| 24      | CREATE USER     |

| type_no | name                |
|---------|---------------------|
| 25      | ALTER USER          |
| 26      | DROP USER           |
| 27      | ALTER SYSTEM        |
| 28      | ALTER SESSION       |
| 29      | SET ROLE            |
| 30      | EXPLAIN             |
| 31      | CREATE TABLESPACE   |
| 32      | ALTER TABLESPACE    |
| 33      | DROP TABLESPACE     |
| 34      | CREATE SEQUENCE     |
| 35      | ALTER SEQUENCE      |
| 36      | DROP SEQUENCE       |
| 37      | CREATE SYNONYM      |
| 38      | ALTER SYNONYM       |
| 39      | DROP SYNONYM        |
| 40      | CREATE ROLE         |
| 41      | ALTER ROLE          |
| 42      | DROP ROLE           |
| 43      | CREATE PROCEDURE    |
| 44      | ALTER PROCEDURE     |
| 45      | DROP PROCEDURE      |
| 46      | CREATE FUNCTION     |
| 47      | ALTER FUNCTION      |
| 48      | DROP FUNCTION       |
| 49      | CREATE PACKAGE      |
| 50      | ALTER PACKAGE       |
| 51      | DROP PACKAGE        |
| 52      | CREATE PACKAGE BODY |
| 53      | DROP PACKAGE BODY   |
| 54      | CREATE TRIGGER      |
| 55      | ALTER TRIGGER       |
| 56      | DROP TRIGGER        |

| <b>type_no</b> | <b>name</b>                 |
|----------------|-----------------------------|
| 57             | CREATE LIBRARY              |
| 58             | DROP LIBRARY                |
| 59             | CREATE JAVA                 |
| 60             | DROP JAVA                   |
| 61             | CREATE DATABASE LINK        |
| 62             | CREATE PUBLIC DATABASE LINK |
| 63             | CREATE ANY DATABASE LINK    |
| 64             | DROP DATABASE LINK          |
| 65             | DROP PUBLIC DATABASE LINK   |
| 66             | DROP ANY DATABASE LINK      |
| 67             | CREATE OUTLINE              |
| 68             | DROP OUTLINE                |
| 69             | CREATE DIRECTORY            |
| 70             | DROP DIRECTORY              |
| 71             | ALTER TYPE                  |
| 72             | CREATE TYPE                 |
| 73             | DROP TYPE                   |
| 74             | CREATE TYPE BODY            |
| 75             | DROP TYPE BODY              |
| 76             | GRANT OBJPRIV               |
| 77             | REVOKE OBJPRIV              |
| 78             | GRANT SYSPRIV               |
| 79             | REVOKE SYSPRIV              |
| 80             | AUDIT                       |
| 81             | NOAUDIT                     |
| 82             | RENAME                      |
| 83             | COMMENT                     |
| 84             | TRUNCATE                    |
| 85             | PURGE                       |
| 86             | FLASHBACK TABLE             |
| 87             | CREATE CONTROL FILE         |
| 88             | ALTER ROLLBACK SEG          |

| <b>type_no</b> | <b>name</b>              |
|----------------|--------------------------|
| 89             | CREATE PROFILE           |
| 90             | ALTER PROFILE            |
| 91             | DROP PROFILE             |
| 92             | COMMIT                   |
| 93             | ROLLBACK                 |
| 94             | SAVEPOINT                |
| 95             | SET TRANSACTION          |
| 96             | LOCK                     |
| 97             | CREATE DISKSPACE         |
| 98             | ALTER DISKSPACE          |
| 99             | DROP DISKSPACE           |
| 100            | MODIFY DISKSPACE         |
| 101            | CREATE BTIP              |
| 102            | CREATE TIP               |
| 103            | CREATE CONTEXT           |
| 104            | DROP CONTEXT             |
| 105            | CREATE SSVR              |
| 106            | CREATE SDISK             |
| 107            | DROP SDISK               |
| 108            | CREATE GDISK             |
| 109            | DROP GDISK               |
| 110            | CREATE SFC               |
| 111            | DROP SFC                 |
| 112            | ALTER SFC                |
| 113            | ALTER IORMPLAN           |
| 114            | CREATE CONTROL FILE SSVR |
| 115            | ALTER RMGR               |
| 116            | DOWNGRADE DATABASE       |
| 117            | CREATE OPERATOR          |
| 118            | ALTER OPERATOR           |
| 119            | DROP OPERATOR            |
| 120            | CREATE INDEXTYPE         |

| <b>type_no</b> | <b>name</b>             |
|----------------|-------------------------|
| 121            | ALTER INDEXTYPE         |
| 122            | DROP INDEXTYPE          |
| 123            | ASSOCIATE STATISTICS    |
| 124            | DISASSOCIATE STATISTICS |



# 색인

## A

- ACF\_CLGC\_CNT, 2
- ACF\_CMPT\_CNT, 2
- ACF\_NMGR\_CM\_AUTO\_REVIVAL, 3
- ACF\_NMGR\_PREVENT\_REVIVAL\_ON\_IR, 3
- ACT\_DUMP\_BACKUP\_LIMIT\_CNT, 5
- ACT\_DUMP\_DEST, 5
- ACT\_DUMP\_TOTAL\_SIZE\_LIMIT, 5
- ACT\_LOG\_DEST, 6
- ACT\_LOG\_FILE\_SIZE, 6
- ACT\_LOG\_TOTAL\_SIZE\_LIMIT, 6
- ACTIVE\_SESSION\_HISTORY, 3
- ACTIVE\_SESSION\_TIMEOUT, 4
- ACTIVE\_TX\_SESSION\_TIMEOUT, 4
- AD\_AUTH\_KBR\_SERVICE\_NAME, 7
- AD\_AUTH\_KEYTAB\_FILE, 7
- AD\_AUTH\_PREFIX, 7
- Adjustable
  - initialization parameter, 2
- ALL\_ARGUMENTS, 215
- ALL\_CATALOG, 216
- ALL\_COL\_COMMENTS, 218
- ALL\_COL\_PENDING\_STATS, 218
- ALL\_COL\_PRIVS, 219
- ALL\_COL\_PRIVS\_MADE, 220
- ALL\_COL\_PRIVS\_RECD, 220
- ALL\_COLL\_TYPES, 217
- ALL\_CONS\_COLUMNS, 223
- ALL\_CONS\_COLUMNS\_INFO, 223
- ALL\_CONSTRAINTS, 221
- ALL\_DB\_LINKS, 224
- ALL\_DEPENDENCIES, 225
- ALL\_DIRECTORIES, 226
- ALL\_ENCRYPTED\_COLUMNS, 227
- ALL\_ERRORS, 227
- ALL\_EXTERNAL\_LOCATIONS, 228
- ALL\_EXTERNAL\_TABLES, 228
- ALL\_IDX\_AND\_LOB\_COLUMNS\_INFO, 229
- ALL\_IDX\_COLUMNS, 229
- ALL\_IDX\_COLUMNS\_INFO, 230
- ALL\_IDX\_EXPRESSIONS, 231
- ALL\_IDX\_PARTITIONS, 232
- ALL\_IDX\_SUBPARTITIONS, 233
- ALL\_IND\_PENDING\_STATS, 237
- ALL\_INDEXES, 235
- ALL\_JAVA\_ARGUMENTS, 238
- ALL\_JAVA\_CLASSES, 239
- ALL\_JAVA\_FIELDS, 240
- ALL\_JAVA\_METHODS, 241
- ALL\_JOBS, 242
- ALL\_JOBS\_WITH\_NAME, 243
- ALL\_LIBRARIES, 244
- ALL\_LOB\_PARTITIONS, 246
- ALL\_LOB\_SUBPARTITIONS, 247
- ALL\_LOBS, 244
- ALL\_LOG\_GROUP\_COLUMNS, 250
- ALL\_LOG\_GROUPS, 249
- ALL\_LOGFILE\_MEMBERS\_SANITY\_CHECK, 8
- ALL\_METHOD\_PARAMS, 250
- ALL\_METHOD\_RESULTS, 251
- ALL\_MVIEW\_COMMENTS, 255
- ALL\_MVIEW\_DETAIL\_RELATIONS, 255
- ALL\_MVIEW\_LOGS, 256
- ALL\_MVIEW\_REFRESH\_TIMES, 256
- ALL\_MVIEWS, 252
- ALL\_OBJECTS, 257
- ALL\_PACKAGE\_COLL\_TYPES, 259
- ALL\_PACKAGE\_MEMBER\_TYPES, 259
- ALL\_PACKAGE\_TYPES, 261
- ALL\_PART\_INDEXES, 261
- ALL\_PART\_KEY\_COLUMNS, 263
- ALL\_PART\_LOBS, 263
- ALL\_PART\_TABLES, 264
- ALL\_PROCEDURES, 266
- ALL\_QUEUE\_SUBSCRIBERS, 268
- ALL\_QUEUE\_TABLES, 269
- ALL\_QUEUES, 267
- ALL\_RECYCLEBIN, 270
- ALL\_SCHEDULER\_CHAINS, 271

ALL\_SCHEDULER\_JOB\_LOG, 275  
 ALL\_SCHEDULER\_JOB\_RUN\_DETAILS, 276  
 ALL\_SCHEDULER\_JOBS, 272  
 ALL\_SCHEDULER\_PROGRAMS, 277  
 ALL\_SCHEDULER\_RULE\_CTX, 279  
 ALL\_SCHEDULER\_RULES, 278  
 ALL\_SCHEDULER\_RUNNING\_JOBS, 279  
 ALL\_SCHEDULER\_SCHEDULES, 283  
 ALL\_SCHEDULER\_STEP\_CTX, 284  
 ALL\_SCHEDULER\_STEPS, 284  
 ALL\_SEGMENTS, 285  
 ALL\_SEQUENCES, 287  
 ALL\_SOURCE, 288  
 ALL\_STAT\_EXTENSIONS, 289  
 ALL\_SUBPART\_KEY\_COLUMNS, 290  
 ALL\_SUBPARTITION\_TEMPLATES, 289  
 ALL\_SYNONYMS, 290  
 ALL\_TAB\_COMMENTS, 295  
 ALL\_TAB\_HISTGRM\_PENDING\_STATS, 295  
 ALL\_TAB\_IDENTITY\_COLS, 296  
 ALL\_TAB\_PENDING\_STATS, 296  
 ALL\_TABLES, 291  
 ALL\_TBL\_COL\_STATISTICS, 301  
 ALL\_TBL\_COLS, 297  
 ALL\_TBL\_COLUMNS, 299  
 ALL\_TBL\_COLUMNS\_WITH\_DNO, 300  
 ALL\_TBL\_HISTOGRAMS, 302  
 ALL\_TBL\_PARTITIONS, 303  
 ALL\_TBL\_PRIVS, 305  
 ALL\_TBL\_PRIVS\_MADE, 306  
 ALL\_TBL\_PRIVS\_RECD, 307  
 ALL\_TBL\_SUBPARTITIONS, 307  
 ALL\_TRIGGER\_COLS, 311  
 ALL\_TRIGGERS, 310  
 ALL\_TYPE\_ATTRS, 313  
 ALL\_TYPE\_METHODS, 313  
 ALL\_TYPES, 312  
 ALL\_UNUSED\_COL\_TABS, 314  
 ALL\_UPDATABLE\_COLUMNS, 315  
 ALL\_USERS, 316  
 ALL\_VIEWS, 316  
 ALL\_VIEWS\_INTERNAL, 316  
 archive file, 1

AS\_ADDR, 8  
 AS\_ALLOW\_ONLY\_RAW\_DISKS, 8  
 AS\_DDL\_DROP\_FILE, 9  
 AS\_DISK\_REPAIR\_WAIT\_TIME, 9  
 AS\_DISKSTRING, 9  
 AS\_PORT, 10  
 AS\_SCAN\_SSVR\_DISK, 10  
 AS\_WTHR\_CNT, 10  
 AUDIT\_FILE\_DEST, 11  
 AUDIT\_FILE\_SIZE, 11  
 AUDIT\_LOG\_TOTAL\_SIZE\_LIMIT, 11  
 AUDIT\_SYS\_OPERATIONS, 12  
 AUDIT\_TRAIL, 12  
 AUTHENTICATION\_TIMEOUT, 12  
 AUTO\_COALESCE, 13  
 AUTO\_COALESCE\_INTERVAL, 13  
 AUTO\_COALESCE\_STOP, 13

## B

BATCH\_INSERT\_ROW\_CNT\_THRESHOLD, 14  
 BATCHUPDATE\_MODE, 14  
 BIND\_VARIABLE\_CAPTURE, 14  
 BIND\_VARIABLE\_CAPTURE\_ALLOC\_COUNT, 15  
 BIND\_VARIABLE\_CAPTURE\_INTERVAL, 15  
 BIND\_VARIABLE\_CAPTURE\_KEEP\_FIRST, 15  
 BIND\_VARIABLE\_CAPTURE\_MAX\_COUNT, 16  
 BIND\_VARIABLE\_CAPTURE\_MAX\_VAR\_COUNT, 16  
 BIND\_VARIABLE\_CAPTURE\_POOL\_PCTDEL, 16  
 BIND\_VARIABLE\_CAPTURE\_POOL\_SIZE, 17  
 BLOCK\_CHANGE\_TRACKING, 17  
 BOOT\_WITH\_AUTO\_DOWN\_CLEAN, 17  
 BOOT\_WITH\_UNKNOWN\_PARAMETER, 18  
 BOOT\_WITH\_WALLET\_CHECK, 18  
 BTIP\_FILE\_PATH, 18

## C

CACHE\_SIZE\_FACTOR, 19  
 CALLSTACK\_DUMP\_BACKUP\_LIMIT\_CNT, 19  
 CALLSTACK\_DUMP\_DEST, 20  
 CALLSTACK\_DUMP\_TOTAL\_SIZE\_LIMIT, 20  
 CC\_CU\_BLKCNT, 22  
 CC\_CU\_PCTUSE, 22

CC\_CU\_WRITEOUT\_THRESHOLD, 22  
 CC\_EXPECTED\_RATE, 23  
 CC\_TYPICAL\_ROW\_SIZE, 23  
 CCC\_CATH\_CNT, 20  
 CCC\_MASTERING\_UNIT, 21  
 CCC\_RECL\_MAX\_RESOURCES, 21  
 CCC\_RSBTBL\_SIZE, 21  
 CERTIFICATE\_FILE, 23  
 CHECK\_RESULT\_FOR\_MEMLOG\_DUMP, 24  
 CLOSE\_REMOTE\_SESSION\_ON\_SESSION\_RESET, 24  
 CLUSTER\_DATABASE, 24  
 CM\_AGENT\_THREAD\_CHECK\_EXPIRE, 25  
 CM\_AGENT\_THREAD\_INTERVAL, 25  
 CM\_AUTO\_RECONNECT\_NODE\_WITH\_ACTIVE\_FHB, 26  
 CM\_AUTO\_RECONNECT\_TIMEOUT, 26  
 CM\_AUTO\_REVIVAL, 27  
 CM\_CLUSTER\_AUTO\_RESTART, 27  
 CM\_CLUSTER\_WORKER\_CNT, 27  
 CM\_CSP\_FENCE\_WAIT\_CNT, 28  
 CM\_DOWN\_ON\_TIME\_DISACCORDANCE, 28  
 CM\_ENABLE\_CLUSTER\_WORKER, 28  
 CM\_ENABLE\_FAST\_ACT\_LOOP, 29  
 CM\_ENABLE\_FAST\_INST\_DOWN\_MODE, 29  
 CM\_ENABLE\_FAST\_MASTER\_ELECTION, 29  
 CM\_ENABLE\_FAST\_NET\_ERROR\_DETECTION, 30  
 CM\_ENABLE\_SET\_LIMITS\_FOR\_INSTANCE, 30  
 CM\_ENABLE\_WAIT\_DOWN\_NOTI\_FOR\_SVC, 30  
 CM\_FENCE, 31  
 CM\_FILE\_AUTO\_RESTART\_SSVR, 31  
 CM\_FORCE\_INSTANCE\_KILL, 31  
 CM\_GUARD\_LOG\_DEST, 32  
 CM\_HEARTBEAT\_EXPIRE, 32  
 CM\_ID, 32  
 CM\_LOCAL\_IP\_VERSION, 33  
 CM\_LOG\_DEST, 33  
 CM\_LOG\_FILE\_SIZE, 33  
 CM\_LOG\_TOTAL\_SIZE\_LIMIT, 34  
 CM\_LOOP\_TIMEOUT, 34  
 CM\_MODE\_OBSERVER, 34  
 CM\_MST\_SAFEGUARD\_EXPIRE, 35  
 CM\_NAME, 35  
 CM\_NET\_EXPIRE\_MARGIN, 36  
 CM\_NETWORK\_AUTO\_RESTART, 35  
 CM\_OBSERVER\_EXPIRE, 36  
 CM\_OBSERVER\_PORT, 36  
 CM\_PORT, 37  
 CM\_REGARD\_NETHB\_AS\_SERIOUS, 37  
 CM\_RESOURCE\_FILE, 37  
 CM\_RESOURCE\_FILE\_BACKUP, 38  
 CM\_RESOURCE\_FILE\_BACKUP\_INTERVAL, 38  
 CM\_REVIVAL\_WAIT\_INST\_TIMEOUT, 39  
 CM\_REVIVAL\_WAIT\_NODE\_TIMEOUT, 39  
 CM\_REVIVAL\_WAIT\_TIMEOUT, 39  
 CM\_SVC\_AUTO\_START\_ON\_BOOT, 40  
 CM\_SVC\_DOWN\_NOTI\_WAIT\_TIMEOUT, 40  
 CM\_TAS\_VM\_ADDR, 40  
 CM\_TAS\_VM\_PORT, 41  
 CM\_TIME\_DISACCORDANCE\_NOTI\_INTERVAL, 41  
 CM\_TIME\_UNIT, 41  
 CM\_UI\_PORT, 42  
 CM\_USE\_ASENV, 42  
 CM\_USE\_AWS\_EC2, 42  
 CM\_USE\_AWS\_EC2\_PRIVATE, 43  
 CM\_USE\_AWS\_EC2\_PUBLIC, 43  
 CM\_USE\_AZURE\_VM\_PRIVATE, 43  
 CM\_USE\_KUBERNETES, 44  
 CM\_USE\_TAS5\_STORAGE, 44  
 CM\_VIP\_CHECK\_KUBERNETES\_SERVICE\_STATUS, 44  
 CM\_VIP\_FAILOVER\_BASED\_ON\_SVC\_STATUS, 45  
 CM\_VIP\_RESTART\_COUNT, 45  
 CM\_VIP\_STAT\_CHECK\_INTERVAL, 45  
 CM\_VIP\_USE\_KUBERNETES\_EXTERNAL\_IP, 46  
 CM\_WATCHDOG\_EXPIRE, 46  
 CMPT\_CNT\_FOR\_RECONFIGURATION, 25  
 command, 559  
 COMMIT\_FLUSH\_REQUEST, 46  
 COMMIT\_FLUSH\_WAIT, 47  
 COMMIT\_POINT\_STRENGTH, 47  
 control file, 1  
 CONTROL\_FILES, 47  
 CPU\_COUNT, 48  
 CREATE\_STORED\_OUTLINES, 48  
 CRYPTO\_LEVEL, 48

CTHR\_CHECK\_CONNECTION\_INTERVAL, 49  
CTX\_PREFERENCE\_VALUES, 317  
CTX\_PREFERENCES, 317  
CTX\_STOPLISTS, 318  
CURSOR\_SHARING, 49  
CWS\_DDD\_TIMEOUT\_ON\_SLOG, 49  
CWS\_RECL\_MAX\_RESOURCES, 50  
CWS\_RESOLVE\_CVT\_DEADLK\_FORCE, 50  
CWS\_RSBTBL\_SIZE, 50

## D

DATABASE\_PROPERTIES, 318  
DATABASE\_REPLAY\_CAPTURE\_DEST, 51  
DATABASE\_REPLAY\_CAPTURE\_DIR\_LIMIT, 51  
DATABASE\_REPLAY\_CAPTURE\_FILE\_SIZE, 51  
DATABASE\_REPLAY\_CAPTURE\_WORKLOAD, 52  
DATABASE\_REPLAY\_DUMP\_MMAP\_SIZE, 52  
DB\_BLOCK\_SIZE, 56  
DB\_CACHE\_SIZE, 56  
DB\_CREATE\_FILE\_DEST, 56  
DB\_FILE\_MULTIBLOCK\_READ\_COUNT, 57  
DB\_KEEP\_CACHE\_SIZE, 57  
DB\_MULTIBLOCK\_PREFETCH\_COUNT, 57  
DB\_NAME, 58  
DB\_RECYCLE\_CACHE\_SIZE, 58  
DB\_UNIQUE\_NAME, 58  
DBA\_2PC\_NEIGHBORS, 318  
DBA\_2PC\_PENDING, 319  
DBA\_AUDIT\_POLICIES, 319  
DBA\_AUDIT\_TRAIL, 321  
DBA\_CATALOG, 322  
DBA\_COL\_COMMENTS, 322  
DBA\_COL\_PENDING\_STATS, 322  
DBA\_COL\_PRIVS, 323  
DBA\_COLL\_TYPES, 322  
DBA\_CONS\_COLUMNS, 323  
DBA\_CONSTRAINTS, 323  
DBA\_DATA\_FILES, 323  
DBA\_DB\_LINKS, 324  
DBA\_DEPENDENCIES, 325  
DBA\_DIRECTORIES, 325  
DBA\_ENCRYPTED\_COLUMNS, 325

DBA\_ERRORS, 325  
DBA\_EXTENTS, 325  
DBA\_EXTERNAL\_LOCATIONS, 326  
DBA\_EXTERNAL\_TABLES, 326  
DBA\_FGA\_AUDIT\_TRAIL, 327  
DBA\_FREE\_SPACE, 327  
DBA\_FREE\_TABLESPACE, 327  
DBA\_HIST\_ACTIVE\_SESS\_HISTORY, 328  
DBA\_HIST\_FILESTAT, 330  
DBA\_HIST\_LATCH, 331  
DBA\_HIST\_LIBRARYCACHE, 331  
DBA\_HIST\_OSSTAT, 332  
DBA\_HIST\_PGASTAT, 332  
DBA\_HIST\_PROCESS\_MEM\_SUMMARY, 333  
DBA\_HIST\_SEG\_STAT, 333  
DBA\_HIST\_SGASTAT, 334  
DBA\_HIST\_SNAPSHOT, 334  
DBA\_HIST\_SQL\_PLAN, 337  
DBA\_HIST\_SQL\_WORKAREA\_HSTGRM, 338  
DBA\_HIST\_SQLBIND, 334  
DBA\_HIST\_SQLSTAT, 335  
DBA\_HIST\_SQLTEXT, 336  
DBA\_HIST\_SYSMETRIC\_SUMMARY, 339  
DBA\_HIST\_SYSTEM\_EVENT, 340  
DBA\_HIST\_UNDOSTAT, 340  
DBA\_HIST\_WAITSTAT, 341  
DBA\_IDX\_COLUMNS, 341  
DBA\_IDX\_EXPRESSIONS, 342  
DBA\_IDX\_PARTITIONS, 342  
DBA\_IDX\_SUBPARTITIONS, 342  
DBA\_IND\_PENDING\_STATS, 342  
DBA\_INDEXES, 342  
DBA\_JAVA\_ARGUMENTS, 343  
DBA\_JAVA\_CLASSES, 343  
DBA\_JAVA\_FIELDS, 343  
DBA\_JAVA\_METHODS, 343  
DBA\_JOBS, 344  
DBA\_JOBS\_WITH\_NAME, 344  
DBA\_LIBRARIES, 344  
DBA\_LOB\_PARTITIONS, 344  
DBA\_LOB\_SUBPARTITIONS, 345  
DBA\_LOBS, 344  
DBA\_LOG\_GROUP\_COLUMNS, 345

|                                    |                                          |
|------------------------------------|------------------------------------------|
| DBA_LOG_GROUPS, 345                | DBA_SQL_PLAN_BASELINES, 361              |
| DBA_METHOD_PARAMS, 345             | DBA_SQL_PROFILES, 363                    |
| DBA_METHOD_RESULTS, 346            | DBA_STAT_EXTENSIONS, 363                 |
| DBA_MVIEW_COMMENTS, 346            | DBA_SUBPART_KEY_COLUMNS, 364             |
| DBA_MVIEW_DETAIL_RELATIONS, 346    | DBA_SUBPARTITION_TEMPLATES, 364          |
| DBA_MVIEW_LOGS, 347                | DBA_SYNONYMS, 364                        |
| DBA_MVIEW_REFRESH_TIMES, 347       | DBA_SYS_PRIVS, 364                       |
| DBA_MVIEWS, 346                    | DBA_TAB_COMMENTS, 368                    |
| DBA_OBJ_AUDIT_OPTS, 347            | DBA_TAB_HISTGRM_PENDING_STATS, 368       |
| DBA_OBJECTS, 347                   | DBA_TAB_IDENTITY_COLS, 368               |
| DBA_PACKAGE_COLL_TYPES, 348        | DBA_TAB_PENDING_STATS, 368               |
| DBA_PACKAGE_MEMBER_TYPES, 349      | DBA_TABLES, 365                          |
| DBA_PACKAGE_TYPES, 349             | DBA_TABLESPACES, 365                     |
| DBA_PARALLEL_EXECUTE_CHUNKS, 349   | DBA_TBL_COL_STATISTICS, 369              |
| DBA_PARALLEL_EXECUTE_TASKS, 350    | DBA_TBL_COLS, 369                        |
| DBA_PART_INDEXES, 351              | DBA_TBL_COLUMNS, 369                     |
| DBA_PART_KEY_COLUMNS, 352          | DBA_TBL_HISTOGRAMS, 369                  |
| DBA_PART_LOBS, 352                 | DBA_TBL_PARTITIONS, 370                  |
| DBA_PART_TABLES, 352               | DBA_TBL_PRIVS, 370                       |
| DBA_PRIV_AUDIT_OPTS, 352           | DBA_TBL_SUBPARTITIONS, 370               |
| DBA_PROCEDURES, 353                | DBA_TEMP_FILES, 370                      |
| DBA_PROFILES, 353                  | DBA_TRIGGER_COLS, 371                    |
| DBA_QUEUE_SUBSCRIBERS, 354         | DBA_TRIGGERS, 371                        |
| DBA_QUEUE_TABLES, 354              | DBA_TS_QUOTAS, 372                       |
| DBA_QUEUES, 353                    | DBA_TYPE_ATTRS, 372                      |
| DBA_RECYCLEBIN, 354                | DBA_TYPE_METHODS, 373                    |
| DBA_REDEFINITION_OBJECTS, 355      | DBA_TYPES, 372                           |
| DBA_ROLE_PRIVS, 356                | DBA_UNDO_EXTENTS, 373                    |
| DBA_ROLES, 356                     | DBA_UNDO_SGMTS, 373                      |
| DBA_SCHEDULER_CHAINS, 357          | DBA_UNUSED_COL_TABS, 374                 |
| DBA_SCHEDULER_JOB_LOG, 357         | DBA_UPDATABLE_COLUMNS, 374               |
| DBA_SCHEDULER_JOB_RUN_DETAILS, 357 | DBA_USERS, 374                           |
| DBA_SCHEDULER_JOBS, 357            | DBA_VIEWS, 376                           |
| DBA_SCHEDULER_PROGRAMS, 357        | DBLINK_BATCH_UPDATE_ROW_COUNT, 52        |
| DBA_SCHEDULER_RULE_CTX, 358        | DBLINK_CONN_TIMEOUT, 53                  |
| DBA_SCHEDULER_RULES, 358           | DBLINK_CONN_VALIDATION_SKIP_INTERVAL, 53 |
| DBA_SCHEDULER_RUNNING_JOBS, 358    | DBLINK_QUERY_TIMEOUT, 53                 |
| DBA_SCHEDULER_SCHEDULES, 358       | DBMS_LOG_DEST, 54                        |
| DBA_SCHEDULER_STEP_CTX, 359        | DBMS_LOG_FILE_SIZE, 54                   |
| DBA_SCHEDULER_STEPS, 359           | DBMS_LOG_TOTAL_SIZE_LIMIT, 55            |
| DBA_SEGMENTS, 359                  | DBMSHXP_SQL_PROFILE_ATTR, 376            |
| DBA_SEQUENCES, 361                 | DBWR_CNT, 55                             |
| DBA_SOURCE, 361                    | DBWR_USE_AIO, 55                         |

DD\_CACHE\_MAX, 59  
DDL, 196  
DEFAULT\_SEQ\_CACHE\_SIZE, 59  
DF\_HDRBLK\_START\_OFFSET, 60  
DFLT\_DUMP\_BACKUP\_LIMIT\_CNT, 59  
DFLT\_DUMP\_TOTAL\_SIZE\_LIMIT, 60  
DIAG\_DUMP\_BACKUP\_LIMIT\_CNT, 61  
DIAG\_DUMP\_DEST, 61  
DIAG\_DUMP\_TOTAL\_SIZE\_LIMIT, 61  
DIAG\_LOG\_DEST, 62  
DIAG\_LOG\_FILE\_SIZE, 62  
DIAG\_LOG\_TOTAL\_SIZE\_LIMIT, 62  
DICTIONARY, 376  
DIRTY\_BLOCK\_WRITE\_ISSUE\_TIME, 63  
DISK\_AFFINITY\_GROUP, 63  
DISK\_PATH, 63  
DONT\_USE\_FILE\_BUFFERING, 64  
DP\_USE\_O\_SYNC, 65  
DPBUF\_BLOCK\_COUNT, 64  
DPL\_INDEX\_METHOD, 64  
DUMP\_EXSTAT\_ON\_STMT\_CANCEL, 65  
DUMP\_PATH\_CHECK\_INTERVAL, 65  
Dynamic  
    initialization parameter, 2  
dynamic view, 417

## E

ENABLE\_BCT\_AT\_BOOTING, 66  
ENABLE\_MULTI\_LOG\_BUFFER, 66  
ENCRYPTION\_WALLET\_FILE, 66  
EX\_MEMORY\_AUTO\_MANAGEMENT, 68  
EX\_MEMORY\_COMPENSATE\_INTERVAL, 68  
EX\_MEMORY\_DRIFT\_CHECK\_INTERVAL, 68  
EX\_MEMORY\_DRIFT\_RATIO, 69  
EX\_MEMORY\_SOFT\_LIMIT\_RATIO, 69  
EXTRA\_LISTENER\_IPS, 67  
EXTRA\_LISTENER\_PORTS, 67  
EXTRA\_LISTENER\_PORTS\_6, 67

## F

FDPOOL\_HASH\_SIZE, 69  
FLASHBACK\_LOG\_ARCHIVE\_DEST, 70

FLASHBACK\_LOG\_ARCHIVE\_FORMAT, 70  
FLASHBACK\_LOG\_BUFFER, 70  
FLASHBACK\_MARKER\_INTERVAL, 71

## G

GATHER\_SQL\_EXEC\_TIME, 71  
GATHER\_SQL\_PLAN\_STAT, 72

## H

HASH\_AREA\_SIZE, 72  
HBASE\_CONNECTOR\_SCAN\_CACHE, 72  
HSM\_LIB\_FILE, 73  
HSM\_NOT\_USE\_WRAPPING, 73  
HSM\_TTE\_KEY\_BACKUP\_FILE\_PATH, 73

## I

IFF\_STORAGE\_PROCESSING, 74  
ILOG\_BACKUP\_SIZE\_LIMIT, 74  
ILOG\_DEST, 74  
ILOG\_FILE\_BACKUP\_ON\_BOOT, 75  
ILOG\_FILE\_SIZE, 75  
ILOG\_KEEP\_BACKUP\_ASIDE, 75  
ILOG\_MAP, 76  
ILOG\_MMAP\_SIZE, 76  
ILOG\_TOTAL\_SIZE\_LIMIT, 77  
initialization parameter class attribute, 2  
INMEMORY\_QUERY, 77  
INMEMORY\_SIZE, 77  
INSTANCE\_NUMBER, 78  
INSTANCE\_TYPE, 78  
IP\_VERSION, 78

## J

JAVA\_CLASS\_PATH, 79  
JOB\_HISTORY\_LOG, 79  
JOB\_SCHEDULING\_MAX\_TIMEOUT, 79

## K

KEEP\_STAT\_PER\_SQL, 80

## L

LGWR\_USE\_AIO, 80  
LIBRARY\_CACHE\_MAX, 80  
LIBRARY\_DD\_CACHE\_MAX, 81  
list of dynamic views, 417  
list of static views, 199  
LISTENER\_IP, 81  
LISTENER\_IP\_6, 81  
LISTENER\_PORT, 82  
LISTENER\_PORT\_6, 82  
LISTENER\_VIP\_PORT, 83  
LOAD\_METRIC\_AUTO\_INTERVAL, 83  
LOAD\_METRIC\_COLLECT\_INTERVAL\_MAX, 83  
LOAD\_METRIC\_COLLECT\_INTERVAL\_MIN, 84  
LOAD\_METRIC\_HISTORY\_COUNT, 84  
LOCAL\_CLUSTER\_ADDR, 84  
LOCAL\_CLUSTER\_ADDR\_IPV6, 85  
LOCAL\_CLUSTER\_IP\_VERSION, 85  
LOCAL\_CLUSTER\_PORT, 85  
LOCAL\_CLUSTER\_PORT\_IPV6, 86  
LOG\_ARCHIVE\_DEST, 86  
LOG\_ARCHIVE\_DEST\_1, 86  
LOG\_ARCHIVE\_DEST\_2, 87  
LOG\_ARCHIVE\_DEST\_3, 87  
LOG\_ARCHIVE\_DEST\_4, 87  
LOG\_ARCHIVE\_DEST\_5, 88  
LOG\_ARCHIVE\_DEST\_6, 88  
LOG\_ARCHIVE\_DEST\_7, 88  
LOG\_ARCHIVE\_DEST\_8, 89  
LOG\_ARCHIVE\_DEST\_9, 89  
LOG\_ARCHIVE\_FORMAT, 89  
LOG\_ARCHIVE\_OVERWRITE, 90  
LOG\_ARCHIVE\_RETRY\_TIME, 90  
LOG\_BUFFER, 90  
LOG\_BUFFER\_CNT, 91  
LOG\_CHECKPOINT\_INTERVAL, 91  
LOG\_CHECKPOINT\_TIMEOUT, 91  
LOG\_DEFAULT\_DEST, 92  
LOG\_FILE\_WITH\_GROUP\_READ\_PERM, 92  
LOG\_LVL\_AS, 92  
LOG\_LVL\_CM, 93  
LOG\_LVL\_LIS, 93

LOG\_ON\_MEMORY, 93  
LOG\_ON\_MEMORY\_SIZE, 94  
LOG\_REPLICATION\_1\_ENABLE, 94  
LOG\_REPLICATION\_2\_ENABLE, 94  
LOG\_REPLICATION\_3\_ENABLE, 95  
LOG\_REPLICATION\_4\_ENABLE, 95  
LOG\_REPLICATION\_5\_ENABLE, 95  
LOG\_REPLICATION\_6\_ENABLE, 96  
LOG\_REPLICATION\_7\_ENABLE, 96  
LOG\_REPLICATION\_8\_ENABLE, 96  
LOG\_REPLICATION\_9\_ENABLE, 97  
LOG\_REPLICATION\_DEST\_1, 97  
LOG\_REPLICATION\_DEST\_2, 97  
LOG\_REPLICATION\_DEST\_3, 98  
LOG\_REPLICATION\_DEST\_4, 98  
LOG\_REPLICATION\_DEST\_5, 98  
LOG\_REPLICATION\_DEST\_6, 99  
LOG\_REPLICATION\_DEST\_7, 99  
LOG\_REPLICATION\_DEST\_8, 100  
LOG\_REPLICATION\_DEST\_9, 100  
LOG\_REPLICATION\_MODE, 100  
LONG\_OPERATION\_KEEP\_COUNT, 101  
LONGOPS\_KEEP\_INTERVAL, 101  
LONGOPS\_THRESHOLD\_SEC, 101  
LSNR\_DENIED\_IP, 102  
LSNR\_DENIED\_IP\_FILE, 102  
LSNR\_INVITED\_IP, 102  
LSNR\_INVITED\_IP\_FILE, 103  
LSNR\_LISTEN\_BACKLOG, 103  
LSNR\_LOG\_DEST, 103  
LSNR\_LOG\_FILE\_SIZE, 104  
LSNR\_LOG\_TOTAL\_SIZE\_LIMIT, 104  
LSNR\_OUTFILE\_BACKUP\_LIMIT\_CNT, 105  
LSNR\_OUTFILE\_DEST, 105  
LSNR\_SCHEDULE\_SAME\_PROC\_FIRST, 105

## M

Mandatory

initialization parameter, 2

MAX\_BG\_SESSION\_COUNT, 106  
MAX\_FG\_SESSION\_COUNT, 106  
MAX\_SESSION\_COUNT, 106

MEM\_LOG\_BACKUP\_LIMIT\_CNT, 108  
MEM\_LOG\_DEST, 109  
MEM\_LOG\_TOTAL\_SIZE\_LIMIT, 109  
MEMORY\_MAX\_TARGET, 107  
MEMORY\_MODEL, 107  
MEMORY\_TABLESPACE\_SIZE, 108  
MEMORY\_TARGET, 108

## N

NAMED\_PIPE\_DIR, 109  
NBU\_ARCHIVE\_POLICY\_NAME, 110  
NBU\_ARCHIVE\_RETRY\_CNT, 110  
NBU\_ARCHIVELOG\_SEARCH, 110  
NBU\_BACKUP\_INST\_SID, 111  
NBU\_BACKUP\_POLICY\_NAME, 111  
NBU\_BSA\_API\_VERSION, 111  
NBU\_BSA\_LOG\_DIRECTORY, 112  
NBU\_BSA\_LOG\_LEVEL, 112  
NBU\_CLIENT\_COUNT, 112  
NBU\_CLIENT\_HOST\_NAME, 116  
NBU\_CLIENT\_HOSTNAME\_0, 113  
NBU\_CLIENT\_HOSTNAME\_1, 113  
NBU\_CLIENT\_HOSTNAME\_2, 113  
NBU\_CLIENT\_HOSTNAME\_3, 114  
NBU\_CLIENT\_HOSTNAME\_4, 114  
NBU\_CLIENT\_HOSTNAME\_5, 114  
NBU\_CLIENT\_HOSTNAME\_6, 115  
NBU\_CLIENT\_HOSTNAME\_7, 115  
NBU\_CLIENT\_HOSTNAME\_8, 115  
NBU\_CLIENT\_HOSTNAME\_9, 116  
NBU\_MASTER\_HOST\_NAME, 116  
NBU\_OBJ\_OWNER\_GROUP\_NAME, 117  
NBU\_OBJ\_OWNER\_NAME, 117  
NBU\_PROC\_WTHR\_CNT, 117  
NBU\_SCHEDULE\_NAME, 118  
NBU\_SKIP\_HOST\_CHECK, 118  
NBU\_TWO\_PHASE\_CF\_BACKUP, 118  
NETWORK\_SECURITY\_AGENT\_MANAGER\_PATH, 119  
NLS\_CALENDAR, 119  
NLS\_COMP, 119  
NLS\_CURRENCY, 120

NLS\_DATABASE\_PARAMETERS, 377  
NLS\_DATE\_FORMAT, 120  
NLS\_DATE\_LANGUAGE, 120  
NLS\_DUAL\_CURRENCY, 121  
NLS\_ISO\_CURRENCY, 121  
NLS\_LANG\_AT\_BOOT, 122  
NLS\_LANGUAGE, 121  
NLS\_LENGTH\_SEMANTICS, 122  
NLS\_NUMERIC\_CHARACTERS, 122  
NLS\_SESSION\_PARAMETERS, 377  
NLS\_SORT, 123  
NLS\_TERRITORY, 123  
NLS\_TIME\_FORMAT, 124  
NLS\_TIME\_TZ\_FORMAT, 125  
NLS\_TIMESTAMP\_FORMAT, 124  
NLS\_TIMESTAMP\_TZ\_FORMAT, 124  
NTF\_LISTEN\_PORT, 125

## O

OMIT\_PSM\_SQL\_LOG\_ON\_MEMORY, 125  
OPEN\_CURSORS, 126  
OPTIMIZER\_CAPTURE\_SQL\_PLAN\_BASELINES, 126  
OPTIMIZER\_CAPTURE\_SQL\_PLAN\_BASELINES\_DUR  
ATION, 126  
OPTIMIZER\_LOG\_OUTLINE, 127  
OPTIMIZER\_MODE, 127  
OPTIMIZER\_TRACE, 127  
OPTIMIZER\_USE\_INVISIBLE\_INDEXES, 128  
OPTIMIZER\_USE\_PENDING\_STATISTICS, 128  
OPTIMIZER\_USE\_SQL\_PLAN\_BASELINES, 128  
OPTIMIZER\_USE\_SQL\_PROFILES, 129  
Optional  
    initialization parameter, 2  
OS\_AUTH\_PREFIX, 129  
OUTFILE\_PATH\_BACKUP\_LIMIT\_CNT, 129

## P

PARALLEL\_DEFAULT\_DOP, 130  
PARALLEL\_GRANULE\_SIZE, 130  
PARALLEL\_THREADS\_PER\_CPU, 130  
PEP\_PROC\_CNT, 131  
Persistent

initialization parameter, 2  
 PP\_SAVE\_SYS\_BINDVAR, 131  
 PRIVKEY\_FILE, 131  
 PSM\_CODE\_TYPE, 132  
 PSM\_DEBUG, 132  
 PSM\_ERROR\_CONV, 132  
 PSM\_LOG\_ON\_THROW, 133  
 PSM\_SHLIB\_DIR, 133  
 PSM\_SQL\_RESULT\_CACHE, 133  
 PUBLICSYN, 377

## Q

QUERY\_REWRITE\_ENABLED, 134  
 QUERY\_REWRITE\_INTEGRITY, 134

## R

RECO\_PROC\_WTHR\_CNT, 134  
 REDACTION\_COLUMNS, 378  
 REDACTION\_COLUMNS\_DBMS\_ERRLOG, 379  
 REDACTION\_POLICIES, 379  
 REDACTION\_VALUES\_FOR\_TYPE\_FULL, 380  
 RESULT\_CACHE\_MAX\_RATIO, 135  
 RESULT\_CACHE\_MAX\_SIZE, 135  
 RESULT\_CACHE\_MODE, 135  
 RESUMABLE\_SPACE\_ALLOC, 136  
 RESUMABLE\_SPACE\_ALLOC\_TIMEOUT, 136  
 RMGR\_BACKUP\_DEST, 136  
 RMGR\_CATALOG\_SID, 137  
 RMGR\_CHECK\_FREE\_DISK\_SIZE, 137  
 RMGR\_NBU\_BACKUP\_DEST, 137  
 RMGR\_PRIMARY\_SID, 138  
 RMGR\_RESTORE\_BACKUP\_CF, 138  
 RMGR\_TAR\_PATH, 138  
 ROLE\_ROLE\_PRIVS, 380  
 ROLE\_SYS\_PRIVS, 381  
 ROLE\_TAB\_PRIVS, 381  
 RSBTBL\_BUCKET\_CNT\_PER\_ROOT\_LIST\_FOR\_RE  
 CONFIGURATION, 139  
 RSRC\_CHECK\_INTERVAL, 139  
 RSRC\_CONTROL\_THRESHOLD, 139  
 RT\_AWAKE\_WAITER\_INTERVAL, 140

RT\_UPDATE\_ALLOWABLE\_INFLOW\_CNT\_INTER  
 VAL, 140

## S

SANITY\_CHECK\_DURING\_LOG\_SWITCH, 141  
 SECU\_CHANGE\_PASSWORD\_WITH\_CASE\_SENSI  
 TIVITY, 141  
 SEGMENT\_EXTEND\_LAST, 141  
 SEM\_KEY, 142  
 SEM\_SET\_COUNT, 142  
 SEQUENCE\_PIN\_BUCKET\_CNT, 142  
 SERVER\_LOAD\_TOLERANCE, 143  
 SERVER\_SIDE\_LOAD\_BALANCE, 143  
 SERVICE\_CONTROL\_STOP\_DOWNMODE, 143  
 SESS\_IGNORE\_INVALID\_CLIENT\_NLS\_SETTING,  
 144  
 Session  
   initialization parameter, 2  
 SESSION\_CONTEXT, 382  
 SESSION\_PRIVS, 382  
 SESSION\_ROLES, 383  
 SHARED\_PARAMETER\_FILE, 144  
 SHM\_KEY, 145  
 SHP\_ALLOC\_DUMP\_IDX, 145  
 SHP\_ALLOC\_FULL\_DUMP, 145  
 SLAB\_BLOCK\_RESERVED\_COUNT, 146  
 SLOG\_DEST, 146  
 SLOG\_DISPLAY\_YEAR, 146  
 SLOG\_FILE\_SIZE, 147  
 SLOG\_HISTORY, 147  
 SLOG\_HISTORY\_BUFSIZE, 147  
 SLOG\_HISTORY\_QSIZE, 148  
 SLOG\_TOTAL\_SIZE\_LIMIT, 148  
 SLOG\_VIEW\_SIZE, 148  
 SMR\_DELAY\_TIME, 149  
 SMR\_SGMT\_DROP\_TIMEOUT, 149  
 SORT\_AREA\_SIZE, 149  
 SQL 파서, 194  
 SQL\_LOG\_ON\_MEMORY, 150  
 SQL\_LOG\_ON\_MEMORY\_SIZE, 150  
 SQL\_STAT\_HISTORY, 151  
 SQL\_STAT\_HISTORY\_QSIZE, 151

SQL\_STAT\_HISTORY\_THRESHOLD, 151  
 SQL\_TRACE, 152  
 SQL\_TRACE\_BACKUP\_LIMIT\_CNT, 152  
 SQL\_TRACE\_DEST, 153  
 SQL\_TRACE\_LEVEL, 153  
 SQL\_TRACE\_TOTAL\_SIZE\_LIMIT, 153  
 SQL\_TRANSLATION\_PROFILE, 154  
 SQTUNE\_CATEGORY, 150  
 SSVR\_FC\_DIRTY\_BIN\_RATIO\_MAX, 154  
 SSVR\_FC\_FREE\_BIN\_RATIO, 154  
 SSVR\_FC\_WS\_CNT, 155  
 SSVR\_RECV\_PORT\_START, 155  
 SSVR\_TPM\_SENDER\_INTERVAL, 155  
 SSVR\_USE\_AGNT, 156  
 SSVR\_USE\_FC, 156  
 SSVR\_USE\_IB, 156  
 SSVR\_USE\_SDM, 157  
 SSVR\_USE\_TCP, 157  
 SSVR\_USE\_TPM, 157  
 SSVR\_WTHR\_CNT, 158  
 STANDBY\_ALLOWED\_IP\_ADDRESSES, 158  
 STANDBY\_ENABLE\_CASCADING, 158  
 STANDBY\_ENABLE\_LOG\_RECOVERY, 159  
 STANDBY\_ENABLE\_TAC\_MULTINODE, 159  
 STANDBY\_FILE\_NAME\_CONVERT, 159  
 STANDBY\_FLUSH\_WAIT, 160  
 STANDBY\_LOG\_ARCHIVE\_FORMAT, 160  
 STANDBY\_USE\_OBSERVER, 160  
 STAR\_TRANSFORMATION\_ENABLED, 161  
 Static  
     initialization parameter, 2  
 static view, 199  
 STORAGE\_PROCESSING, 161  
 SUMMARY\_FIRST\_IN\_GROUPBY\_CUBE, 161  
 SVR\_OUTFILE\_BACKUP\_LIMIT\_CNT, 162  
 SVR\_OUTFILE\_DEST, 162  
 SYS 사용자, 192  
 SYS\_OBJECTS, 383  
 System  
     initialization parameter, 2

## T

TABLE\_GLOBAL\_COMPRESSION, 162  
 TAS\_DB\_UUID, 163  
 TAS\_HOT\_BACKUP\_SYNC\_WITH\_TAC, 163  
 TAS\_OPEN\_FILE\_MAX, 164  
 TAS\_SERVER\_IP, 164  
 TAS\_VM\_ADDR, 164  
 TAS\_VM\_PORT, 165  
 TAS5\_STORAGE\_MODE, 163  
 TCP\_RCVBUF\_SIZE, 165  
 TCP\_SNDBUF\_SIZE, 165  
 TDE\_HASH\_VERIFICATION, 165  
 TDE\_SELF\_UPDATE\_MODIFICATION\_DEFENCE, 166  
 TEMP\_GRANULE\_SIZE, 166  
 TEMP\_TS\_CNT\_MAX, 166  
 THREAD, 167  
 THROW\_WHEN\_GETTING\_OSSTAT\_FAIL, 167  
 TIBERO\_PERFORMANCE\_REPOSITORY, 167  
 TOTAL\_SHM\_MAX\_SIZE, 168  
 TOTAL\_SHM\_SIZE, 168  
 TPR\_AGGREGATION, 168  
 TPR\_METRIC, 169  
 TPR\_PARTITION\_COUNT\_IN\_ADVANCE, 169  
 TPR\_SEGMENT\_STATISTICS, 169  
 TPR\_SNAPSHOT\_RETENTION, 170  
 TPR\_SNAPSHOT\_SAMPLING\_INTERVAL, 170  
 TPR\_SNAPSHOT\_TOP\_SEGMENT\_CNT, 170  
 TPR\_SNAPSHOT\_TOP\_SQL\_CNT, 171  
 TPR\_SQL\_PLAN\_STAT, 171  
 trace file, 1  
 TRACE\_BACKUP\_LIMIT\_CNT, 171  
 TRACE\_DUMP\_BACKUP\_LIMIT\_CNT, 172  
 TRACE\_DUMP\_DEST, 172  
 TRACE\_DUMP\_TOTAL\_SIZE\_LIMIT, 173  
 TRACE\_PATH\_CHECK\_INTERVAL, 173  
 TRACE\_TOTAL\_SIZE\_LIMIT, 173  
 TSAM\_PREFETCH\_SIZE, 174  
 TSN\_SYNC\_UDP\_PORT, 174  
 TX\_FORCE\_COMMIT\_DUMP, 174

## U

UNDO\_RETENTION, 175  
UNDO\_TABLESPACE, 175  
USE\_ACCURATE\_NUMERIC\_FUNC, 175  
USE\_ACTIVE\_STORAGE, 176  
USE\_AD\_AUTH, 176  
USE\_BINARY\_UPDATE, 176  
USE\_CASE\_SENSITIVE\_PASSWORD, 177  
USE\_DISK\_AFFINITY, 177  
USE\_DUMP\_PATH\_MANAGEMENT, 177  
USE\_HSM, 178  
USE\_HUGE\_PAGE, 178  
USE\_ILOG, 178  
USE\_NBU\_FOR\_ARCHIVELOG, 179  
USE\_NBU\_FOR\_BACKUPSET, 179  
USE\_NET\_KEEPAIVE, 180  
USE\_NETWORK\_SECURITY\_AGENT\_MANAGER, 179  
USE\_OUTFILE\_PATH\_MANAGEMENT, 180  
USE\_PROFILE, 180  
USE\_PROFILE\_LOGIN\_LOGOUT\_TIME, 181  
USE\_RECYCLEBIN, 181  
USE\_RECYCLEBIN\_AUTO\_PURGE, 181  
USE\_RESOURCE\_MANAGER, 182  
USE\_SERVER\_EVENT\_NOTIFICATION, 182  
USE\_SQL\_STAT\_HIST\_IN\_BG\_PROC, 182  
USE\_STANDBY\_REDO\_LOG, 183  
USE\_STORED\_OUTLINES, 183  
USE\_TRACE\_PATH\_MANAGEMENT, 183  
USE\_TRUNCATE\_OBJ\_PRIVILEGE, 184  
USE\_TRUNCATE\_PRIVILEGE, 184  
USE\_TS\_QUOTA, 184  
USE\_ZETA, 185  
USER\_ARGUMENTS, 383  
USER\_AUDIT\_TRAIL, 384  
USER\_CATALOG, 385  
USER\_COL\_COMMENTS, 385  
USER\_COL\_PENDING\_STATS, 385  
USER\_COL\_PRIVS, 385  
USER\_COL\_PRIVS\_MADE, 386  
USER\_COL\_PRIVS\_RECD, 386  
USER\_COLL\_TYPES, 385  
USER\_CONS\_COLUMNS, 386  
USER\_CONSTRAINTS, 386  
USER\_DB\_LINKS, 387  
USER\_DEPENDENCIES, 387  
USER\_ENCRYPTED\_COLUMNS, 387  
USER\_ERRORS, 387  
USER\_EXTENTS, 388  
USER\_EXTERNAL\_LOCATIONS, 388  
USER\_EXTERNAL\_TABLES, 388  
USER\_FREE\_SPACE, 389  
USER\_IDX\_COLUMNS, 389  
USER\_IDX\_EXPRESSIONS, 389  
USER\_IDX\_PARTITIONS, 390  
USER\_IND\_PENDING\_STATS, 390  
USER\_INDEXES, 390  
USER\_JAVA\_ARGUMENTS, 390  
USER\_JAVA\_CLASSES, 390  
USER\_JAVA\_FIELDS, 391  
USER\_JAVA\_METHODS, 391  
USER\_JOBS, 391  
USER\_JOBS\_WITH\_NAME, 391  
USER\_LIBRARIES, 392  
USER\_LOB\_PARTITIONS, 392  
USER\_LOB\_SUBPARTITIONS, 392  
USER\_LOBS, 392  
USER\_LOG\_GROUP\_COLUMNS, 393  
USER\_LOG\_GROUPS, 392  
USER\_METHOD\_PARAMS, 393  
USER\_METHOD\_RESULTS, 393  
USER\_MVIEW\_COMMENTS, 394  
USER\_MVIEW\_DETAIL\_RELATIONS, 394  
USER\_MVIEW\_LOGS, 394  
USER\_MVIEW\_REFRESH\_TIMES, 394  
USER\_MVIEWS, 393  
USER\_OBJ\_AUDIT\_OPTS, 395  
USER\_OBJECTS, 395  
USER\_PACKAGE\_COLL\_TYPES, 396  
USER\_PACKAGE\_MEMBER\_TYPES, 396  
USER\_PACKAGE\_TYPES, 396  
USER\_PARALLEL\_EXECUTE\_CHUNKS, 397  
USER\_PARALLEL\_EXECUTE\_TASKS, 398  
USER\_PART\_INDEXES, 399  
USER\_PART\_KEY\_COLUMNS, 399

USER\_PART\_LOBS, 399  
 USER\_PART\_TABLES, 400  
 USER\_PROCEDURES, 400  
 USER\_QUEUE\_SUBSCRIBERS, 400  
 USER\_QUEUE\_TABLES, 400  
 USER\_QUEUES, 400  
 USER\_RECYCLEBIN, 401  
 USER\_ROLE\_PRIVS, 401  
 USER\_SCHEDULER\_CHAINS, 402  
 USER\_SCHEDULER\_JOB\_LOG, 402  
 USER\_SCHEDULER\_JOB\_RUN\_DETAILS, 402  
 USER\_SCHEDULER\_JOBS, 402  
 USER\_SCHEDULER\_PROGRAMS, 403  
 USER\_SCHEDULER\_RULE\_CTX, 403  
 USER\_SCHEDULER\_RULES, 403  
 USER\_SCHEDULER\_RUNNING\_JOBS, 403  
 USER\_SCHEDULER\_SCHEDULES, 403  
 USER\_SCHEDULER\_STEP\_CTX, 404  
 USER\_SCHEDULER\_STEPS, 404  
 USER\_SEGMENTS, 404  
 USER\_SEQUENCES, 404  
 USER\_SOURCE, 405  
 USER\_STAT\_EXTENSIONS, 405  
 USER\_SUBPART\_KEY\_COLUMNS, 405  
 USER\_SUBPARTITION\_TEMPLATES, 405  
 USER\_SYNONYMS, 405  
 USER\_SYS\_PRIVS, 406  
 USER\_TAB\_COMMENTS, 409  
 USER\_TAB\_HISTGRM\_PENDING\_STATS, 409  
 USER\_TAB\_IDENTITY\_COLS, 410  
 USER\_TAB\_PENDING\_STATS, 410  
 USER\_TABLES, 406  
 USER\_TABLESPACES, 406  
 USER\_TBL\_COL\_STATISTICS, 411  
 USER\_TBL\_COLS, 410  
 USER\_TBL\_COLUMNS, 410  
 USER\_TBL\_HISTOGRAMS, 411  
 USER\_TBL\_PARTITIONS, 411  
 USER\_TBL\_PRIVS, 411  
 USER\_TBL\_PRIVS\_MADE, 411  
 USER\_TBL\_PRIVS\_RECD, 412  
 USER\_TRIGGER\_COLS, 412  
 USER\_TRIGGERS, 412

USER\_TYPE\_ATTRS, 413  
 USER\_TYPE\_METHODS, 413  
 USER\_TYPES, 413  
 USER\_UNUSED\_COL\_TABS, 413  
 USER\_UPDATABLE\_COLUMNS, 414  
 USER\_USERS, 414  
 USER\_VIEWS, 415  
 USGMT\_AUTO\_SHRINK\_INTERVAL, 185  
 USGMT\_ONLINE\_MAX, 185  
 USGMT\_ONLINE\_MIN, 186  
 USGMT\_STEAL\_IGNORE\_RETENTION, 186  
 UTL\_FILE\_DIR, 187

## V

V\$ACCESS, 424  
 V\$ACTIVE\_SESSION\_HISTORY, 425  
 V\$ALL\_FILESTAT, 427  
 V\$ALL\_LOG\_DEST, 428  
 V\$ARCHIVE\_DEST\_FILES, 429  
 V\$ARCHIVED\_LOG, 428  
 V\$AS\_ALIAS, 429  
 V\$AS\_DISK, 430  
 V\$AS\_DISK\_HISTORY, 432  
 V\$AS\_DISKSPACE, 431  
 V\$AS\_EXTENT\_MAP, 432  
 V\$AS\_FILE, 433  
 V\$AS\_OPERATION, 434  
 V\$AS\_USER, 434  
 V\$AS\_USERGROUP, 435  
 V\$AS\_USERGROUP\_MEMBER, 435  
 V\$AUTO\_COALESCE, 436  
 V\$BACKUP, 436  
 V\$BACKUP\_ARCHIVED\_LOG, 437  
 V\$BACKUP\_LIST, 437  
 V\$BACKUP\_LIST\_ARCHIVELOG, 438  
 V\$BACKUP\_LIST\_CONTROLFILE, 438  
 V\$BACKUP\_LIST\_DATAFILE, 438  
 V\$BACKUP\_SET, 439  
 V\$BACKUP\_SET\_TABLESPACE, 440  
 V\$BG\_SESSION, 440  
 V\$BG\_SESSTAT, 442  
 V\$BH, 443

V\$BLOCK\_CHANGE\_TRACKING, 444  
 V\$BLOCKER\_SESSION, 443  
 V\$BPARAMETERS, 444  
 V\$BUFFER\_CACHE\_ADVICE, 445  
 V\$BUFFER\_POOL, 445  
 V\$BUFFER\_POOL\_STATISTICS, 445  
 V\$CHECKPOINT, 446  
 V\$CLUSTER\_INTERCONNECTS, 447  
 V\$CONTEXT, 448  
 V\$CONTROLFILE, 448  
 V\$CORRUPT\_XID, 448  
 V\$DATABASE, 449  
 V\$DATAFILE, 451  
 V\$DATAFILE\_HEADER, 451  
 V\$DB\_PIPES, 454  
 V\$DBFILE, 452  
 V\$DBLINK, 453  
 V\$DBLINK\_STATUS, 453  
 V\$DML\_STATS, 454  
 V\$ENCRYPTED\_TABLESPACES, 455  
 V\$ENCRYPTION\_WALLET, 455  
 V\$ENQUEUE\_STAT, 455  
 V\$EST\_CHECKPOINT\_TIME, 456  
 V\$EVENT\_NAME, 456  
 V\$FILESTAT, 457  
 V\$FLASHBACK\_ARCHIVED\_LOG, 457  
 V\$FLASHBACK\_LOG, 458  
 V\$FLASHBACK\_MARKER, 459  
 V\$GLOBAL\_TRANSACTION, 459  
 V\$IM\_COLUMN\_LEVEL, 460  
 V\$IM\_SEGMENTS, 460  
 V\$INSTANCE, 462  
 V\$INSTANCEMETRIC, 463  
 V\$INSTANCEMETRIC\_HISTORY, 463  
 V\$INTERCONNECT\_LATENCY, 465  
 V\$INTERCONNECTION, 464  
 V\$IPARAM\_DESC, 465  
 V\$LATCH, 465  
 V\$LIBRARYCACHE, 466  
 V\$LICENSE, 467  
 V\$LISTENER\_PORT, 468  
 V\$LOCK, 468  
 V\$LOCKED\_OBJECT, 469

V\$LOG, 470  
 V\$LOG\_HISTORY, 471  
 V\$LOGFILE, 470  
 V\$MEDIA\_RECOVERY\_PROGRESS, 472  
 V\$MEM\_MGR, 473  
 V\$MEMORY\_TABLESPACE, 472  
 V\$METRIC, 473  
 V\$METRICNAME, 474  
 V\$MODIFIED\_PARAMETERS, 475  
 V\$MYSTAT, 475  
 V\$NLS\_PARAMETERS, 475  
 V\$OBJECT\_USAGE, 476  
 V\$OPEN\_CURSOR, 476  
 V\$OSSTAT, 477  
 V\$OSSTAT2, 477  
 V\$OUTLINE, 478  
 V\$PARAMETERS, 478  
 V\$PATCH, 478  
 V\$PE\_PESPLAN, 479  
 V\$PE\_PESSTAT, 479  
 V\$PE\_PROCESS, 479  
 V\$PE\_SESSION, 480  
 V\$PE\_SYSSTAT, 481  
 V\$PE\_TQSTAT, 481  
 V\$PGASTAT, 482  
 V\$PROC\_CONN\_INFO, 484  
 V\$PROCESS, 483  
 V\$PROXY\_SQL\_PARAM, 484  
 V\$PROXY\_SQLTEXT, 484  
 V\$RECOVER\_FILE, 487  
 V\$RECOVERY\_FILE\_STATUS, 485  
 V\$RECOVERY\_PROGRESS, 485  
 V\$RECOVERY\_STATUS, 486  
 V\$RESERVED\_WORDS, 487  
 V\$RESULT\_CACHE\_DEPENDENCY, 487  
 V\$RESULT\_CACHE\_OBJECT\_STATISTICS, 489  
 V\$RESULT\_CACHE\_OBJECTS, 488  
 V\$RESULT\_CACHE\_STATISTICS, 489  
 V\$ROLLSTAT, 490  
 V\$ROWCACHE, 491  
 V\$RSBSTAT, 491  
 V\$RSRC\_CONSUMER\_GROUP, 492  
 V\$RSRC\_PLAN, 493

V\$RSRC\_SESSION\_INFO, 493  
 V\$RUNNING\_JOBS, 494  
 V\$RUNNING\_JOBS\_WITH\_NAME, 494  
 V\$SEGMENT\_STATISTICS, 495  
 V\$SERVER\_EVENT\_STATUS, 495  
 V\$SERVER\_EVENT\_SUBTYPE, 495  
 V\$SERVER\_EVENT\_TYPE, 496  
 V\$SESS\_TIME\_MODEL, 504  
 V\$SESSION, 496  
 V\$SESSION\_EVENT, 501  
 V\$SESSION\_IO, 501  
 V\$SESSION\_LATEST\_STAT, 502  
 V\$SESSION\_LONGOPS, 502  
 V\$SESSION\_WAIT, 503  
 V\$SESSTAT, 504  
 V\$SGA, 505  
 V\$SGASTAT, 506  
 V\$SLOG, 506  
 V\$SPINLOCK\_HOLDER, 507  
 V\$SPINUSE\_STAT, 507  
 V\$SQL, 507  
 V\$SQL\_BIND\_CAPTURE, 516  
 V\$SQL\_BIND\_CAPTURE\_ALL, 516  
 V\$SQL\_EXPC\_INFO, 517  
 V\$SQL\_MONITOR, 518  
 V\$SQL\_PLAN, 521  
 V\$SQL\_PLAN\_MONITOR, 522  
 V\$SQL\_PLAN\_STATISTICS, 524  
 V\$SQL\_WORKAREA, 526  
 V\$SQL\_WORKAREA\_HISTOGRAM, 528  
 V\$SQLAREA, 509  
 V\$SQLSTATS, 511  
 V\$SQLTEXT, 513  
 V\$SQLTEXT\_WITH\_NEWLINES, 514  
 V\$SQLTEXT\_WITH\_NEWLINES2, 515  
 V\$SQLTEXT2, 514  
 V\$SSVR\_CLIENT, 528  
 V\$SSVR\_FLASHCACHE, 529  
 V\$SSVR\_GRID\_DISK, 529  
 V\$SSVR\_IOLOAD, 530  
 V\$SSVR\_IORM\_CATEGORY\_PLAN, 530  
 V\$SSVR\_IORM\_DB\_PLAN, 531  
 V\$SSVR\_IOSTAT, 531

V\$SSVR\_MEMSTAT, 532  
 V\$SSVR\_SLAB\_STAT, 532  
 V\$SSVR\_STORAGE\_DISK, 532  
 V\$STANDBY, 533  
 V\$STANDBY\_DEST, 534  
 V\$STANDBY\_LOG, 535  
 V\$STANDBY\_LOGFILE, 535  
 V\$STANDBY\_STATUS, 536  
 V\$STATNAME, 536  
 V\$SYS\_TIME\_MODEL, 538  
 V\$SYSSTAT, 537  
 V\$SYSSTAT\_VALUE, 537  
 V\$SYSTEM\_EVENT, 538  
 V\$SYSTEM\_PARAMETERS, 538  
 V\$TABLESPACE, 539  
 V\$TACSTAT, 539  
 V\$TEMPFILE, 540  
 V\$TEMPSEG\_OP\_USAGE, 540  
 V\$TEMPSEG\_USAGE, 541  
 V\$TEMPSTAT, 542  
 V\$THR\_ACT\_INFO, 544  
 V\$THR\_EVENT, 544  
 V\$THRINFO, 543  
 V\$TIMER, 545  
 V\$TIMEZONE\_NAMES, 545  
 V\$TRANSACTION, 546  
 V\$TSN\_TIME, 548  
 V\$UNDO\_FREE\_SPACE, 549  
 V\$UNDOSTAT, 548  
 V\$USGMT\_TXENTRY, 549  
 V\$VERSION, 550  
 V\$WAITER\_SESSION, 550  
 V\$WAITSTAT, 550  
 V\$WAITUSE\_STAT, 551  
 V\$WAITUSE\_THR\_STAT, 551

## W

WALLET\_FILE, 187  
 WTHR\_PER\_PEP, 187  
 WTHR\_PER\_PROC, 188  
 WTHR\_PROC\_CNT, 188  
 WTHR\_PROC\_CNT\_MAX, 188

## X

XA\_MAX\_BRANCH\_CNT, 189

XA\_TIMER\_INTERVAL, 189

## ㄱ

기반 테이블, 192

## ㄴ

데이터 사전, 191

동적 뷰, 193

## ㄷ

메타데이터, 191

## ㄹ

사용자 유형별 데이터 사전의 접근 권한, 191

## ㅇ

의미 분석 과정, 194

## ㅈ

정적 뷰, 193

정적 뷰의 구분, 193

질의 최적화기, 195

## ㅊ

특별한 목적의 DDL 문장, 197

