



Rutinas y vistas administrativas



Rutinas y vistas administrativas

Nota

Antes de utilizar esta información y el producto al que da soporte, lea la información general contenida en el Apéndice B, "Avisos", en la página 1201.

Nota de edición

Este manual es la traducción del original en inglés *IBM DB2 9.7 for Linux, UNIX, and Windows Administrative Routines and Views* (SC27-2436-00).

Este documento contiene información propiedad de IBM. Se proporciona según un acuerdo de licencia y está protegido por la ley de la propiedad intelectual. La información contenida en esta publicación no incluye ninguna garantía de producto, por lo que ninguna declaración proporcionada en este manual deberá interpretarse como tal.

Puede realizar pedidos de publicaciones de IBM en línea o a través del representante de IBM de su localidad.

- Para realizar pedidos en línea, vaya a IBM Publications Center ubicado en el sitio web www.ibm.com/shop/publications/order
- Para encontrar al representante de IBM de su localidad, vaya al IBM Directory of Worldwide Contacts en el sitio web www.ibm.com/planetwide

Para realizar pedidos de publicaciones de DB2 desde DB2 Marketing and Sales, en los EE.UU. o en Canadá, llame al 1-800-IBM-4YOU (426-4968).

Cuando envía información a IBM, otorga a IBM un derecho no exclusivo a utilizar o distribuir dicha información en la forma en que IBM considere adecuada, sin contraer por ello ninguna obligación con el remitente.

© Copyright International Business Machines Corporation 2006, 2009.

Contenido

Capítulo 1. Vistas y rutinas definidas por el sistema 1

| | |
|---|---|
| Autorización para vistas administrativas | 1 |
| Vistas administrativas comparadas con funciones de tablas | 1 |

Capítulo 2. Rutinas y vistas de SQL definidas por el sistema soportadas . . . 3

Capítulo 3. Rutinas del supervisor de actividades 21

| | |
|---|----|
| AM_BASE_RPT_RECOMS – Recomendaciones para informes de actividad | 21 |
| AM_BASE_RPTS – Informes del supervisor de actividad | 22 |
| RAM_DROP_TASK – Supresión de una tarea de supervisión | 23 |
| RAM_GET_LOCK_CHN_TB – Recuperación de los datos de la cadena de bloqueo de la aplicación en formato tabular | 24 |
| RAM_GET_LOCK_CHNS – Recuperación de información de bloqueo de cadenas para una aplicación específica | 25 |
| RAM_GET_LOCK_RPT – Recuperación de los detalles del bloqueo de la aplicación | 25 |
| RAM_GET_RPT – Recuperación de datos del supervisor de actividad | 33 |
| RAM_SAVE_TASK – Creación o modificación de una tarea de supervisión | 34 |

Capítulo 4. Procedimiento ADMIN_CMD y rutinas asociadas 37

| | |
|--|----|
| ADMIN_CMD – Ejecutar mandatos administrativos | 37 |
| Mandato ADD CONTACT utilizando el procedimiento ADMIN_CMD | 39 |
| Mandato ADD CONTACTGROUP utilizando el procedimiento ADMIN_CMD | 41 |
| Mandato AUTOCONFIGURE utilizando el procedimiento ADMIN_CMD | 42 |
| Mandato BACKUP DATABASE utilizando el procedimiento ADMIN_CMD | 47 |
| Mandato DESCRIBE utilizando el procedimiento ADMIN_CMD | 55 |
| Mandato DROP CONTACT utilizando el procedimiento ADMIN_CMD | 69 |
| Mandato DROP CONTACTGROUP utilizando el procedimiento ADMIN_CMD | 70 |
| Mandato EXPORT utilizando el procedimiento ADMIN_CMD | 71 |
| Mandato FORCE APPLICATION utilizando el procedimiento ADMIN_CMD | 83 |
| Mandato GET STMM TUNING DBPARTITIONNUM mediante el procedimiento ADMIN_CMD | 85 |

| | |
|---|-----|
| Mandato IMPORT utilizando el procedimiento ADMIN_CMD | 86 |
| Mandato INITIALIZE TAPE utilizando el procedimiento ADMIN_CMD | 114 |
| Mandato LOAD utilizando el procedimiento ADMIN_CMD | 115 |
| Mandato PRUNE HISTORY/LOGFILE utilizando el procedimiento ADMIN_CMD | 158 |
| Mandato QUIESCE DATABASE utilizando el procedimiento ADMIN_CMD | 160 |
| Mandato QUIESCE TABLESPACES FOR TABLE utilizando el procedimiento ADMIN_CMD | 162 |
| Mandato REDISTRIBUTE DATABASE PARTITION GROUP utilizando el procedimiento ADMIN_CMD | 165 |
| Mandato REORG INDEXES/TABLE utilizando el procedimiento ADMIN_CMD | 176 |
| Mandato RESET ALERT CONFIGURATION utilizando el procedimiento ADMIN_CMD | 190 |
| Mandato RESET DATABASE CONFIGURATION utilizando el procedimiento ADMIN_CMD | 192 |
| Mandato RESET DATABASE MANAGER CONFIGURATION utilizando el procedimiento ADMIN_CMD | 194 |
| Mandato REWIND TAPE utilizando el procedimiento ADMIN_CMD | 195 |
| Mandato RUNSTATS utilizando el procedimiento ADMIN_CMD | 196 |
| Mandato SET TAPE POSITION utilizando el procedimiento ADMIN_CMD | 208 |
| Mandato UNQUIESCE DATABASE utilizando el procedimiento ADMIN_CMD | 209 |
| Mandato UPDATE ALERT CONFIGURATION utilizando el procedimiento ADMIN_CMD | 210 |
| Mandato UPDATE CONTACT utilizando el procedimiento ADMIN_CMD | 216 |
| Mandato UPDATE CONTACTGROUP utilizando el procedimiento ADMIN_CMD | 217 |
| Mandato UPDATE DATABASE CONFIGURATION utilizando el procedimiento ADMIN_CMD | 218 |
| Mandato UPDATE DATABASE MANAGER CONFIGURATION utilizando el procedimiento ADMIN_CMD | 222 |
| Mandato UPDATE HEALTH NOTIFICATION CONTACT LIST utilizando el procedimiento ADMIN_CMD | 224 |
| Mandato UPDATE HISTORY utilizando el procedimiento ADMIN_CMD | 225 |
| Mandato UPDATE STMM TUNING DBPARTITIONNUM mediante el procedimiento ADMIN_CMD | 227 |
| Función ADMIN_EST_INLINE_LENGTH - Estimar la longitud necesaria en datos en línea | 228 |

| | |
|---|-----|
| Función de tabla ADMIN_GET_DBP_MEM_USAGE - Obtener el consumo de memoria total por instancia | 230 |
| Función de tabla ADMIN_GET_INDEX_COMPRESS_INFO - Devolver información de índice de compresión | 232 |
| Función de tabla ADMIN_GET_INDEX_INFO - Devolver información de índice | 236 |
| Función de tabla ADMIN_GET_MSGS - Recuperar mensajes generados por un programa de utilidad de movimiento de datos que utiliza el procedimiento ADMIN_CMD | 239 |
| Función ADMIN_IS_INLINED - Determinar si los datos están en línea | 241 |
| Procedimiento ADMIN_REMOVE_MSGS - Limpiar los mensaje que genera un programa de utilidad de movimiento de datos que utiliza el procedimiento ADMIN_CMD | 242 |
| Procedimiento ADMIN_REVALIDATE_DB_OBJECTS - Revalidar objetos de base de datos no válidos | 243 |
| Vista administrativa ADMINTABCOMPRESSINFO y función de tabla ADMIN_GET_TAB_COMPRESS_INFO_V97 - Devolver información de compresión | 246 |
| Vista administrativa ADMINTABINFO y función de tabla ADMIN_GET_TAB_INFO_V97 - Recuperar información de tamaño y estado | 254 |
| Vista administrativa ADMINTEMPCOLUMNS y función de tabla ADMIN_GET_TEMP_COLUMNS - Recuperar información de columna para tablas temporales | 262 |
| Vista administrativa ADMINTEMPTABLES y función de tabla ADMIN_GET_TEMP_TABLES - Recuperar información para tablas temporales | 266 |

Capítulo 5. Vistas y rutinas del planificador de tareas administrativas. 271

| | |
|--|-----|
| Procedimiento ADMIN_TASK_ADD - Planificar una nueva tarea | 271 |
| Formato cron de UNIX | 274 |
| Vista administrativa ADMIN_TASK_LIST - Recuperar información sobre las tareas en el planificador | 276 |
| Procedimiento ADMIN_TASK_REMOVE - Eliminar registros de tareas planificadas o de estados de tareas | 278 |
| Vista administrativa ADMIN_TASK_STATUS - Recuperar información de estado de tarea | 279 |
| Procedimiento ADMIN_TASK_UPDATE - Actualizar una tarea existente | 281 |

Capítulo 6. Rutinas y procedimientos de auditoría 285

| | |
|---|-----|
| Procedimiento y función de tabla AUDIT_ARCHIVE - Archivar archivo de registro de comprobación | 285 |
| AUDIT_DELIM_EXTRACT - Realizar extracción en archivo delimitado | 286 |

| | |
|---|-----|
| Función de tabla AUDIT_LIST_LOGS - Enumera los archivos de anotaciones cronológicas de auditoría archivados | 287 |
|---|-----|

Capítulo 7. Rutinas de mantenimiento automático 289

| | |
|--|-----|
| Procedimiento AUTOMAINT_GET_POLICY - recuperar la política de mantenimiento automático | 289 |
| Procedimiento AUTOMAINT_GET_POLICYFILE - recuperar la política de mantenimiento automático | 290 |
| Procedimiento AUTOMAINT_SET_POLICY - Configurar política de mantenimiento automático | 291 |
| Procedimiento AUTOMAINT_SET_POLICYFILE - configurar la política de mantenimiento automático | 292 |

Capítulo 8. Procedimientos de la API de SQL común. 295

| | |
|---|-----|
| Parámetros de entrada y salida comunes | 295 |
| Creación de versiones de documentos XML | 296 |
| Documentos de entrada XML | 297 |
| Modalidad completa para la devolución de documentos de entrada XML válidos | 298 |
| Documentos de salida XML | 298 |
| Expresiones XPath para el filtrado de salidas | 299 |
| Documentos de mensaje XML | 300 |
| Procedimiento CANCEL_WORK - Cancelar trabajo | 301 |
| Procedimiento GET_CONFIG - Obtener datos de configuración | 308 |
| Procedimiento GET_MESSAGE - Obtener texto de mensaje | 315 |
| Procedimiento GET_SYSTEM_INFO - Obtener información de sistema | 322 |
| Procedimiento SET_CONFIG - Establecer parámetros de configuración | 329 |

Capítulo 9. Rutinas y vistas de configuración 341

| | |
|--|-----|
| DB_PARTITIONS | 341 |
| Vista administrativa DBCFG - Recuperar información sobre los parámetros de configuración de la base de datos | 342 |
| Vista administrativa DBMCFG - Recuperar información sobre los parámetros de configuración del gestor de bases de datos | 344 |
| Vista administrativa REG_VARIABLES - Recuperar valores de registro de DB2 en uso | 347 |

Capítulo 10. Vistas del entorno. 349

| | |
|--|-----|
| Vista administrativa ENV_FEATURE_INFO - Devolver información de licencia para las características de DB2 | 349 |
| Vista administrativa ENV_INST_INFO - Recuperar información sobre la instancia actual | 350 |
| Vista administrativa ENV_PROD_INFO - Recuperar información sobre productos DB2 instalados | 352 |
| Vista administrativa ENV_SYS_INFO - Recuperar información sobre el sistema | 353 |

Vista administrativa ENV_SYS_RESOURCES -
Devolver información de sistema 354

Capítulo 11. Rutinas del supervisor 359

Procedimiento
EVMON_FORMAT_UE_TO_TABLES - Trasladar un documento XML a tablas relacionales 360
Función de tabla EVMON_FORMAT_UE_TO_XML - Convertir sucesos sin formatear a XML 367
Función de tabla MON_GET_ACTIVITY_DETAILS - Obtener detalles completos sobre la actividad . . . 370
Función de tabla MON_GET_BUFFERPOOL - Obtener métrica de agrupación de almacenamientos intermedios 379
Función de tabla MON_GET_CONNECTION - Obtener métrica de conexión 383
Función de tabla MON_GET_CONNECTION_DETAILS - Obtener métrica de conexión detallada 389
Función de tabla MON_GET_CONTAINER - Obtener métrica del contenedor de espacio de tablas 395
MON_GET_EXTENT_MOVEMENT_STATUS - obtener el progreso del traslado de extensiones . . 398
Función de tabla MON_GET_INDEX - Obtener métrica de índice 400
Función de tabla MON_GET_PKG_CACHE_STMT - Obtener métrica de actividad de sentencia de SQL en la antememoria de paquetes. 402
Función de tabla MON_GET_SERVICE_SUBCLASS - Obtener métrica de subclase de servicio 408
Función de tabla MON_GET_SERVICE_SUBCLASS_DETAILS - Obtener métrica detallada de subclase de servicio . 415
Función de tabla MON_GET_TABLE - Obtener métrica de tabla 422
Función de tabla MON_GET_TABLESPACE - Obtener métrica de espacio de tablas 425
Función de tabla MON_GET_UNIT_OF_WORK - Obtener métrica de unidad de trabajo 429
Función de tabla MON_GET_UNIT_OF_WORK_DETAILS - Obtener métrica detallada de unidad de trabajo 435
Función de tabla MON_GET_WORKLOAD - Obtener métrica de carga de trabajo 443
Función de tabla MON_GET_WORKLOAD_DETAILS - Obtener métrica detallada de carga de trabajo 448

Capítulo 12. Rutinas de MQSeries . . 457

MQPUBLISH 457
MQREAD 458
MQREADALL 459
MQREADALLCLOB 461
MQREADCLOB 463
MQRECEIVE 464
MQRECEIVEALL 466
MQRECEIVEALLCLOB 468
MQRECEIVECLOB 470
MQSEND 472

MQSUBSCRIBE 473
MQUNSUBSCRIBE 474

Capítulo 13. Rutinas y vistas de seguridad 477

AUTH_LIST_AUTHORITIES_FOR_AUTHID . . . 477
Función de tabla AUTH_LIST_GROUPS_FOR_AUTHID - Recuperación lista de pertenencia al grupo para un ID de autorización determinado 480
Función AUTH_LIST_ROLES_FOR_AUTHID - Devuelve la lista de roles 482
Vista administrativa AUTHORIZATIONIDS - Recuperar ID y tipos de autorización 484
Vista administrativa OBJECTOWNERS - Recuperar información sobre propiedad de objetos 485
Vista administrativa PRIVILEGES - Recuperar información sobre privilegios 486

Capítulo 14. Rutinas y vistas de instantáneas. 489

Vista administrativa APPL_PERFORMANCE - Recuperar porcentaje de filas seleccionadas para una aplicación 489
Vista administrativa APPLICATIONS - Recuperar información de aplicación de base de datos conectada 490
Vista administrativa BP_HITRATIO - Recuperar información sobre la tasa de aciertos de agrupación de almacenamientos intermedios 494
Vista administrativa BP_READ_IO - Recuperar información de rendimiento de lectura de las agrupaciones de almacenamientos intermedios . . 497
Vista administrativa BP_WRITE_IO - Recuperar información de rendimiento de grabación de las agrupaciones de almacenamientos intermedios . . 499
Vista administrativa CONTAINER_UTILIZATION - Recuperar información de utilización y contenedor de espacio de tablas 501
Vista administrativa LOCKS_HELD - Recuperar información acerca de los bloqueos retenidos. . . 503
Vista administrativa LOCKWAITS - Recuperar información de bloqueos actuales en espera de ser activados. 506
Vista administrativa LOG_UTILIZATION - Recuperar información de utilización de las anotaciones cronológicas 510
Vista administrativa LONG_RUNNING_SQL . . . 511
Vista administrativa QUERY_PREP_COST - Recuperar información de tiempo de preparación de sentencias 514
Vista administrativa SNAPAGENT y función de tabla SNAP_GET_AGENT - Recuperar información de instantáneas de la aplicación del grupo de datos lógicos agent 515
Vista administrativa SNAPAGENT_MEMORY_POOL y función de tabla SNAP_GET_AGENT_MEMORY_POOL - Recuperar información de instantáneas del grupo de datos lógicos memory_pool 518

| | |
|--|-----|
| Vista administrativa SNAPAPPL_INFO y función de tabla SNAP_GET_APPL_INFO_V95 – Recuperar información de instantáneas del grupo de datos lógicos appl_info | 523 |
| Vista administrativa SNAPAPPL y función de tabla SNAP_GET_APPL_V95 – Recuperar información de instantáneas del grupo de datos lógicos appl | 530 |
| Vista administrativa SNAPBP y función de tabla SNAP_GET_BP_V95 – Recuperar información de instantáneas del grupo lógico bufferpool | 540 |
| Vista administrativa SNAPBP_PART y función de tabla SNAP_GET_BP_PART – Recuperar información de instantáneas del grupo de datos lógicos bufferpool_nodeinfo | 546 |
| Vista administrativa SNAPCONTAINER y función de tabla SNAP_GET_CONTAINER_V91 – Recuperar información acerca de la instantánea de grupo de datos lógicos tablespace_container | 549 |
| Vista administrativa SNAPDB y función de tabla SNAP_GET_DB_V95 – Recuperar información de instantánea desde el grupo lógico dbase | 554 |
| Vista administrativa SNAPDB_MEMORY_POOL y función de tabla SNAP_GET_DB_MEMORY_POOL – Recuperar información sobre el uso de la memoria en el nivel de la base de datos | 567 |
| Vista administrativa SNAPDBM y función de tabla SNAP_GET_DBM_V95 – Recuperar la información de instantáneas del grupo de datos lógicos dbm | 572 |
| Vista administrativa SNAPDBM_MEMORY_POOL y función de tabla SNAP_GET_DBM_MEMORY_POOL – Recuperar información sobre el uso de la memoria en el nivel del gestor de bases de datos | 576 |
| Vista administrativa SNAPDETAILLOG y función de tabla SNAP_GET_DETAILLOG_V91 - Recuperar información de instantánea del grupo de datos lógicos detail_log | 580 |
| Vista administrativa SNAPDYN_SQL y función de tabla SNAP_GET_DYN_SQL_V95 – Recuperar información de instantáneas del grupo de datos lógicos dynsql | 583 |
| Vista administrativa SNAPFCM y función de tabla SNAP_GET_FCM – Recuperar la información de instantáneas del grupo de datos lógicos fcm | 589 |
| Vista administrativa SNAPFCM_PART y función de tabla SNAP_GET_FCM_PART – Recuperar la información de instantáneas del grupo de datos lógicos fcm_node | 591 |
| Vista administrativa SNAPHADR y función de tabla SNAP_GET_HADR – Recuperar información de instantáneas del grupo de datos lógicos hadr | 594 |
| Vista administrativa SNAPLOCK y función de tabla SNAP_GET_LOCK – Recuperar información de instantáneas del grupo de datos lógicos lock | 599 |
| Vista administrativa SNAPLOCKWAIT y función de tabla SNAP_GET_LOCKWAIT – Recuperar información de instantáneas del grupo de datos lógicos lockwait | 604 |
| Vista administrativa SNAPSTMT y función de tabla SNAP_GET_STMT – Recuperar información de instantáneas de sentencias | 611 |

| | |
|---|-----|
| Vista administrativa SNAPSTORAGE_PATHS y función de tabla SNAP_GET_STORAGE_PATHS_V97 - Recuperar información de vía de acceso de almacenamiento automático | 618 |
| Vista administrativa SNAPSUBSECTION y función de tabla SNAP_GET_SUBSECTION – Recuperar información de instantáneas del grupo de supervisor lógico de la subsección | 621 |
| Vista administrativa SNAPSWITCHES y función de tabla SNAP_GET_SWITCHES - Recuperar información de estado de conmutador de instantáneas de base de datos | 625 |
| Vista administrativa SNAPTAB y función de tabla SNAP_GET_TAB_V91 – Recuperar información de instantáneas del grupo de datos lógicos table | 628 |
| Vista administrativa SNAPTAB_REORG y función de tabla SNAP_GET_TAB_REORG - Recuperar información sobre instantáneas de reorganización de tabla | 632 |
| Vista administrativa SNAPTbsp y función de tabla SNAP_GET_TBSP_V91 – Recuperar información de instantáneas del grupo de datos lógicos de espacio de tablas | 639 |
| Vista administrativa SNAPTbsp_PART y función de tabla SNAP_GET_TBSP_PART_V97 – Recuperar información de instantáneas del grupo de datos lógicos tablespace_nodeinfo | 645 |
| Vista administrativa SNAPTbsp_QUIESCER y función de tabla SNAP_GET_TBSP_QUIESCER – Recuperar información acerca de la instantánea de espacio de tablas de inmovilizador | 650 |
| Vista administrativa SNAPTbsp_RANGE y función de tabla SNAP_GET_TBSP_RANGE – Recuperar información acerca de la instantánea de rango | 654 |
| Vista administrativa SNAPUTIL y función de tabla SNAP_GET_UTIL - Recuperar información de instantánea de grupo de datos lógicos utility_info | 658 |
| Vista administrativa SNAPUTIL_PROGRESS y función de tabla SNAP_GET_UTIL_PROGRESS - Recuperar información de instantánea de grupo de datos lógicos de progreso | 662 |
| Procedimiento SNAP_WRITE_FILE | 666 |
| Vista administrativa SNAPAGENT y función de tabla SNAP_GET_AGENT – Recuperar información de instantáneas de la aplicación del grupo de datos lógicos agent | 667 |
| Vista administrativa SNAPAGENT_MEMORY_POOL y función de tabla SNAP_GET_AGENT_MEMORY_POOL – Recuperar información de instantáneas del grupo de datos lógicos memory_pool | 670 |
| Vista administrativa SNAPAPPL_INFO y función de tabla SNAP_GET_APPL_INFO_V95 – Recuperar información de instantáneas del grupo de datos lógicos appl_info | 674 |
| Vista administrativa SNAPAPPL y función de tabla SNAP_GET_APPL_V95 – Recuperar información de instantáneas del grupo de datos lógicos appl | 682 |

| | |
|--|-----|
| Vista administrativa SNAPBP y función de tabla SNAP_GET_BP_V95 – Recuperar información de instantáneas del grupo lógico bufferpool | 692 |
| Vista administrativa SNAPBP_PART y función de tabla SNAP_GET_BP_PART – Recuperar información de instantáneas del grupo de datos lógicos bufferpool_nodeinfo | 698 |
| Vista administrativa SNAPCONTAINER y función de tabla SNAP_GET_CONTAINER_V91 – Recuperar información acerca de la instantánea de grupo de datos lógicos tablespace_container | 701 |
| Vista administrativa SNAPDB y función de tabla SNAP_GET_DB_V95 – Recuperar información de instantánea desde el grupo lógico dbase | 706 |
| Vista administrativa SNAPDB_MEMORY_POOL y función de tabla SNAP_GET_DB_MEMORY_POOL – Recuperar información sobre el uso de la memoria en el nivel de la base de datos | 719 |
| Vista administrativa SNAPDBM y función de tabla SNAP_GET_DBM_V95 – Recuperar la información de instantáneas del grupo de datos lógicos dbm. | 724 |
| Vista administrativa SNAPDBM_MEMORY_POOL y función de tabla SNAP_GET_DBM_MEMORY_POOL – Recuperar información sobre el uso de la memoria en el nivel del gestor de bases de datos | 728 |
| Vista administrativa SNAPDETAILLOG y función de tabla SNAP_GET_DETAILLOG_V91 - Recuperar información de instantánea del grupo de datos lógicos detail_log | 732 |
| Vista administrativa SNAPDYN_SQL y función de tabla SNAP_GET_DYN_SQL_V95 – Recuperar información de instantáneas del grupo de datos lógicos dynsql | 735 |
| Vista administrativa SNAPFCM y función de tabla SNAP_GET_FCM – Recuperar la información de instantáneas del grupo de datos lógicos fcm | 741 |
| Vista administrativa SNAPFCM_PART y función de tabla SNAP_GET_FCM_PART – Recuperar la información de instantáneas del grupo de datos lógicos fcm_node | 743 |
| Vista administrativa SNAPHADR y función de tabla SNAP_GET_HADR – Recuperar información de instantáneas del grupo de datos lógicos hadr. | 746 |
| Vista administrativa SNAPLOCK y función de tabla SNAP_GET_LOCK – Recuperar información de instantáneas del grupo de datos lógicos lock | 751 |
| Vista administrativa SNAPLOCKWAIT y función de tabla SNAP_GET_LOCKWAIT – Recuperar información de instantáneas del grupo de datos lógicos lockwait | 756 |
| Vista administrativa SNAPSTMT y función de tabla SNAP_GET_STMT – Recuperar información de instantáneas de sentencias | 763 |
| Vista administrativa SNAPSTORAGE_PATHS y función de tabla SNAP_GET_STORAGE_PATHS – Recuperar información de vía de acceso del almacenamiento automático | 770 |

| | |
|---|-----|
| Vista administrativa SNAPSUBSECTION y función de tabla SNAP_GET_SUBSECTION – Recuperar información de instantáneas del grupo de supervisor lógico de la subsección | 773 |
| Vista administrativa SNAPSWITCHES y función de tabla SNAP_GET_SWITCHES - Recuperar información de estado de conmutador de instantáneas de base de datos | 777 |
| Vista administrativa SNAPTAB y función de tabla SNAP_GET_TAB_V91 – Recuperar información de instantáneas del grupo de datos lógicos table. | 781 |
| Vista administrativa SNAPTAB_REORG y función de tabla SNAP_GET_TAB_REORG - Recuperar información sobre instantáneas de reorganización de tabla | 784 |
| Vista administrativa SNAPTbsp y función de tabla SNAP_GET_TBSP_V91 – Recuperar información de instantáneas del grupo de datos lógicos de espacio de tablas | 791 |
| Vista administrativa SNAPTbsp_PART y función de tabla SNAP_GET_TBSP_PART_V91 – Recuperar información de instantáneas del grupo de datos lógicos tablespace_nodeinfo | 797 |
| Vista administrativa SNAPTbsp QUIESCER y función de tabla SNAP_GET_TBSP QUIESCER – Recuperar información acerca de la instantánea de espacio de tablas de inmovilizador | 803 |
| Vista administrativa SNAPTbsp_RANGE y función de tabla SNAP_GET_TBSP_RANGE – Recuperar información acerca de la instantánea de rango | 807 |
| Vista administrativa SNAPUTIL y función de tabla SNAP_GET_UTIL - Recuperar información de instantánea de grupo de datos lógicos utility_info | 811 |
| Vista administrativa SNAPUTIL_PROGRESS y función de tabla SNAP_GET_UTIL_PROGRESS - Recuperar información de instantánea de grupo de datos lógicos de progreso | 815 |
| Procedimiento SNAP_WRITE_FILE | 819 |
| Vista administrativa TBSP_UTILIZATION - Recuperar información de configuración y utilización de espacio de tablas | 820 |
| Vista administrativa TOP_DYNAMIC_SQL – Recuperar información acerca de las sentencias de SQL dinámico más frecuentes | 824 |

Capítulo 15. Rutinas de procedimientos de SQL 827

| | |
|---|-----|
| Procedimiento ALTER_ROUTINE_PACKAGE | 827 |
| RGET_ROUTINE_OPTS | 828 |
| GET_ROUTINE_SAR. | 828 |
| PUT_ROUTINE_SAR. | 829 |
| Procedimiento REBIND_ROUTINE_PACKAGE - volver a vincular un paquete | 831 |
| RSET_ROUTINE_OPTS | 833 |

Capítulo 16. Rutinas de redistribución paso a paso 835

| | |
|---|-----|
| Procedimiento ANALYZE_LOG_SPACE - Recuperar información de análisis de espacio de anotaciones | 835 |
|---|-----|

| | |
|---|-----|
| Procedimiento GENERATE_DISTFILE - Generar archivo de distribución de datos | 837 |
| Procedimiento GET_SWRD_SETTINGS - Recuperar información de redistribución | 838 |
| Procedimiento SET_SWRD_SETTINGS - Crear o cambiar registro de redistribución | 841 |
| Procedimiento STEPWISE_REDISTRIBUTE_DBPG - Redistribuir parte del grupo de particiones de base de datos | 843 |

Capítulo 17. Rutinas de la herramienta de gestión de almacenamiento 845

| | |
|---|-----|
| Procedimiento CAPTURE_STORAGEMGMT_INFO - Recuperar información relacionada con el almacenamiento para un objeto raíz concreto | 845 |
| Procedimiento CREATE_STORAGEMGMT_TABLES - Crear tablas de gestión de almacenamiento. | 846 |
| Procedimiento DROP_STORAGEMGMT_TABLES - Descartar todas las tablas de gestión de almacenamiento | 847 |

Capítulo 18. Rutinas de búsqueda de texto 849

| | |
|---|-----|
| Procedimiento almacenado SYSTS_ADMIN_CMD - Ejecutar mandatos de administración de búsqueda de texto | 849 |
| Procedimiento SYSTS_ALTER - Cambiar las características de actualización de un índice | 850 |
| Procedimiento SYSTS_CLEAR_COMMANDLOCKS - Eliminar bloqueos de mandato para índices de búsqueda de texto. | 854 |
| Procedimiento SYSTS_CLEAR_EVENTS - Suprimir los sucesos de indexación de una tabla de sucesos de índice | 856 |
| Procedimiento SYSTS_CREATE - Crear un índice de búsqueda de texto en una columna | 858 |
| Procedimiento SYSTS_DISABLE - Inhabilitar base de datos actual para búsqueda de textos | 865 |
| Procedimiento SYSTS_DROP - Descartar un índice de búsqueda de texto. | 867 |
| Procedimiento SYSTS_ENABLE - Habilitar base de datos actual para búsqueda de texto. | 870 |
| Procedimiento SYSTS_UPDATE - Actualizar el índice de búsqueda de texto | 871 |

Capítulo 19. Rutinas de gestión de cargas de trabajo. 875

| | |
|--|-----|
| WLM_CANCEL_ACTIVITY - Cancelar una actividad | 875 |
| WLM_CAPTURE_ACTIVITY_IN_PROGRESS - Recopilar información para supervisor de sucesos de actividades | 876 |
| WLM_COLLECT_STATS - Coleccionar y restaurar estadísticas de gestión de carga de trabajo. | 878 |
| Función de tabla WLM_GET_QUEUE_STATS - Devolver estadísticas de cola de umbral | 879 |

| | |
|---|-----|
| Función de tabla WLM_GET_SERVICE_CLASS_AGENTS_V97 - Listar los agentes que se ejecutan en una clase de servicio | 883 |
| WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES_V97 - Lista de apariciones de carga de trabajo | 890 |
| Función de tabla WLM_GET_SERVICE_SUBCLASS_STATS_V97 - Devolver estadísticas de subclases de servicio | 895 |
| WLM_GET_SERVICE_SUPERCLASS_STATS - Devolver estadísticas de superclases de servicio | 904 |
| WLM_GET_WORK_ACTION_SET_STATS - Devolver estadísticas de conjunto de acciones de trabajo. | 906 |
| WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES_V97 - Devolver una lista de actividades | 908 |
| Función de tabla WLM_GET_WORKLOAD_STATS_V97 - Devolver estadísticas de carga de trabajo | 913 |
| Procedimiento WLM_SET_CLIENT_INFO - Establecer información de cliente. | 917 |

Capítulo 20. Rutinas y vistas varias 921

| | |
|---|-----|
| Procedimiento ADMIN_COPY_SCHEMA - Copiar un esquema específico y sus objetos | 921 |
| Procedimiento ADMIN_DROP_SCHEMA - Descartar un esquema específico y sus objetos | 925 |
| Procedimiento ADMIN_MOVE_TABLE - trasladar una tabla en línea | 927 |
| Procedimiento ADMIN_MOVE_TABLE_UTIL - Modificar el procedimiento de trasladar tablas en línea | 939 |
| ALTOBJ | 941 |
| APPLICATION_ID | 944 |
| Función de tabla COMPILATION_ENV - Recuperar elementos del entorno de compilación | 945 |
| Vista administrativa CONTACTGROUPS - Recuperar la lista de grupos de contacto | 947 |
| Vista administrativa CONTACTS - Recuperar lista de contactos. | 948 |
| Vista administrativa DB_HISTORY - Recuperar información de archivo histórico | 949 |
| Vista administrativa DBPATHS - Recuperar vías de acceso de bases de datos | 954 |
| EXPLAIN_FORMAT_STATS | 958 |
| EXPLAIN_GET_MSGS | 962 |
| GET_DBSIZE_INFO | 965 |
| Vista administrativa NOTIFICATIONLIST - Recuperar lista de contactos para notificación de salud | 967 |
| PD_GET_DIAG_HIST - Devolver registros de un determinado recurso | 968 |
| Vista administrativa PDLOGMSG_LAST24HOURS y función de tabla PD_GET_LOG_MSGS - Recuperar mensajes de determinación de problemas | 975 |
| Procedimiento REORGCHK_IX_STATS - Recuperar estadísticas de índice para evaluación de la reorganización | 982 |

| | |
|--|-----|
| Procedimiento REORGCHK_TB_STATS - Recuperar estadísticas de tabla para evaluación de la reorganización | 984 |
| Funciones escalares SQLERRM – Recuperar información de mensajes de error | 986 |
| SYSINSTALLOBJECTS | 989 |

Capítulo 21. Rutinas administrativas de SQL obsoletas y sus vistas o rutinas de sustitución. 991

| | |
|---|------|
| Tabla de función ADMIN_GET_TAB_INFO - Recuperar información de tamaño y estado para las tablas | 993 |
| Vista ADMIN_TABCOMPRESSINFO y ADMIN_GET_TAB_COMPRESS_INFO | 1001 |
| RGET_DB_CONFIG | 1006 |
| GET_DBM_CONFIG | 1008 |
| Rutinas de instantáneas de salud | 1009 |
| EALTH_CONT_HI | 1009 |
| HEALTH_CONT_HI_HIS | 1010 |
| HEALTH_CONT_INFO | 1012 |
| HEALTH_DB_HI | 1014 |
| HEALTH_DB_HI_HIS | 1017 |
| HEALTH_DB_HIC | 1021 |
| HEALTH_DB_HIC_HIS | 1023 |
| HEALTH_DB_INFO | 1026 |
| HEALTH_DBM_HI | 1027 |
| HEALTH_DBM_HI_HIS | 1029 |
| HEALTH_DBM_INFO | 1031 |
| HEALTH_GET_ALERT_ACTION_CFG | 1032 |
| HEALTH_GET_ALERT_CFG | 1036 |
| HEALTH_GET_IND_DEFINITION | 1039 |
| RHEALTH_HI_REC | 1041 |
| HEALTH_TBS_HI | 1043 |
| HEALTH_TBS_HI_HIS | 1045 |
| HEALTH_TBS_INFO | 1049 |
| Función de tabla SNAP_GET_APPL – Recuperar información de instantáneas del grupo de datos lógico appl | 1051 |
| Función de tabla SNAP_GET_APPL_INFO – Recuperación de información de instantáneas del grupo de datos lógico appl_info | 1059 |
| Función de tabla SNAP_GET_BP – Recuperación de la información de instantáneas del grupo de datos lógico bufferpool | 1065 |
| SNAP_GET_CONTAINER | 1070 |
| SNAP_GET_DB | 1071 |
| Función de tabla SNAP_GET_DBM – Recuperación de la información de instantáneas del grupo de datos lógicos dbm | 1079 |
| Función de tabla SNAP_GET_DB_V91 - Recuperar información de instantánea desde el grupo lógico dbase | 1083 |
| Función de tabla SNAP_GET_DYN_SQL_V91 - Recuperar información de la instantánea del grupo de datos lógicos dynsql | 1095 |
| SNAP_GET_DYN_SQL | 1099 |
| SNAP_GET_STO_PATHS | 1101 |
| SNAP_GET_TAB | 1102 |
| SNAP_GET_TBSP | 1103 |

| | |
|--|------|
| SNAP_GET_TBSP_PART | 1107 |
| SNAPSHOT_AGENT | 1109 |
| SNAPSHOT_APPL | 1110 |
| SNAPSHOT_APPL_INFO | 1116 |
| SNAPSHOT_BP | 1118 |
| SNAPSHOT_CONTAINER | 1121 |
| SNAPSHOT_DATABASE | 1122 |
| SNAPSHOT_DBM | 1129 |
| SNAPSHOT_DYN_SQL | 1131 |
| SNAPSHOT_FCM | 1133 |
| SNAPSHOT_FCMNODE | 1134 |
| SNAPSHOT_FILEW | 1135 |
| SNAPSHOT_LOCK | 1136 |
| SNAPSHOT_LOCKWAIT | 1137 |
| SNAPSHOT QUIESCERS | 1138 |
| SNAPSHOT_RANGES | 1140 |
| SNAPSHOT_STATEMENT | 1141 |
| SNAPSHOT_SUBSECT | 1143 |
| SNAPSHOT_SWITCHES | 1145 |
| SNAPSHOT_TABLE | 1146 |
| SNAPSHOT_TBREORG | 1148 |
| SNAPSHOT_TBS | 1149 |
| SNAPSHOT_TBS_CFG | 1152 |
| SQLCACHE_SNAPSHOT | 1154 |
| RSYSINSTALLROUTINES | 1156 |
| WLM_GET_ACTIVITY_DETAILS - Devolver información detallada sobre una actividad específica | 1156 |
| WLM_GET_SERVICE_CLASS_AGENTS - Listar agentes en ejecución en una clase de servicio | 1163 |
| WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES - Lista de apariciones de carga de trabajo | 1168 |
| WLM_GET_SERVICE_SUBCLASS_STATS - Devolver estadísticas de subclases de servicio | 1173 |
| WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES - Devolver una lista de actividades | 1180 |
| WLM_GET_WORKLOAD_STATS - Devolver estadísticas de carga de trabajo | 1185 |

Apéndice A. Visión general de la información técnica de DB2. 1189

| | |
|--|------|
| Biblioteca técnica de DB2 en copia impresa o en formato PDF | 1189 |
| Pedido de manuales de DB2 en copia impresa | 1192 |
| Visualización de la ayuda para estados de SQL desde el procesador de línea de mandatos | 1193 |
| Acceso a diferentes versiones del Centro de información de DB2 | 1193 |
| Visualización de temas en su idioma preferido en el Centro de información de DB2 | 1194 |
| Actualización del Centro de información de DB2 instalado en el sistema o en el servidor de intranet | 1194 |
| Actualización manual del Centro de información de DB2 instalado en el sistema o en el servidor de intranet | 1196 |
| Guías de aprendizaje de DB2 | 1198 |
| Información de resolución de problemas de DB2 | 1198 |
| Términos y condiciones | 1199 |

Apéndice B. Avisos 1201

Índice 1205

Capítulo 1. Vistas y rutinas definidas por el sistema

Las vistas y rutinas definidas por el sistema proporcionan una interfaz programática, fácil de utilizar y primaria, y usan DB2 mediante SQL. Incluyen una colección de vistas incorporadas, funciones de tabla, procedimientos y funciones escalares para realizar diversas tareas de DB2. Por ejemplo, las rutinas definidas por el sistema están disponibles para reorganizar una tabla, capturar y recuperar datos del supervisor o recuperar el ID de aplicación de la conexión actual.

Estas rutinas y vistas se pueden invocar desde una aplicación basada en SQL, una línea de mandatos de DB2 o un script de mandatos.

Autorización para vistas administrativas

Para todas las vistas administrativas del esquema SYSIBMADM, necesita el privilegio SELECT para la vista. Se puede validar con la consulta siguiente para comprobar que su ID de autorización, o un grupo o un rol al que pertenezca, dispone del privilegio SELECT (es decir, cumple los criterios de búsqueda y aparece listado en la columna GRANTEE):

```
SELECT GRANTEE, GRANTEETYPE
FROM SYSCAT.TABAUTH
WHERE TABSCHEMA = 'SYSIBMADM' AND TABNAME = '<nombre_vista>' AND
SELECTAUTH <> 'N'
```

donde <nombre_vista> es el nombre de la vista administrativa.

Con la excepción de SYSIBMADM.AUTHORIZATIONIDS, SYSIBMADM.OBJECTOWNERS y SYSIBMADM.PRIVILEGES, también necesita el privilegio EXECUTE para la función de tabla administrativa. La función de tabla administrativa subyacente aparece listada en la sección de autorización de la vista administrativa. Se puede validar con la consulta siguiente:

```
SELECT GRANTEE, GRANTEETYPE
FROM SYSCAT.ROUTINEAUTH
WHERE SCHEMA = 'SYSPROC' AND SPECIFICNAME = '<nombre_rutina>' AND
EXECUTEAUTH <> 'N'
```

donde <nombre_rutina> es el nombre de la función de tabla administrativa tal como aparece en la documentación.

Algunas vistas administrativas requieren autorizaciones adicionales además de SELECT para la vista y EXECUTE para la función de tabla administrativa subyacente. Las autorizaciones adicionales necesarias están documentadas en la información de consulta donde se describe la vista.

Vistas administrativas comparadas con funciones de tablas

DB2 Versión 9.5 presentó vistas administrativas que ofrecen una interfaz de programación de aplicaciones fácil de usar para las funciones administrativas de DB2 a través de SQL.

Las vistas administrativas se dividen en tres categorías:

- Vistas basadas en vistas de catálogos.

- Vistas basadas en funciones de tablas sin parámetros de entrada.
- Vistas basadas en funciones de tablas con uno o más parámetros de entrada.

Las vistas administrativas son las únicas interfaces documentadas y las preferidas para las vistas basadas en vistas de catálogos y las vistas basadas en funciones de tablas sin parámetros de entrada porque las funciones de tablas no proporcionan información adicional ni ventajas en el rendimiento.

Para las vistas administrativas basadas en funciones de tablas con uno o más parámetros de entrada se pueden utilizar la vista administrativa y la función de tabla, cada una de las cuales consigue un objetivo:

- La vista administrativa ADMINTABINFO y la función de tabla ADMIN_GET_TAB_INFO_V95: la vista administrativa recupera información para todas las tablas de la base de datos. Esto puede tener ventajas significativas en el rendimiento de grandes bases de datos. El impacto en el rendimiento se puede reducir mediante el uso de la función de tabla y la especificación de un esquema, una tabla o ambos como entrada.
- La vista administrativa PDLOGMSG_LAST24HOURS y la función de tabla PD_GET_LOG_MSGS: la vista administrativa, que recupera mensajes de anotaciones cronológicas de notificación, ofrece un acceso rápido a los datos de las últimas 24 horas, mientras que la función de tabla permite recuperar datos de un período concreto.
- Todas las vistas administrativas del supervisor de instantáneas y funciones de tablas (vistas administrativas SNAP*, funciones de tablas SNAP_GET_*): las vistas administrativas del supervisor de instantáneas ofrecen acceso a los datos desde cada partición de base de datos. Las funciones de tablas ofrecen la opción de elegir entre los datos de una única partición de base de datos o los datos agregados en todas las particiones de base de datos.

Es posible que se deban cambiar las aplicaciones que utilizan las funciones de tablas en lugar de las vistas porque las funciones de tablas pueden cambiar en cada versión, para habilitar que se devuelva la nueva información. La función de tabla nueva tendrá el mismo nombre base que la función original y '_Vxx' se añadirá como sufijo, para indicar la versión del producto en que se añadió (por ejemplo, _V97). Las vistas administrativas siempre se basarán en la versión más actual de las funciones de tablas, y por consiguiente permitirán una mayor portabilidad de aplicaciones. Puesto que las columnas pueden variar de una versión a la siguiente, se recomienda la selección de columnas específicas desde las vistas administrativas o que se describa el conjunto de resultados si una aplicación utiliza una sentencia SELECT *.

Capítulo 2. Rutinas y vistas de SQL definidas por el sistema soportadas

Las tablas siguientes resumen información sobre las vistas y rutinas administrativas SQL soportadas.

- Rutinas administrativas SQL de supervisión de actividad: Tabla 1
- Procedimiento ADMIN_CMD almacenado y rutinas administrativas SQL asociadas: Tabla 2 en la página 4
- Vistas y rutinas del planificador de tareas administrativas: Tabla 3 en la página 5
- Comprobar rutinas y procedimientos Tabla 4 en la página 5
- Vistas y rutinas administrativas SQL del mantenimiento automático: Tabla 5 en la página 5
- Procedimientos almacenados de la API de SQL común: Tabla 6 en la página 6
- Vistas y rutinas administrativas SQL de la configuración: Tabla 7 en la página 6
- Entorno de vistas administrativas: Tabla 8 en la página 7
- Rutinas administrativas SQL de instantáneas de salud: Tabla 9 en la página 7
- Rutinas administrativas SQL de supervisión de actividad: Tabla 10 en la página 9
- Rutinas administrativas SQL de MQSeries: Tabla 11 en la página 10
- Vistas y rutinas administrativas SQL de la seguridad: Tabla 12 en la página 11
- Vistas y rutinas administrativas SQL de instantáneas: Tabla 13 en la página 11
- Rutinas administrativas SQL de procedimientos SQL: Tabla 14 en la página 15
- Rutinas administrativas SQL de redistribución paso a paso: Tabla 15 en la página 15
- Rutinas administrativas SQL de herramientas de almacenamiento: Tabla 16 en la página 16
- Rutinas administrativas SQL de búsqueda de texto: Tabla 17 en la página 16
- Rutinas de gestión de carga de trabajo: Tabla 18 en la página 17
- Vistas y rutinas administrativas SQL diversas: Tabla 19 en la página 17

Tabla 1. Rutinas administrativas SQL de supervisión de actividad

| Nombre de la rutina | Esquema | Descripción |
|--------------------------------------|---------|---|
| AM_BASE_RPT_RECOMS, función de tabla | SYSPROC | Esta función de tabla devuelve las recomendaciones para los informes de actividad que utiliza el supervisor de actividad. |
| AM_BASE_RPTS, función de tabla | SYSPROC | Esta función de tabla devuelve los informes de actividad que utiliza el supervisor de actividad. |
| AM_DROP_TASK, procedimiento | SYSPROC | Este procedimiento suprime una tarea de supervisión. |
| AM_GET_LOCK_CHN_TB, procedimiento | SYSPROC | Este procedimiento devuelve datos de la cadena de bloqueo de aplicación en formato tabular. |
| AM_GET_LOCK_CHNS, procedimiento | SYSPROC | Este procedimiento visualiza las series de bloqueo de una aplicación especificada utilizando una serie con formato. |

Tabla 1. Rutinas administrativas SQL de supervisión de actividad (continuación)

| Nombre de la rutina | Esquema | Descripción |
|--------------------------------|---------|--|
| AM_GET_LOCK_RPT, procedimiento | SYSPROC | Este procedimiento visualiza los detalles del bloque de una aplicación. |
| AM_GET_RPT, procedimiento | SYSPROC | Este procedimiento visualiza los datos sobre el supervisor de actividad para un informe. |
| AM_SAVE_TASK, procedimiento | SYSPROC | Este procedimiento crea o modifica una tarea de supervisión. |

Tabla 2. Procedimiento ADMIN_CMD almacenado y rutinas administrativas SQL asociadas

| Nombre de la rutina | Esquema | Descripción |
|--|---|---|
| ADMIN_CMD, procedimiento | SYSPROC | Este procedimiento permite al administrador ejecutar mandatos administrativos (incluidos los mandatos CLP del procesador de línea de mandatos DB2 ejecutando ADMIN_CMD mediante una sentencia CALL. |
| Función ADMIN_EST_INLINE_LENGTH | SYSIBM | Esta función devuelve una estimación de la longitud en línea necesaria para poner en línea los datos almacenados en una columna XML, BLOB, CLOB o DBCLOB. |
| Función de tabla ADMIN_GET_DBP_MEM_USAGE | SYSPROC | Esta función de tabla obtiene el consumo total de memoria de una instancia determinada. |
| ADMIN_GET_INDEX_COMPRESS_INFO | SYSPROC | Esta función de tabla devuelve los ahorros de compresión de índice potenciales para índices no comprimidos o notifica las estadísticas de compresión de índice desde los catálogos. |
| ADMIN_GET_INDEX_INFO, función de tabla | SYSPROC | Esta función de tabla devuelve información de índice que no está disponible en las vistas de catálogo. |
| ADMIN_GET_MSGS, función de tabla | SYSPROC | Esta función de tabla ADMIN_GET_MSGS se utiliza para recuperar los mensajes generados por un programa de utilidad de movimiento de datos que se ejecuta mediante el procedimiento ADMIN_CMD. |
| Función ADMIN_IS_INLINED | SYSIBM | Esta función recupera la información de estado de los datos en línea correspondiente a una columna XML, BLOB, CLOB o DBCLOB. |
| ADMIN_REMOVE_MSGS, procedimiento | SYSPROC | Este procedimiento se utiliza para borrar los mensajes generados por los programas de utilidad de movimiento de datos que se ejecutan mediante el procedimiento ADMIN_CMD. |
| Procedimiento ADMIN_REVALIDATE_DB_OBJECTS | SYSPROC | Este procedimiento revalida objetos de base de datos no válidos. |
| Vista ADMINTABCOMPRESSINFO y función de tabla ADMIN_GET_TAB_COMPRESS_INFO_V97 | SYSIBMADM (vista administrativa), SYSPROC (función de tabla) | Esta vista y esta función devuelven información de compresión para tablas, tablas de consultas materializadas (MQT) y tablas de jerarquía. |

Tabla 2. Procedimiento ADMIN_CMD almacenado y rutinas administrativas SQL asociadas (continuación)

| Nombre de la rutina | Esquema | Descripción |
|--|---|---|
| ADMINTABINFO y ADMIN_GET_TAB_INFO_V97 | SYSIBMADM (vista administrativa), SYSPROC (función de tabla) | Esta vista y esta función de tabla devuelven información de tamaño y estado para las tablas, las tablas de consultas materializadas (MQT) y las tablas de jerarquía. |
| Vista ADMINTEMPCOLUMNS y función de tabla ADMIN_GET_TEMP_COLUMNS | SYSIBMADM (vista administrativa), SYSPROC (función de tabla) | Esta vista y esta función de tabla recuperan información de atributos de columna para las tablas temporales creadas y las tablas temporales declaradas. |
| Vista ADMINTEMPTABLES y función de tabla ADMIN_GET_TEMP_TABLES | SYSIBMADM (vista administrativa), SYSPROC (función de tabla) | Esta vista y esta función de tabla recuperan información de atributos de tabla y de tiempo de creación de instancias para las instancias de tablas temporales creadas y tablas temporales declaradas. |

Tabla 3. Vistas y rutinas del planificador de tareas administrativas

| Nombre de la vista o rutina | Esquema | Descripción |
|-----------------------------|----------|--|
| ADMIN_TASK_ADD | SYSPROC | Este procedimiento planifica una tarea administrativa. |
| ADMIN_TASK_LIST | SYSTOOLS | Esta vista administrativa recupera información sobre cada tarea definida en el planificador. |
| ADMIN_TASK_REMOVE | SYSPROC | Este procedimiento elimina registros de estados de tareas o de tareas planificadas. |
| ADMIN_TASK_STATUS | SYSTOOLS | Esta vista administrativa recupera información sobre el estado de cada tarea. |
| ADMIN_TASK_UPDATE | SYSPROC | Este procedimiento actualiza una tarea existente |

Tabla 4. Comprobar rutinas y procedimientos

| Nombre de la vista o rutina | Esquema | Descripción |
|---|---------|---|
| Procedimiento y función de tabla AUDIT_ARCHIVE | SYSPROC | Este procedimiento y la función de tabla archivan el registro de comprobación actual. |
| Procedimiento AUDIT_DELM_EXTRACT | SYSPROC | Este procedimiento extrae datos de los registros archivados de forma binaria y los carga en archivos delimitados. |
| Función de tabla AUDIT_LIST_LOGS | SYSPROC | Esta función de tabla devuelve una lista de los registros de comprobación archivados en la vía de acceso especificada para la base de datos actual. |

Tabla 5. Vistas y rutinas administrativas SQL del mantenimiento automático

| Nombre de la vista o rutina | Esquema | Descripción |
|---|---------|---|
| Procedimiento AUTOMAINT_GET_POLICY | SYSPROC | Este procedimiento obtiene los valores actuales del mantenimiento automático para la base de datos. |
| Procedimiento AUTOMAINT_GET_POLICYFILE | SYSPROC | Este procedimiento obtiene los valores actuales del mantenimiento automático para la base de datos. |

Tabla 5. Vistas y rutinas administrativas SQL del mantenimiento automático (continuación)

| Nombre de la vista o rutina | Esquema | Descripción |
|--|---------|--|
| Procedimiento AUTOMAINT_SET_POLICY | SYSPROC | Este procedimiento establece los valores de la política de mantenimiento automático para la base de datos conectada actualmente. |
| Procedimiento AUTOMAINT_SET_POLICYFILE | SYSPROC | Este procedimiento establece los valores del mantenimiento automático para la base de datos conectada actualmente. |

Tabla 6. Procedimientos almacenados de la API de SQL común

| Nombre de la vista o rutina | Esquema | Descripción |
|-------------------------------|---------|---|
| Procedimiento CANCEL_WORK | SYSPROC | Este procedimiento cancela una actividad especificada. Si no se ha especificado un ID de actividad exclusivo, cancela todas las actividades de una aplicación conectada y fuerza a la aplicación a salir del sistema. |
| Procedimiento GET_CONFIG | SYSPROC | Este procedimiento recupera datos de configuración del servidor de datos, incluidos datos del archivo nodes.cfg, datos de configuración del gestor de bases de datos, datos de configuración de la base de datos y valores de registro de todas las particiones de base de datos. |
| Procedimiento GET_MESSAGE | SYSPROC | Este procedimiento recupera el texto de mensaje abreviado, el texto de mensaje largo y el SQLSTATE para un SQLCODE. |
| Procedimiento GET_SYSTEM_INFO | SYSPROC | Este procedimiento recupera información sobre el servidor de datos, incluida información sobre el sistema, la instancia actual, los productos de base de datos DB2 instalados, las variables de entorno, las CPU disponibles y otra información de sistema. |
| Procedimiento SET_CONFIG | SYSPROC | Este procedimiento actualiza los parámetros de configuración recuperados con el procedimiento GET_CONFIG. |

Tabla 7. Vistas y rutinas administrativas SQL de la configuración

| Nombre de la vista o rutina | Esquema | Descripción |
|-------------------------------------|-----------|---|
| DB_PARTITIONS, función de tabla | SYSPROC | Esta función de tabla devuelve el contenido del archivo db2nodes.cfg en formato de tabla. |
| DBCFCG, vista administrativa | SYSIBMADM | Esta vista administrativa devuelve información sobre la configuración de la base de datos. |
| DBMCFG, vista administrativa | SYSIBMADM | Esta vista administrativa devuelve información sobre la configuración del gestor de bases de datos. |
| REG_VARIABLES, vista administrativa | SYSIBMADM | Esta vista administrativa devuelve los valores de registro de DB2 de todas las particiones de bases de datos. |

Tabla 8. Vistas administrativas del entorno

| Nombre de vista | Esquema | Descripción |
|--|-----------|---|
| Vista administrativa ENV_FEATURE_INFO | SYSPROC | Esta vista administrativa devuelve información sobre todas las características disponibles para las que se necesita una licencia. |
| Vista administrativa ENV_INST_INFO | SYSIBMADM | Esta vista administrativa devuelve información acerca de la instancia actual. |
| ENV_PROD_INFO, vista administrativa | SYSIBMADM | Esta vista administrativa devuelve información acerca de los productos de base de datos DB2 instalados. |
| ENV_SYS_INFO, vista administrativa | SYSIBMADM | Esta vista administrativa devuelve información acerca del sistema. |
| Vista administrativa ENV_SYS_RESOURCES | SYSIBMADM | Esta vista administrativa indica el sistema operativo, la CPU, la memoria y demás información relacionada con el sistema |

Tabla 9. Rutinas administrativas SQL de instantáneas de salud

| Nombre de la rutina | Esquema | Descripción |
|-------------------------------------|---------|---|
| Función de tabla HEALTH_CONT_HI | SYSPROC | Esta función de tabla devuelve una tabla con la información sobre el indicador de salud de los contenedores a partir de una instantánea de salud de una base de datos. |
| Función de tabla HEALTH_CONT_HI_HIS | SYSPROC | Esta función de tabla devuelve una tabla con la información histórica sobre el indicador de salud a partir de una instantánea de salud de una base de datos. |
| HEALTH_CONT_INFO, función de tabla | SYSPROC | Esta función de tabla devuelve una tabla con la información sobre el estado de la alerta retrotraída para contenedores a partir de una instantánea de salud de una base de datos. |
| Función de tabla HEALTH_DB_HI | SYSPROC | Esta función de tabla devuelve una tabla con la información sobre el indicador de salud a partir de una instantánea de salud de una base de datos. |
| Función de tabla HEALTH_DB_HI_HIS | SYSPROC | Esta función de tabla devuelve una tabla con la información histórica sobre el indicador de salud a partir de una instantánea de salud de una base de datos. |
| HEALTH_DB_HIC, función de tabla | SYSPROC | Esta función de tabla devuelve información sobre el indicador de salud de la colección a partir de una instantánea de salud de una base de datos. |
| HEALTH_DB_HIC_HIS, función de tabla | SYSPROC | Esta función de tabla devuelve información histórica sobre el indicador de salud de la colección a partir de una instantánea de salud de una base de datos. |
| HEALTH_DB_INFO, función de tabla | SYSPROC | Esta función de tabla devuelve una tabla con la información sobre el estado de la alerta retrotraída a partir de una instantánea de salud de una o de todas las bases de datos. |

Tabla 9. Rutinas administrativas SQL de instantáneas de salud (continuación)

| Nombre de la rutina | Esquema | Descripción |
|---|---------|---|
| Función de tabla HEALTH_DBM_HI | SYSPROC | Esta función de tabla devuelve una tabla con la información sobre el indicador de salud a partir de una instantánea de salud del gestor de bases de datos DB2. |
| Función de tabla HEALTH_DBM_HI_HIS | SYSPROC | Esta función de tabla devuelve una tabla con la información histórica sobre el indicador de salud a partir de una instantánea de salud del gestor de bases de datos DB2. |
| HEALTH_DBM_INFO, función de tabla | SYSPROC | Esta función de tabla devuelve una tabla con la información sobre el estado de la alerta retrotraída a partir de una instantánea de salud del gestor de bases de datos DB2. |
| HEALTH_GET_ALERT_ACTION_CFG, función de tabla | SYSPROC | Esta función de tabla devuelve valores de configuración de acciones de alerta de salud correspondientes a diversos tipos de objetos (gestor de bases de datos, base de datos, espacio de tablas y contenedores de espacios de tablas) y a diversos niveles de configuración (valor por omisión de instalación, instancia, global y objeto). |
| HEALTH_GET_ALERT_CFG, función de tabla | SYSPROC | Esta función de tabla devuelve valores de configuración de alertas de salud correspondientes a diversos tipos de objetos (gestor de bases de datos, base de datos, espacio de tablas y contenedores de espacios de tablas) y a diversos niveles de configuración (valor por omisión de instalación, global y objeto). |
| HEALTH_GET_IND_DEFINITION, función de tabla | SYSPROC | Esta función de tabla devuelve las definiciones del indicador de salud. |
| Procedimiento HEALTH_HI_REC | SYSPROC | Este procedimiento recupera un conjunto de recomendaciones que direccionan un indicador de salud en estado de alerta en un objeto DB2 determinado. |
| Función de tabla HEALTH_TBS_HI | SYSPROC | Esta función de tabla devuelve una tabla con la información sobre el indicador de salud de los espacios de tablas a partir de una instantánea de salud de una base de datos. |
| Función de tabla HEALTH_TBS_HI_HIS | SYSPROC | Esta función de tabla devuelve una tabla con la información sobre el indicador histórico de salud de los espacios de tablas a partir de una instantánea de salud de una base de datos. |
| HEALTH_TBS_INFO, función de tabla | SYSPROC | Esta función de tabla devuelve una tabla con la información sobre el estado de la alerta retrotraída para espacios de tablas a partir de una instantánea de salud de una base de datos. |

Tabla 10. Rutinas de SQL del supervisor

| Nombre de la rutina | Esquema | Descripción |
|--|---------|---|
| EVMON_FORMAT_UE_TO_TABLES, procedimiento | SYSPROC | Este procedimiento recupera los datos almacenados en una tabla de sucesos sin formato y mueve el documento XML a un conjunto de tablas relacionales. |
| Función de tabla EVMON_FORMAT_UE_TO_TABLES | SYSPROC | Esta función de tabla extrae sucesos binarios de una tabla de sucesos sin formato, les da formato y los incluye en un documento XML. |
| MON_GET_ACTIVITY_DETAILS | SYSPROC | Esta función de tabla devuelve detalles sobre una actividad, incluida información de actividad general y un conjunto de métricas para la actividad. |
| Función de tabla MON_GET_BUFFERPOOL | SYSPROC | Esta función de tabla devuelve métricas del supervisor para una o varias agrupaciones de almacenamientos intermedios. |
| Función de tabla MON_GET_CONNECTION | SYSPROC | Esta función de tabla devuelve métricas para una o varias conexiones. |
| Función de tabla MON_GET_CONNECTION_DETAILS | SYSPROC | Esta función de tabla devuelve métricas detalladas para una o varias conexiones. |
| Función de tabla MON_GET_CONTAINER | SYSPROC | Esta función de tabla devuelve métricas del supervisor para uno o varios contenedores de espacios de tablas. |
| Función de tabla MON_GET_EXTENT_MOVEMENT_STATUS | SYSPROC | Esta función de tabla devuelve el estado de la operación de traslado de extensiones. |
| MON_GET_INDEX, función de tabla | SYSPROC | Esta función de tabla devuelve métricas para uno o varios índices. |
| MON_GET_PKG_CACHE_STMT, función de tabla | SYSPROC | Esta función de tabla devuelve una vista de punto en el tiempo de las sentencias de SQL tanto estático como dinámico en la antememoria del paquete de bases de datos. |
| Función de tabla MON_GET_SERVICE_SUBCLASS | SYSPROC | Esta función de tabla devuelve métricas para una o varias subclases de servicio. |
| Función de tabla MON_GET_SERVICE_SUBCLASS_DETAILS | SYSPROC | Esta función de tabla devuelve métricas detalladas para una o varias subclases de servicio. |
| Función de tabla MON_GET_TABLE | SYSPROC | Esta función de tabla devuelve métricas del supervisor para una o varias tablas. |
| Función de tabla MON_GET_TABLESPACE | SYSPROC | Esta función de tabla devuelve métricas del supervisor para uno o varios espacios de tablas. |

Tabla 10. Rutinas de SQL del supervisor (continuación)

| Nombre de la rutina | Esquema | Descripción |
|---|---------|---|
| Función de tabla MON_GET_UNIT_OF_WORK | SYSPROC | Esta función de tabla devuelve métricas para una o varias unidades de trabajo. |
| Función de tabla MON_GET_UNIT_OF_WORK_DETAILS | SYSPROC | Esta función de tabla devuelve métricas detalladas para una o varias unidades de trabajo. |
| Función de tabla MON_GET_WORKLOAD | SYSPROC | Esta función de tabla devuelve métricas para una o varias cargas de trabajo. |
| Función de tabla MON_GET_WORKLOAD_DETAILS | SYSPROC | Esta función de tabla devuelve métricas detalladas para una o varias cargas de trabajo. |

Tabla 11. Rutinas administrativas SQL de MQSeries

| Nombre de la rutina | Esquema | Descripción |
|-----------------------------------|----------------|---|
| Función escalar MQPUBLISH | DB2MQ, DB2MQ1C | Esta función escalar publica datos en una ubicación MQSeries. |
| Función escalar MQREAD | DB2MQ, DB2MQ1C | Esta función escalar devuelve un mensaje de una ubicación MQSeries. |
| Función de tabla MQREADALL | DB2MQ, DB2MQ1C | Esta función de tabla devuelve una tabla con mensajes y metadatos de mensajes de una ubicación MQSeries. |
| Función de tabla MQREADALLCLOB | DB2MQ | Esta función de tabla devuelve una tabla que contiene mensajes y metadatos de mensajes desde una ubicación MQSeries especificada. |
| Función escalar MQREADCLOB | DB2MQ | Esta función escalar devuelve un mensaje de una ubicación MQSeries especificada. |
| Función escalar MQRECEIVE | DB2MQ, DB2MQ1C | Esta función escalar devuelve un mensaje de una ubicación MQSeries y suprime el mensaje de la cola asociada. |
| Función de tabla MQRECEIVEALL | DB2MQ, DB2MQ1C | Esta función de tabla (una rutina administrativa de SQL) devuelve una tabla que contiene mensajes y metadatos de los mensajes desde una ubicación MQSeries y suprime los mensajes de las colas asociadas. |
| Función de tabla MQRECEIVEALLCLOB | DB2MQ | Esta función de tabla devuelve una tabla que contiene mensajes y metadatos de mensajes desde una ubicación MQSeries especificada. |
| Función escalar MQRECEIVECLOB | DB2MQ | Esta función escalar devuelve un mensaje de una ubicación MQSeries especificada. |
| Función escalar MQSEND | DB2MQ, DB2MQ1C | Esta función escalar envía datos a una ubicación MQSeries. |
| Función escalar MQSUBSCRIBE | DB2MQ, DB2MQ1C | Esta función escalar suscribe a los mensajes MQSeries publicados sobre un tema específico. |
| Función escalar MQUNSUBSCRIBE | DB2MQ, DB2MQ1C | Esta función escalar cancela la suscripción a los mensajes MQSeries publicados sobre un tema específico. |

Tabla 12. Vistas y rutinas administrativas SQL de la seguridad:

| Nombre de la vista o rutina | Esquema | Descripción |
|--|-----------|---|
| Función de tabla AUTH_LIST_AUTHORITIES_FOR_AUTHID | SYSPROC | Esta función de tabla devuelve todas las autorizaciones de este ID de autorización encontradas en el archivo de configuración de la base de datos u otorgada a un ID de autorización directa o indirectamente mediante un grupo o un rol. |
| AUTH_LIST_GROUPS_FOR _AUTHID, función de tabla | SYSPROC | Esta función de tabla devuelve la lista de grupos de los que es miembro el ID de autorización determinado. |
| Función AUTH_LIST_ROLES_FOR_AUTHID | SYSPROC | Esta función devuelve la lista de roles de los que es miembro el ID de autorización determinado. |
| AUTHORIZATIONIDS, vista administrativa | SYSIBMADM | Esta vista administrativa contiene una lista de los ID de autorización a los que se han otorgado privilegios o autorizaciones, junto con sus tipos, para la base de datos conectada actualmente. |
| OBJECTOWNERS, vista administrativa | SYSIBMADM | Esta vista administrativa contiene toda la información sobre la propiedad de objetos correspondiente a la base de datos conectada actualmente. |
| PRIVILEGES, vista administrativa | SYSIBMADM | Esta vista administrativa contiene todos los privilegios explícitos correspondientes a la base de datos conectada actualmente. |

Tabla 13. Vistas y rutinas administrativas SQL de instantáneas

| Nombre de la vista o rutina | Esquema | Descripción |
|---|-----------|--|
| Vista administrativa APPL_PERFORMANCE | SYSIBMADM | Esta vista administrativa muestra información acerca del índice filas seleccionado en comparación con las filas leídas por aplicación. |
| Vista administrativa APPLICATIONS | SYSIBMADM | Esta vista administrativa devuelve información sobre las aplicaciones de base de datos conectadas. |
| Vista administrativa BP_HITRATIO | SYSIBMADM | Esta vista de administración devuelve la tasa de aciertos de la agrupación de antememorias de la base de datos, incluidos el total, los datos y el índice. |
| Vista administrativa BP_READ_IO | SYSIBMADM | Esta vista administrativa devuelve información sobre el rendimiento de lectura de la agrupación de almacenamientos intermedios. |
| Vista administrativa BP_WRITE_IO | SYSIBMADM | Esta vista administrativa devuelve información sobre el rendimiento de grabación de cada agrupación de almacenamientos intermedios. |
| Vista administrativa CONTAINER_UTILIZATION | SYSIBMADM | Esta vista administrativa devuelve información acerca de los contenedores de espacios de tablas y los índices de utilización. |

Tabla 13. Vistas y rutinas administrativas SQL de instantáneas (continuación)

| Nombre de la vista o rutina | Esquema | Descripción |
|--|---|---|
| Vista administrativa LOCKS_HELD | SYSIBMADM | Esta vista administrativa devuelve información sobre los bloqueos actuales. |
| Vista administrativa LOCKWAITS | SYSIBMADM | Esta vista administrativa devuelve información sobre los bloqueos que están a la espera de ser concedidos. |
| Vista administrativa LOG_UTILIZATION | SYSIBMADM | Esta vista administrativa devuelve información sobre la utilización de las anotaciones cronológicas para la base de datos conectada actualmente. |
| Vista administrativa LONG_RUNNING_SQL | SYSIBMADM | Esta vista administrativa devuelve las sentencias SQL de ejecución más larga de la base de datos conectada actualmente. |
| Vista administrativa QUERY_PREP_COST | SYSIBMADM | Esta vista administrativa devuelve una lista de las sentencias con información acerca del tiempo necesario para preparar la sentencia. |
| Procedimiento SNAP_WRITE_FILE | SYSPROC | Este procedimiento graba datos de instantánea del sistema en un archivo ubicado en el subdirectorio tmp del directorio de la instancia. |
| Vista administrativa SNAPAGENT y función de tabla SNAP_GET_AGENT | SYSIBMADM (vista administrativa), SYSPROC (función de tabla) | Esta vista administrativa y función de tabla devuelven información acerca de los agentes a partir de una instantánea de la aplicación, en concreto del grupo de datos lógico agent. |
| Vista administrativa SNAPAGENT_MEMORY_POOL y función de tabla SNAP_GET_AGENT_MEMORY_POOL | SYSIBMADM (vista administrativa), SYSPROC (función de tabla) | La vista administrativa y la función de tabla devuelven información acerca del uso de la memoria a nivel de agente. |
| Vista administrativa SNAPAPPL y función de tabla SNAP_GET_APPL_V95 | SYSIBMADM (vista administrativa), SYSPROC (función de tabla) | Esta vista administrativa y función de tabla devuelven información acerca de las aplicaciones a partir de una instantánea de la aplicación, en concreto del grupo de datos lógico appl. |
| Vista administrativa SNAPAPPL_INFO y función de tabla SNAP_GET_APPL_INFO_V95 | SYSIBMADM (vista administrativa), SYSPROC (función de tabla) | Esta vista administrativa y función de tabla devuelven información acerca de las aplicaciones a partir de una instantánea de la aplicación, en concreto del grupo de datos lógico appl_info. |
| Vista administrativa SNAPBP y función de tabla SNAP_GET_BP_V95 | SYSIBMADM (vista administrativa), SYSPROC (función de tabla) | Esta vista administrativa y función de tabla devuelven información acerca de las agrupaciones de almacenamientos intermedios a partir de una instantánea de las agrupaciones de almacenamientos intermedios, en concreto del grupo de datos lógico bufferpool. |
| Vista administrativa SNAPBP_PART y función de tabla SNAP_GET_BP_PART | SYSIBMADM (vista administrativa), SYSPROC (función de tabla) | Esta vista administrativa y función de tabla devuelven información acerca de las agrupaciones de almacenamientos intermedios a partir de una instantánea de las agrupaciones de almacenamientos intermedios, en concreto del grupo de datos lógico bufferpool_nodeinfo. |

Tabla 13. Vistas y rutinas administrativas SQL de instantáneas (continuación)

| Nombre de la vista o rutina | Esquema | Descripción |
|--|---|---|
| Vista administrativa SNAPCONTAINER y función de tabla SNAP_GET_CONTAINER_V91 | SYSIBMADM (vista administrativa), SYSPROC (función de tabla) | Esta vista administrativa y función de tabla devuelven información de la instantánea del espacio de tablas a partir del grupo de datos lógicos tablespace_container. |
| Vista administrativa SNAPDB y función de tabla SNAP_GET_DB_V95 | SYSIBMADM (vista administrativa), SYSPROC (función de tabla) | Esta vista administrativa y función de tabla devuelven información de la instantánea a partir de los grupos lógicos de base de datos (dbase) y de almacenamiento de base de datos (db_storage_group). |
| Vista administrativa SNAPDB_MEMORY_POOL y función de tabla SNAP_GET_DB_MEMORY_POOL | SYSIBMADM (vista administrativa), SYSPROC (función de tabla) | Esta vista administrativa y función de tabla devuelven información acerca del uso de la memoria en la base de datos sólo para plataformas UNIX®. |
| Vista administrativa SNAPDBM y función de tabla SNAP_GET_DBM_V95 | SYSIBMADM (vista administrativa), SYSPROC (función de tabla) | Esta vista administrativa y función de tabla devuelven información acerca de los grupos lógicos del gestor de bases de datos DB2 (dbm) y del supervisor de instantáneas. |
| Vista administrativa SNAPDBM_MEMORY_POOL y función de tabla SNAP_GET_DBM_MEMORY_POOL | SYSIBMADM (vista administrativa), SYSPROC (función de tabla) | Esta vista administrativa y función de tabla devuelven información acerca del uso de la memoria en el gestor de bases de datos. |
| Vista administrativa SNAPDETAILLOG y función de tabla SNAP_GET_DETAILLOG_V91 | SYSIBMADM (vista administrativa), SYSPROC (función de tabla) | Esta vista administrativa y función de tabla devuelven información de las instantáneas del grupo de datos lógicos detail_log. |
| Vista administrativa SNAPDYN_SQL y función de tabla SNAP_GET_DYN_SQL_V95 | SYSIBMADM (vista administrativa), SYSPROC (función de tabla) | Esta vista administrativa y función de tabla devuelven información de las instantáneas a partir del grupo de datos lógicos dynsql. |
| Vista administrativa SNAPFCM y función de tabla SNAP_GET_FCM | SYSIBMADM (vista administrativa), SYSPROC (función de tabla) | Esta vista administrativa y función de tabla devuelven información sobre FCM (Fast Communication Manager) a partir de una instantánea del gestor de bases de datos, en concreto, el grupo de datos lógicos fcm. |
| Vista administrativa SNAPFCM_PART y función de tabla SNAP_GET_FCM_PART | SYSIBMADM (vista administrativa), SYSPROC (función de tabla) | Esta vista administrativa y función de tabla devuelven información sobre FCM (Fast Communication Manager) a partir de una instantánea del gestor de bases de datos, en concreto, el grupo de datos lógicos fcm_node. |
| Vista administrativa SNAPHADR y función de tabla SNAP_GET_HADR | SYSIBMADM (vista administrativa), SYSPROC (función de tabla) | Esta vista administrativa y función de tabla devuelven información sobre la recuperación de catástrofes de alta disponibilidad a partir de la instantánea de una base de datos, en concreto el grupo de datos lógicos hadr. |
| Vista administrativa SNAPLOCK y función de tabla SNAP_GET_LOCK | SYSIBMADM (vista administrativa), SYSPROC (función de tabla) | Esta vista administrativa y función de tabla devuelven información de instantáneas acerca de los bloqueos, en concreto el grupo de datos lógicos lock. |

Tabla 13. Vistas y rutinas administrativas SQL de instantáneas (continuación)

| Nombre de la vista o rutina | Esquema | Descripción |
|--|---|---|
| Vista administrativa SNAPLOCKWAIT y función de tabla SNAP_GET_LOCKWAIT | SYSIBMADM (vista administrativa), SYSPROC (función de tabla) | Esta vista administrativa y función de tabla devuelven información de instantáneas acerca de las esperas de bloqueo, en concreto el grupo de datos lógicos lockwait. |
| Vista administrativa SNAPSTMT y función de tabla SNAP_GET_STMT | SYSIBMADM (vista administrativa), SYSPROC (función de tabla) | Esta vista administrativa y función devuelven información acerca de las sentencias de una instantánea de aplicación. |
| Vista administrativa SNAPSTORAGE_PATHS y función de tabla SNAP_GET_STORAGE_PATHS | SYSIBMADM (vista administrativa), SYSPROC (función de tabla) | Esta vista administrativa y función de tabla devuelven una lista de vías de acceso del almacenamiento automático para la base de datos, incluida la información del sistema de archivos correspondiente a cada vía de acceso del almacenamiento, concretamente del grupo de datos lógicos db_storage_group. |
| Vista administrativa SNAPSUBSECTION y función de tabla SNAP_GET_SUBSECTION | SYSIBMADM (vista administrativa), SYSPROC (función de tabla) | Esta vista administrativa y función de tabla devuelven información sobre las subsecciones de aplicaciones, concretamente la agrupación del supervisor lógico subsection. |
| Vista administrativa SNAPSWITCHES y función de tabla SNAP_GET_SWITCHES | SYSIBMADM (vista administrativa), SYSPROC (función de tabla) | Esta vista administrativa y función de tabla devuelven información acerca del estado del conmutador de instantánea de base de datos. |
| Vista administrativa SNAPTAB y función de tabla SNAP_GET_TAB_V91 | SYSIBMADM (vista administrativa), SYSPROC (función de tabla) | Esta vista administrativa y función de tabla devuelven información de las instantáneas del grupo de datos lógicos table. |
| Vista administrativa SNAPTAB_REORG y función de tabla SNAP_GET_TAB_REORG | SYSIBMADM (vista administrativa), SYSPROC (función de tabla) | Esta vista administrativa y función de tabla devuelven información acerca de la reorganización de tabla. |
| Vista administrativa SNAPTbsp y función de tabla SNAP_GET_TBSP_V91 | SYSIBMADM (vista administrativa), SYSPROC (función de tabla) | Esta vista administrativa y función de tabla devuelven información de las instantáneas del grupo de datos lógicos table space. |
| Vista administrativa SNAPTbsp_PART y función de tabla SNAP_GET_TBSP_PART_V91 | SYSIBMADM (vista administrativa), SYSPROC (función de tabla) | Esta vista administrativa y función de tabla devuelven información de las instantáneas del grupo de datos lógicos tablespace_nodeinfo. |
| Vista administrativa SNAPTbsp_QUIESCER y función de tabla SNAP_GET_TBSP_QUIESCER | SYSIBMADM (vista administrativa), SYSPROC (función de tabla) | Esta vista administrativa y función de tabla devuelven información acerca de los inmovilizadores desde una instantánea del espacio de tablas. |
| Vista administrativa SNAPTbsp_RANGE y función de tabla SNAP_GET_TBSP_RANGE | SYSIBMADM (vista administrativa), SYSPROC (función de tabla) | Esta vista administrativa y función de tabla devuelven información de una instantánea de rango. |
| Vista administrativa SNAPUTIL y función de tabla SNAP_GET_UTIL | SYSIBMADM (vista administrativa), SYSPROC (función de tabla) | Esta vista administrativa y función de tabla devuelven información de las instantáneas del grupo de datos lógicos utility_info. |

Tabla 13. Vistas y rutinas administrativas SQL de instantáneas (continuación)

| Nombre de la vista o rutina | Esquema | Descripción |
|--|---|---|
| Vista administrativa SNAPUTIL_PROGRESS y función de tabla SNAP_GET_UTIL_PROGRESS | SYSIBMADM (vista administrativa), SYSPROC (función de tabla) | Esta vista administrativa y función de tabla devuelven información acerca del progreso de los programas de utilidad, en particular, el grupo de datos lógicos progress. |
| Vista administrativa TBSP_UTILIZATION | SYSIBMADM | Esta vista administrativa devuelve información de utilización y configuración de espacio de tablas. |
| TOP_DYNAMIC_SQL, vista administrativa | SYSIBMADM | Esta vista administrativa devuelve las sentencias de SQL dinámico más frecuentes que pueden clasificarse por número de ejecuciones, tiempo de ejecución medio, número de clasificaciones o clasificaciones por sentencia. |

Tabla 14. Rutinas administrativas SQL de procedimientos SQL

| Nombre de la rutina | Esquema | Descripción |
|--------------------------------------|---------|--|
| Procedimiento ALTER_ROUTINE_PACKAGE | SYSPROC | Este procedimiento modifica los valores del paquete asociado con una rutina de SQL compilado o un activador compilado, sin necesidad de realizar revinculaciones. |
| Función escalar GET_ROUTINE_OPTS | SYSPROC | Esta función escalar devuelve un valor de serie de caracteres de las opciones que deben utilizarse para la creación de procedimientos SQL en la sesión actual. |
| GET_ROUTINE_SAR, procedimiento | SYSFUN | Este procedimiento devuelve la información necesaria para instalar una rutina idéntica en otro servidor de bases de datos que funcione por lo menos al mismo nivel y con el mismo sistema operativo. |
| Procedimiento PUT_ROUTINE_SAR | SYSFUN | Este procedimiento pasa la información necesaria para crear y definir una rutina SQL en el servidor de bases de datos. |
| Procedimiento REBIND_ROUTINE_PACKAGE | SYSPROC | Este procedimiento vuelve a vincular el paquete asociado con un procedimiento de SQL. |
| Procedimiento SET_ROUTINE_OPTS | SYSPROC | Este procedimiento establece las opciones que deben utilizarse para la creación de procedimientos SQL en la sesión actual. |

Tabla 15. Rutinas administrativas SQL de redistribución paso a paso

| Nombre de la rutina | Esquema | Descripción |
|---------------------------------|---------|--|
| Procedimiento ANALYZE_LOG_SPACE | SYSPROC | Este procedimiento devuelve información de análisis del espacio de anotaciones cronológicas. |
| Procedimiento GENERATE_DISTFILE | SYSPROC | Este procedimiento genera un archivo de distribución de datos. |
| Procedimiento GET_SWRD_SETTINGS | SYSPROC | Este procedimiento devuelve información de redistribución. |
| Procedimiento SET_SWRD_SETTINGS | SYSPROC | Este procedimiento crea o modifica el registro de redistribución. |

Tabla 15. Rutinas administrativas SQL de redistribución paso a paso (continuación)

| Nombre de la rutina | Esquema | Descripción |
|---|---------|--|
| Procedimiento STEPWISE_REDISTRIBUTE_DBPG | SYSPROC | Este procedimiento redistribuye parte de un grupo de particiones de base de datos. |

Tabla 16. Rutinas administrativas SQL de la herramienta de administración del almacenamiento

| Nombre de la rutina | Esquema | Descripción |
|--|---------|---|
| Procedimiento CAPTURE_STORAGEMGMT_INFO | SYSPROC | Este procedimiento devuelve información relacionada con el almacenamiento de un objeto raíz concreto. |
| Procedimiento CREATE_STORAGEMGMT_TABLES | SYSPROC | Este procedimiento crea tablas de gestión de almacenamiento. |
| Procedimiento DROP_STORAGEMGMT_TABLES | SYSPROC | Este procedimiento inactiva las tablas de gestión de almacenamiento. |

Tabla 17. Rutinas administrativas SQL de búsqueda de texto

| Nombre de la rutina | Esquema | Descripción |
|---|---------|---|
| Procedimiento almacenado SYSTS_ADMIN_CMD | SYSPROC | Este procedimiento ejecuta mandatos administrativos de búsqueda de texto utilizando la sentencia SQL CALL. |
| Procedimiento SYSTS_ALTER | SYSPROC | Este procedimiento cambia las características de actualización de un índice. |
| Procedimiento SYSTS_CLEAR_COMMANDLOCKS | SYSPROC | Este procedimiento elimina todos los bloqueos de mandatos para un índice de búsqueda de texto específico o para todos los índices de búsqueda de texto en la base de datos. |
| Procedimiento SYSTS_CLEAR_EVENTS | SYSPROC | Este procedimiento suprime los sucesos de indexación de una tabla de sucesos de índice utilizados para la administración. |
| SYSTS_CREATE, procedimiento | SYSPROC | Este procedimiento crea un índice de búsqueda de textos para una columna de texto que permite que se efectúen búsquedas en los datos de la columna utilizando las funciones de búsqueda de texto. |
| SYSTS_DISABLE, procedimiento | SYSPROC | Este procedimiento inhabilita la búsqueda de texto de DB2 para la base de datos actual. |
| Procedimiento SYSTS_DROP | SYSPROC | Este procedimiento descarta un índice de búsqueda de texto existente asociado a cualquier columna de tabla. |
| Procedimiento SYSTS_ENABLE | SYSPROC | Este procedimiento debe emitirse de modo satisfactorio antes de que puedan crearse índices de búsqueda de texto en las columnas de las tablas de la base de datos. |
| Procedimiento SYSTS_UPDATE | SYSPROC | Este procedimiento actualiza el índice de búsqueda de textos para reflejar el contenido actual de las columnas de texto a las que está asociado el índice. |

Tabla 18. Rutinas administrativas SQL de gestión de carga de trabajo

| Nombre de la rutina | Esquema | Descripción |
|---|---------|--|
| procedimiento WLM_CANCEL_ACTIVITY | SYSPROC | Este procedimiento cancela la actividad dada. |
| procedimiento WLM_CAPTURE_ACTIVITY_IN_PROGRESS | SYSPROC | Este procedimiento envía información de una actividad determinada al supervisor de eventos de actividades. |
| WLM_COLLECT_STATS, procedimiento | SYSPROC | Este procedimiento envía estadísticas para clases de servicio, cargas de trabajo, clases de trabajo y colas de umbral al supervisor de eventos de estadísticas y restaura la copia en memoria de las estadísticas. |
| función de tabla WLM_GET_QUEUE_STATS | SYSPROC | Esta función de tabla devuelve información estadística básica de una o más colas de umbral. |
| WLM_GET_SERVICE_CLASS_AGENTS_V97, función de tabla | SYSPROC | Esta función de tabla devuelve la lista de agentes en la partición dada que ejecutan en la clase de servicio SERVICE_SUPERCLASS_NAME y SERVICE_SUBCLASS_NAME o en nombre de la aplicación dada por APPLICATION_HANDLE. |
| Función de tabla WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES_V97 | SYSPROC | Esta función de tabla devuelve la lista de todas las ocurrencias de carga de trabajo que se ejecutan en una determinada clase de servicio de una determinada partición. |
| Función de tabla WLM_GET_SERVICE_SUBCLASS_STATS_V97 | SYSPROC | Esta función de tabla devuelve las estadísticas básicas de una o más subclases de servicio. |
| función de tabla WLM_GET_SERVICE_SUPERCLASS_STATS | SYSPROC | Esta función de tabla devuelve las estadísticas básicas de una o más superclases de servicio. |
| función de tabla WLM_GET_WORK_ACTION_SET_STATS | SYSPROC | Esta función de tabla devuelve las estadísticas básicas de clases de trabajo en un conjunto de acciones de trabajo. |
| Función de tabla WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES_V97 | SYSPROC | Esta función de tabla devuelve la lista de todas las actividades enviadas a través de la aplicación especificada de la partición especificada y que aún no se han completado. |
| Función de tabla WLM_GET_WORKLOAD_STATS_V97 | SYSPROC | Esta función de tabla devuelve las estadísticas básicas de una o más cargas de trabajo. |
| WLM_SET_CLIENT_INFO, procedimiento | SYSPROC | Este procedimiento establece información de cliente asociada con la conexión actual en el servidor DB2. |

Tabla 19. Vistas y rutinas administrativas SQL varias

| Nombre de la vista o rutina | Esquema | Descripción |
|----------------------------------|---------|--|
| ADMIN_COPY_SCHEMA, procedimiento | SYSPROC | Este procedimiento sirve para copiar un esquema específico y todos los objetos contenidos en el mismo. |

Tabla 19. Vistas y rutinas administrativas SQL varias (continuación)

| Nombre de la vista o rutina | Esquema | Descripción |
|--|--|--|
| ADMIN_DROP_SCHEMA, procedimiento | SYSPROC | Este procedimiento sirve para desactivar un esquema específico y todos los objetos contenidos en el mismo. |
| ADMIN_MOVE_TABLE, procedimiento | SYSPROC | Este procedimiento mueve datos de una tabla activa a un objeto de tabla nuevo que tiene el mismo nombre, y durante el proceso los datos continúan en línea y se puede acceder a ellos. |
| ADMIN_MOVE_TABLE_UTIL, procedimiento | SYSPROC | Este procedimiento modifica los valores definibles por el usuario empleados por el procedimiento ADMIN_MOVE_TABLE. |
| ALTOBJ, procedimiento | SYSPROC | Este procedimiento altera una tabla existente utilizando la sentencia CREATE TABLE de entrada como la definición de tabla de destino. |
| APPLICATION_ID, función escalar | SYSFUN | Esta función escalar devuelve el ID de aplicación de la conexión actual. |
| COMPILATION_ENV, función de tabla | SYSPROC | Esta función de tabla devuelve los elementos de un entorno de compilación. |
| CONTACTGROUPS, vista administrativa | SYSIBMADM | Esta vista administrativa devuelve información acerca de la lista de contactos. |
| CONTACTS, vista administrativa | SYSIBMADM | La vista administrativa devuelve la lista de contactos definidos en el servidor de bases de datos. |
| DB_HISTORY, vista administrativa | SYSIBMADM | Esta administrativa devuelve información del archivo histórico asociado a la partición de base de datos conectada actualmente. |
| DBPATHS, vista administrativa | SYSIBMADM | Esta vista administrativa devuelve los valores de las vías de acceso de bases de datos necesarias para tareas como dividir copias de seguridad duplicadas. |
| EXPLAIN_FORMAT_STATS, función escalar | SYSPROC | La nueva función escalar se usa para mostrar información estadística formateada analizada y extraída de una instantánea de explicación capturada para una consulta determinada. |
| Función de tabla EXPLAIN_GET_MSGS | El esquema es el mismo que el de la tabla Explain. | Esta función de tabla consulta las tablas Explain de EXPLAIN_DIAGNOSTIC y EXPLAIN_DIAGNOSTIC_DATA y devuelve mensajes con formato. |
| GET_DBSIZE_INFO, procedimiento | SYSPROC | Este procedimiento calcula el tamaño de base de datos y la capacidad máxima. |
| NOTIFICATIONLIST, vista administrativa | SYSIBMADM | Esta vista administrativa devuelve la lista de contactos y grupos de contactos a los que se notifica sobre la salud de una instancia. |
| función de tabla PD_GET_DIAG_HIST | SYSPROC | La función de tabla devuelve registros de anotación cronológica, registros de sucesos y registros de notificación de un recurso determinado. |

Tabla 19. Vistas y rutinas administrativas SQL varias (continuación)

| Nombre de la vista o rutina | Esquema | Descripción |
|---|---|--|
| Vista administrativa PDLOGMSGs_LAST24HOURS y función de tabla PD_GET_LOG_MSGS | SYSIBMADM (vista administrativa), SYSPROC (función de tabla) | La vista administrativa y la función de tabla devuelven mensajes de anotaciones cronológicas de determinación de problemas que se han registrado en las anotaciones cronológicas de notificaciones de DB2. La información está destinada a los administradores de bases de datos y del sistema. |
| REORGCHK_IX_STATS, procedimiento | SYSPROC | Este procedimiento comprueba las estadísticas del índice para determinar si es o no es necesaria una reorganización. |
| REORGCHK_TB_STATS, procedimiento | SYSPROC | Este procedimiento comprueba las estadísticas de la tabla para determinar si es o no es necesaria una reorganización. |
| SQLERRM, función escalar | SYSPROC | Esta función escalar tiene dos versiones. La primera ofrece flexibilidad completa en la recuperación de mensajes, incluidos símbolos de mensajes y selección de idioma. La segunda es una sencilla interfaz que toma únicamente un SQLCODE como parámetro de entrada y devuelve el mensaje abreviado en inglés. |
| SYSINSTALLOBJECTS, procedimiento | SYSPROC | El procedimiento crea o descarta los objetos de base de datos requeridos para una herramienta específica. |

Capítulo 3. Rutinas del supervisor de actividades

AM_BASE_RPT_RECOMS – Recomendaciones para informes de actividad

La función de tabla AM_BASE_RPT_RECOMS devuelve recomendaciones para los informes de actividad utilizados por el supervisor de actividad.

Sintaxis

►► AM_BASE_RPT_RECOMS (—*id-informe*—, —*entorno-local-cliente*—) ◀◀

El esquema es SYSPROC.

Parámetros de la función de tabla

id-informe

Argumento de entrada de tipo INTEGER que especifica un ID de informe. Si el argumento es nulo, se devuelven recomendaciones para todos los informes disponibles.

entorno-local-cliente

Argumento de entrada de tipo VARCHAR(33) que especifica el identificador del idioma del cliente. Si el argumento es nulo o una serie vacía, el valor por omisión es 'En_US' (inglés). Si los archivos de mensajes del entorno local especificado no están disponibles en el servidor, se utilizará 'En_US'.

Autorización

Privilegio EXECUTE para la función de tabla AM_BASE_RPT_RECOMS.

Ejemplos

Ejemplo 1: Solicite recomendaciones (en inglés) para el informe del supervisor de actividad con el ID n° 1. Suponga que el identificador del idioma del cliente por omisión es 'En_US'.

```
SELECT *  
FROM TABLE(SYSPROC.AM_BASE_RPT_RECOMS(1, CAST(NULL AS VARCHAR(33))))  
AS RECOMS
```

Ejemplo 2: Solicite recomendaciones (en francés) para el informe del supervisor de actividad con el ID n° 12.

```
SELECT *  
FROM TABLE(SYSPROC.AM_BASE_RPT_RECOMS(12, CAST('Fr_FR' AS VARCHAR(33))))  
AS RECOMS
```

Información devuelta

Tabla 20. Información devuelta por la función de tabla AM_BASE_RPT_RECOMS

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|-------------------|
| REPORT_ID | INTEGER | El ID de informe. |

Tabla 20. Información devuelta por la función de tabla
AM_BASE_RPT_RECOMS (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| RECOM_NAME | VARCHAR(256) | El nombre o descripción abreviada de la recomendación. |
| RECOM_DESCRIPTION | CLOB(32K) | La descripción detallada de la recomendación. |

AM_BASE_RPTS – Informes del supervisor de actividad

La función de tabla AM_BASE_RPTS devuelve informes de actividad utilizados por el supervisor de actividad.

Sintaxis

►► AM_BASE_RPTS—(—*id-informe*—, —*tipo*—, —*entorno-local-cliente*—)◀◀

El esquema es SYSPROC.

Parámetros de la función de tabla

id-informe

Argumento de entrada de tipo INTEGER que especifica un ID de informe exclusivo. Si el argumento es nulo, se devolverán informes con cualquier ID de informe.

tipo

Argumento de entrada de tipo CHAR(4) que especifica el tipo de informe. Los valores válidos son:

APPL Aplicación

STMT Sentencia de SQL

TRAN Transacción

CACH Antememoria de sentencias de SQL dinámico

Los valores se pueden especificar en caracteres en mayúsculas o minúsculas. Si el argumento es nulo o una serie vacía, se devuelven informes de cualquier tipo.

entorno-local-cliente

Argumento de entrada de tipo VARCHAR(33) que especifica el identificador del idioma del cliente. Si el argumento es nulo o una serie vacía o los archivos de mensajes del entorno local especificado no están disponibles en el servidor, se utiliza 'En_US'.

Autorización

Privilegio EXECUTE para la función de tabla AM_BASE_RPTS.

Ejemplos

Ejemplo 1:

```
SELECT * FROM TABLE(SYSPROC.AM_BASE_RPTS(CAST(NULL AS INTEGER),
CAST(NULL AS CHAR(4)), CAST(NULL AS VARCHAR(33)))) AS REPORTS
```

Ejemplo 2:

```
SELECT ID, NAME FROM TABLE(SYSPROC.AM_BASE_RPTS(
CAST(NULL AS INTEGER), CAST('STMT' AS CHAR(4)), 'En_US'))
AS REPORTS WHERE TYPE = 'STMT'
```

Información devuelta

Tabla 21. Información devuelta por la función de tabla AM_BASE_RPTS

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|----------------|--|
| ID | INTEGER | El ID de informe exclusivo. |
| TYPE | CHAR(4) | El tipo de informe. Los valores válidos son: APPL, STMT, TRAN, CACH. |
| NAME | VARCHAR(256) | El nombre o descripción abreviada del informe. |
| DESCRIPTION | VARCHAR(16384) | La descripción detallada del informe. |
| SWITCHES | VARCHAR(100) | Los conmutadores de supervisor necesarios para este informe. |

RAM_DROP_TASK – Supresión de una tarea de supervisión

El procedimiento AM_DROP_TASK suprime una tarea de supervisión. No devuelve ningún dato.

Sintaxis

►►—AM_DROP_TASK—(—*id-tarea*—)—————►►

El esquema es SYSPROC.

Parámetro del procedimiento

id-tarea

Argumento de entrada de tipo INTEGER que especifica un ID de tarea de supervisión exclusivo.

Autorización

Privilegio EXECUTE para el procedimiento AM_DROP_TASK.

Ejemplo

Descartar la tarea de supervisión con el ID 5.

```
CALL SYSPROC.AM_DROP_TASK(5)
```

RAM_GET_LOCK_CHN_TB – Recuperación de los datos de la cadena de bloqueo de la aplicación en formato tabular

El procedimiento AM_GET_LOCK_CHN_TB devuelve datos de cadena de bloqueo de aplicación en formato tabular. Una cadena de bloqueo se compone de todas las aplicaciones que mantiene o espera la aplicación actual, directa o indirectamente.

Sintaxis

►—AM_GET_LOCK_CHN_TB—(—*id-agente*—)——►

El esquema es SYSPROC.

Parámetros del procedimiento

id-agente

Argumento de entrada de tipo BIGINT que especifica el ID de agente de la aplicación para la que se deben recuperar datos de cadena de bloqueo.

Autorización

- Autorización SYSMON
- Privilegio EXECUTE para el procedimiento AM_GET_LOCK_CHN_TB.

Ejemplo

Recuperar información de cadena de bloqueo para el ID de agente 68.

```
CALL SYSPROC.AM_GET_LOCK_CHN_TB(68)
```

Información devuelta

El procedimiento devuelve una tabla tal como se muestra más abajo. Cada fila de la tabla representa una relación de bloqueo-espera. El conjunto de resultados también contiene una fila para cada aplicación de sólo mantenimiento; en este caso, la columna HOLDING_AGENT_ID es nula y las otras cuatro columnas son para la aplicación de sólo mantenimiento.

Tabla 22. Información devuelta por el procedimiento AM_GET_LOCK_CHN_TB

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| HOLDING_AGENT_ID | BIGINT | El ID de agente de la aplicación que mantiene el bloqueo. |
| AGENT_ID | BIGINT | El ID de agente de la aplicación que espera el bloqueo. |
| APPL_NAME | VARCHAR(255) | El nombre de la aplicación que espera el bloqueo. |
| AUTH_ID | VARCHAR(128) | El ID de autorización de la aplicación que espera el bloqueo. |
| APPL_ID | VARCHAR(64) | El ID de la aplicación que espera el bloqueo. |

RAM_GET_LOCK_CHNS – Recuperación de información de bloqueo de cadenas para una aplicación específica

El procedimiento AM_GET_LOCK_CHNS devuelve cadenas de bloqueo para la aplicación especificada como una serie formateada. Una cadena de bloqueo se compone de todas las aplicaciones que mantiene o espera la aplicación actual, directa o indirectamente.

Sintaxis

▶▶—AM_GET_LOCK_CHNS—(—*id-agente*—,—*cadenas-bloqueo*—)————▶▶

El esquema es SYSPROC.

Parámetros del procedimiento

id-agente

Argumento de entrada de tipo BIGINT que especifica el ID de agente de la aplicación cuyas cadenas de bloqueo se deben visualizar.

cadenas-bloqueo

Argumento de salida de tipo CLOB(2M) que muestra todas las cadenas de bloqueo para la aplicación especificada.

Autorización

- Autorización SYSMON
- Privilegio EXECUTE para el procedimiento AM_GET_LOCK_CHNS.

Ejemplo

```
CALL SYSPROC.AM_GET_LOCK_CHNS(17,?)
```

```
Valor parámetros de salida
```

```
-----  
Nombre de parámetro : LOCK_CHAINS
```

```
Valor de parámetro  : >db2bp.exe (Agent ID: 17) (Auth ID: AMUSERC )
```

```
<db2bp.exe (Agent ID: 17) (Auth ID: AMUSERC )
```

```
<db2bp.exe (Agent ID: 18) (Auth ID: AMUSERB )
```

```
<db2bp.exe (Agent ID: 16) (Auth ID: AMUSERA )
```

```
Estado de devolución = 0
```

RAM_GET_LOCK_RPT – Recuperación de los detalles del bloqueo de la aplicación

El procedimiento AM_GET_LOCK_RPT devuelve detalles relacionados con el bloqueo para una aplicación en tres conjuntos de resultados de salida.

Sintaxis

▶▶—AM_GET_LOCK_RPT—(—*id-agente*—)————▶▶

El esquema es SYSPROC.

Parámetro del procedimiento

id-agente

Argumento de entrada de tipo BIGINT que especifica el ID de agente de la aplicación cuyos detalles de bloqueo se deben devolver.

Autorización

- Autorización SYSMON
- Privilegio EXECUTE para el procedimiento AM_GET_LOCK_RPT.

Ejemplo

```
CALL SYSPROC.AM_GET_LOCK_RPT(68)
```

Nota sobre uso

El conmutador de supervisor DFT_MON_LOCK debe estar activado para que este procedimiento devuelva información.

Información devuelta

El procedimiento devuelve tres conjuntos de resultados: uno para información general de la aplicación; otro para los bloqueos mantenidos por la aplicación y otro para los bloqueos que la aplicación está esperando.

Tabla 23. Información general de aplicación devuelta por el procedimiento AM_GET_LOCK_RPT

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| AGENT_ID | BIGINT | agent_id - Descriptor de contexto de aplicación (ID de agente) |
| APPL_NAME | VARCHAR(256) | appl_name - Nombre de aplicación |
| AUTH_ID | VARCHAR(128) | auth_id - ID de autorización |
| APPL_ID | VARCHAR(128) | appl_id - ID de aplicación |

Tabla 23. Información general de aplicación devuelta por el procedimiento AM_GET_LOCK_RPT (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|---------------------|---------------|--|
| APPL_STATUS | VARCHAR(22) | <p>appl_status - Estado de aplicación. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h, y es uno de los siguientes:</p> <ul style="list-style-type: none"> • BACKUP • COMMIT_ACT • COMP • CONNECTED • CONNECTPEND • CREATE_DB • DECOUPLED • DISCONNECTPEND • INTR • IOERROR_WAIT • LOAD • LOCKWAIT • QUIESCE_TABLESPACE • RECOMP • REMOTE_RQST • RESTART • RESTORE • ROLLBACK_ACT • ROLLBACK_TO_SAVEPOINT • TEND • THABRT • THCOMT • TPREP • UNLOAD • UOWEXEC • UOWWAIT • WAITFOR_REMOTE |
| COORD_PARTITION_NUM | SMALLINT | coord_node - Nodo de coordinación |
| SEQUENCE_NO | VARCHAR(4) | sequence_no - Número de secuencia |
| CLIENT_PRDID | VARCHAR(128) | client_prdid - ID de producto/versión de cliente |
| CLIENT_PID | BIGINT | client_pid - ID de proceso de cliente |

Tabla 23. Información general de aplicación devuelta por el procedimiento AM_GET_LOCK_RPT (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| CLIENT_PLATFORM | VARCHAR(12) | <p>client_platform - Plataforma operativa de cliente. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h,</p> <ul style="list-style-type: none"> • AIX • AIX64 • AS400_DRDA • DOS • DYNIX • HP • HP64 • HPIA • HPIA64 • LINUX • LINUX390 • LINUXIA64 • LINUXPPC • LINUXPPC64 • LINUXX8664 • LINUXZ64 • MAC • MVS_DRDA • NT • NT64 • OS2 • OS390 • SCO • SGI • SNI • SUN • SUN64 • UNKNOWN • UNKNOWN_DRDA • VM_DRDA • VSE_DRDA • WINDOWS • WINDOWS95 |

Tabla 23. Información general de aplicación devuelta por el procedimiento AM_GET_LOCK_RPT (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|----------------------|---------------|--|
| CLIENT_PROTOCOL | VARCHAR(10) | client_protocol - Protocolo de comunicaciones de cliente. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h, <ul style="list-style-type: none"> • CPIC • LOCAL • NETBIOS • NPIPE • TCPIP (para DB2 Universal Database o DB2 UDB) • TCPIP4 • TCPIP6 |
| CLIENT_NNAME | VARCHAR(128) | El elemento del supervisor client_nname ha quedado obsoleto. El valor devuelto no es un valor válido. |
| LOCKS_HELD | BIGINT | locks_held - Bloqueos retenidos |
| LOCK_WAIT_START_TIME | TIMESTAMP | lock_wait_start_time - Indicación de fecha y hora de inicio de espera de bloqueo |
| LOCK_WAIT_TIME | BIGINT | lock_wait_time - Tiempo esperado en bloqueos |
| LOCK_WAITS | BIGINT | lock_waits - Esperas de bloqueo |
| LOCK_TIMEOUTS | BIGINT | lock_timeouts - Número de tiempos de espera de bloqueo |
| LOCK_ESCALS | BIGINT | lock_escals - Número de escalamientos de bloqueo |
| X_LOCK_ESCALS | BIGINT | x_lock_escals - Escalas de bloqueo exclusivas |
| DEADLOCKS | BIGINT | deadlocks - Puntos muertos detectados |

Tabla 24. Información retenida por los bloqueos devuelta por el procedimiento AM_GET_LOCK_RPT

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| TBSP_NAME | VARCHAR(128) | tablespace_name - Nombre de espacio de tablas |
| TABSCHEMA | VARCHAR(128) | table_schema - Nombre de esquema de tabla |
| TABNAME | VARCHAR(128) | table_name - Nombre de tabla |

Tabla 24. Información retenida por los bloqueos devuelta por el procedimiento AM_GET_LOCK_RPT (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| LOCK_OBJECT_TYPE | VARCHAR(18) | <p>lock_object_type - Tipo de objeto de bloqueo esperado. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes:</p> <ul style="list-style-type: none"> • AUTORESIZE_LOCK • AUTOSTORAGE_LOCK • BLOCK_LOCK • EOT_LOCK • INPLACE_REORG_LOCK • INTERNAL_LOCK • INTERNALB_LOCK • INTERNALC_LOCK • INTERNALJ_LOCK • INTERNALL_LOCK • INTERNALO_LOCK • INTERNALQ_LOCK • INTERNALP_LOCK • INTERNALS_LOCK • INTERNALT_LOCK • INTERNALV_LOCK • KEYVALUE_LOCK • ROW_LOCK • SYSBOOT_LOCK • TABLE_LOCK • TABLE_PART_LOCK • TABLESPACE_LOCK • XML_PATH_LOCK |
| LOCK_MODE | VARCHAR(10) | <p>lock_mode - Modalidad de bloqueo. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes:</p> <ul style="list-style-type: none"> • IN • IS • IX • NON (si no existe bloqueo) • NS • NW • S • SIX • U • X • Z |

Tabla 24. Información retenida por los bloqueos devuelta por el procedimiento AM_GET_LOCK_RPT (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| LOCK_STATUS | VARCHAR(10) | lock_status - Estado de bloqueo. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • CONV • GRNT |
| LOCK_ESCALATION | SMALLINT | lock_escalation - Escalamiento de bloqueos |
| LOCK_NAME | VARCHAR(32) | lock_name - Nombre de bloqueo |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Tabla 25. Información de espera de los bloqueos devuelta por el procedimiento AM_GET_LOCK_RPT

| Nombre de columna | Tipo de datos | Descripción |
|----------------------|---------------|---|
| AGENT_ID_HOLDING_LK | BIGINT | agent_id_holding_lock - ID de agente que mantiene el bloqueo |
| APPL_ID_HOLDING_LK | VARCHAR(128) | appl_id_holding_lk - ID de aplicación que mantiene el bloqueo |
| LOCK_WAIT_START_TIME | TIMESTAMP | lock_wait_start_time - Indicación de fecha y hora de inicio de espera de bloqueo |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |
| TBSP_NAME | VARCHAR(128) | tablespace_name - Nombre de espacio de tablas |
| TABSCHEMA | VARCHAR(128) | table_schema - Nombre de esquema de tabla |
| TABNAME | VARCHAR(128) | table_name - Nombre de tabla |

Tabla 25. Información de espera de los bloqueos devuelta por el procedimiento AM_GET_LOCK_RPT (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| LOCK_OBJECT_TYPE | VARCHAR(18) | <p>lock_object_type - Tipo de objeto de bloqueo esperado. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes:</p> <ul style="list-style-type: none"> • AUTORESIZE_LOCK • AUTOSTORAGE_LOCK • BLOCK_LOCK • EOT_LOCK • INPLACE_REORG_LOCK • INTERNAL_LOCK • INTERNALB_LOCK • INTERNALC_LOCK • INTERNALJ_LOCK • INTERNALL_LOCK • INTERNALO_LOCK • INTERNALQ_LOCK • INTERNALP_LOCK • INTERNALS_LOCK • INTERNALT_LOCK • INTERNALV_LOCK • KEYVALUE_LOCK • ROW_LOCK • SYSBOOT_LOCK • TABLE_LOCK • TABLE_PART_LOCK • TABLESPACE_LOCK • XML_PATH_LOCK |
| LOCK_MODE | VARCHAR(10) | <p>lock_mode - Modalidad de bloqueo. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes:</p> <ul style="list-style-type: none"> • IN • IS • IX • NON (si no existe bloqueo) • NS • NW • S • SIX • U • X • Z |

Tabla 25. Información de espera de los bloqueos devuelta por el procedimiento AM_GET_LOCK_RPT (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|---------------------|---------------|---|
| LOCK_MODE_REQUESTED | VARCHAR(10) | lock_mode_requested - Modalidad de bloqueo solicitada. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • IN • IS • IX • NON (si no existe bloqueo) • NS • NW • S • SIX • U • X • Z |
| LOCK_ESCALATION | SMALLINT | lock_escalation - Escalamiento de bloqueos |

RAM_GET_RPT – Recuperación de datos del supervisor de actividad

El procedimiento AM_GET_RPT devuelve datos del supervisor de actividad para un informe.

Sintaxis

```

▶▶ AM_GET_RPT (—partición de base de datos—, —id-informe—, —filtro-apl—, —
▶—número-máx—)

```

El esquema es SYSPROC.

Parámetros del procedimiento

partición de base de datos

Argumento de entrada de tipo INTEGER que especifica un número de partición de base de datos. Los valores correctos son -2 (que indica todas las particiones de base de datos) y el número de partición de base de datos de cualquier partición de base de datos existente.

id-informe

Argumento de entrada de tipo INTEGER que especifica un ID de informe exclusivo.

filtro-apl

Argumento de entrada de tipo CLOB(32K) que especifica un filtro de aplicaciones. Un filtro de aplicaciones es una condición de búsqueda que implica cualquiera de las tres columnas AGENT_ID, APPL_NAME y

AUTH_ID, o todas ellas, donde AGENT_ID y AUTH_ID son enteros y APPL_NAME es una serie de caracteres. Si el argumento es nulo o una serie vacía, no se realiza ningún filtrado.

número-máx

Argumento de entrada de tipo INTEGER que especifica el número máximo de aplicaciones, sentencias o transacciones que se visualizarán. Si el argumento es nulo, se visualizarán todas las aplicaciones, sentencias y transacciones.

Autorización

- Autorización SYSMON
- Privilegio EXECUTE para el procedimiento AM_GET_RPT.

Ejemplo

```
CALL SYSPROC.AM_GET_RPT(-2, 18,  
    CAST('AGENT_ID=29 AND AUTH_ID <> ''dbuser'' AND APPL_NAME LIKE ''db2%''  
    AS CLOB(32K)), 100)
```

Nota sobre uso

El conjunto de resultados devuelto es diferente por cada ID de informe. El objetivo de este informe es soportar la herramienta gráfica de Supervisor de actividad. Para crear informes que se puedan analizar, deberán utilizarse instantáneas de vistas y rutinas administrativas SQL.

RAM_SAVE_TASK – Creación o modificación de una tarea de supervisión

El procedimiento AM_SAVE_TASK crea o modifica una tarea de supervisión.

Sintaxis

```
▶▶ AM_SAVE_TASK—(—modalidad—,—id-tarea—,—nombre-tarea—,—filtro-apl—,—▶▶  
▶—mostrar-cadenas-bloqueo—,—id-informe—)—————▶▶
```

El esquema es SYSPROC.

Parámetros del procedimiento

modalidad

Argumento de entrada de tipo CHAR(1) que especifica si debe crearse una nueva tarea de supervisión ('C') o modificarse una existente ('M').

id-tarea

Argumento de entrada de tipo INTEGER que especifica un ID de tarea de supervisión exclusivo. Cuando la *modalidad* es 'C', se ignora cualquier entrada especificada para el *id-tarea*. El procedimiento generará un ID para la nueva tarea de supervisión y lo devolverá en la salida. Cuando la *modalidad* es 'M', especifica el ID de la tarea de supervisión que se va a modificar.

nombre-tarea

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre o descripción abreviada para una tarea de supervisión.

filtro-apl

Argumento de entrada de tipo CLOB(32K) que especifica un filtro de aplicaciones. Un filtro de aplicaciones es una condición de búsqueda que implica cualquiera de las tres columnas AGENT_ID, APPL_NAME y AUTH_ID, o todas ellas, donde AGENT_ID y AUTH_ID son enteros y APPL_NAME es una serie de caracteres. Si el argumento es nulo o una serie vacía, no se realiza ningún filtrado.

mostrar-cadenas-bloqueo

Argumento de entrada de tipo CHAR(1) que especifica si se deben mostrar cadenas de bloqueo. Los valores correctos son 'Y' y 'N'. Si el argumento es nulo, no se mostrarán cadenas de bloqueo.

id-informe

Argumento de entrada de tipo VARCHAR(3893) que especifica uno o más ID de informe separados por comas.

Autorización

Privilegio EXECUTE para el procedimiento AM_SAVE_TASK.

Ejemplo

Ejemplo:

```
CALL SYSPROC.AM_SAVE_TASK('M',11,'Task ABC',CAST (NULL AS CLOB(32K)),  
  'N','1,2,4,8,9,12')
```

Capítulo 4. Procedimiento ADMIN_CMD y rutinas asociadas

ADMIN_CMD – Ejecutar mandatos administrativos

Las aplicaciones utilizan el procedimiento ADMIN_CMD para ejecutar mandatos administrativos mediante la sentencia CALL de SQL.

Sintaxis

►►—ADMIN_CMD—(—*serie-mandatos*—)—————►►

El esquema es SYSPROC.

Parámetro del procedimiento

serie-mandato

Argumento de entrada del tipo CLOB (2M) que especifica un único mandato que se debe ejecutar.

Autorización

Privilegio EXECUTE para el procedimiento ADMIN_CMD.

El procedimiento da soporte actualmente a los siguientes mandatos del procesador de línea de mandatos (CLP) de DB2:

- ADD CONTACT
- ADD CONTACTGROUP
- AUTOCONFIGURE
- BACKUP - online only
- DESCRIBE
- DROP CONTACT
- DROP CONTACTGROUP
- EXPORT
- FORCE APPLICATION
- IMPORT
- INITIALIZE TAPE
- LOAD
- PRUNE HISTORY/LOGFILE
- QUIESCE DATABASE
- QUIESCE TABLESPACES FOR TABLE
- REDISTRIBUTE
- REORG INDEXES/TABLE
- RESET ALERT CONFIGURATION
- RESET DATABASE CONFIGURATION
- RESET DATABASE MANAGER CONFIGURATION
- REWIND TAPE

- RUNSTATS
- SET TAPE POSITION
- UNQUIESCE DATABASE
- UPDATE ALERT CONFIGURATION
- UPDATE CONTACT
- UPDATE CONTACTGROUP
- UPDATE DATABASE CONFIGURATION
- UPDATE DATABASE MANAGER CONFIGURATION
- UPDATE HEALTH NOTIFICATION CONTACT LIST
- UPDATE HISTORY

Nota: Es posible que algunos mandatos tengan una sintaxis soportada ligeramente diferente cuando se ejecutan mediante el procedimiento ADMIN_CMD.

El procedimiento también da soporte a los mandatos siguientes que el CLP no soporta:

- GET STMM TUNING DBPARTITIONNUM
- UPDATE STMM TUNING DBPARTITIONNUM

Notas de uso

Recuperación de la información de ejecución del mandato:

- Puesto que el procedimiento ADMIN_CMD se ejecuta en el servidor, también se generan mensajes de utilidad en el servidor. La opción MESSAGES ON SERVER (consulte el mandato específico para obtener más información) indica que el archivo de mensaje se creará en el servidor.
- El estado de ejecución del mandato se devuelve en la SQLCA resultante de la sentencia CALL.
- Si la ejecución del mandato administrativo es satisfactoria y el mandato devuelve más de un estado de ejecución, la información adicional se devuelve en forma de conjunto de resultados (hasta dos conjuntos de resultados). Por ejemplo, si el mandato EXPORT se ejecuta satisfactoriamente, el conjunto de resultados devuelto contiene información sobre el número de filas exportadas; sin embargo, si el mandato RUNSTATS se ejecuta satisfactoriamente, no se devuelve ningún conjunto de resultados. La información del conjunto de resultados se documenta con el mandato correspondiente.
- Si la ejecución del mandato administrativo no es satisfactoria, el procedimiento ADMIN_CMD devuelve un mensaje de aviso SQL20397W con un conjunto de resultados que contiene más detalles sobre el motivo del error del mandato administrativo. Cualquier aplicación que utilice el procedimiento ADMIN_CMD debería comprobar el SQLCODE devuelto por el procedimiento. Si el SQLCODE es ≥ 0 , se debe recuperar el conjunto de resultados del mandato administrativo. La tabla siguiente indica qué información se puede devolver, según si se utiliza la opción MESSAGES ON SERVER.

Tabla 26. *SQLCODE e información devueltos por el procedimiento ADMIN_CMD*

| Estado de ejecución del mandato administrativo | Opción MESSAGES ON SERVER especificada | Opción MESSAGES ON SERVER no especificada |
|--|--|--|
| Satisfactorio | El SQLCODE devuelto es ≥ 0 : Información adicional (conjuntos de resultados) devuelta, si la hay. | El SQLCODE devuelto es ≥ 0 : Información adicional (conjuntos de resultados) devuelta, si la hay, pero las columnas MSG_RETRIEVAL y MSG_REMOVAL son NULL. |
| Error | El SQLCODE ha devuelto 20397: Información adicional (conjuntos de resultados) devuelta, pero sólo las columnas MSG_RETRIEVAL y MSG_REMOVAL están llenas. | El SQLCODE devuelto es < 0 : No se ha devuelto información adicional (conjuntos de resultados). |

- Los conjuntos de resultados se pueden recuperar del CLP o de aplicaciones como JDBC y CLI de DB2, pero no de aplicaciones C incorporadas.
- Los nombres sensibles a las mayúsculas y minúsculas y los nombres de juego de caracteres de doble byte (DBCS) deben incluirse entre una barra inclinada invertida y un delimitador de comillas dobles, por ejemplo, `\ " Mi Tabla \ "`.

En todos los mandatos ejecutados mediante ADMIN_CMD, el ID de usuario que estableció la conexión con la base de datos se utiliza para la autenticación.

Cualquier autorización adicional necesaria, por ejemplo para los mandatos que necesiten acceder al sistema de archivos en el servidor de la base de datos, se documenta en la información de referencia que describe el mandato.

No se puede llamar a este procedimiento desde una función definida por el usuario (SQLSTATE 38001) o un activador.

Mandato ADD CONTACT utilizando el procedimiento ADMIN_CMD

El mandato añade un contacto a la lista de contactos que se puede definir localmente en el sistema o en una lista global. Los contactos son usuarios a los que los procesos, tales como el Planificador y el Supervisor de salud, envían mensajes. El valor del parámetro de configuración **contact_host** del Servidor de administración de bases de datos (DAS) determina si la lista es local o global.

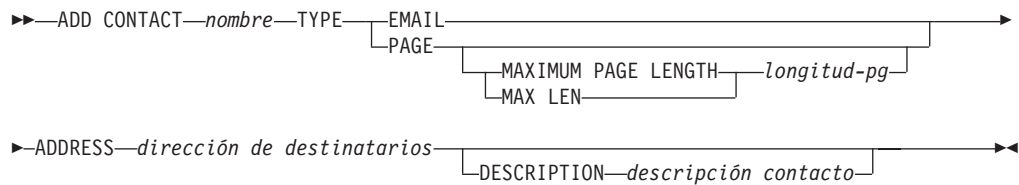
Autorización

Ninguna

Conexión necesaria

Base de datos. El DAS debe estar en ejecución.

Sintaxis del mandato



Parámetros del mandato

ADD CONTACT *nombre*

Nombre del contacto que se añadirá. Por omisión, el contacto se añadirá al sistema local, a menos que el parámetro de configuración **contact_host** del Servidor de administración de DB2 apunte a otro sistema.

TYPE Método de contacto, que debe ser uno de los dos siguientes:

EMAIL

Este contacto desea recibir las notificaciones por correo electrónico en (ADDRESS).

PAGE Este contacto desea recibir las notificaciones mediante un mensaje de buscapersonas enviado a ADDRESS.

MAXIMUM PAGE LENGTH *longitud-pg*

Si el servicio de buscapersonas tiene restringida la longitud de los mensajes, se especifica aquí en caracteres.

El sistema de notificación utiliza el protocolo SMTP para enviar la notificación al servidor de correo especificado por el parámetro de configuración **smtp_server** del Servidor de administración de DB2. Es responsabilidad del servidor SMTP enviar el correo electrónico o llamar al buscapersonas.

ADDRESS *dirección-destinatario*

Dirección del buzón de SMTP del destinatario. Por ejemplo, jose@algunsitio.org. El parámetro de configuración **smtp_server** de DAS debe establecerse como el nombre del servidor SMTP.

DESCRIPTION *descripción contacto*

Texto descriptivo del contacto. Tiene una longitud máxima de 128 caracteres.

Ejemplo

Añadir un contacto para el usuario 'testuser' con la dirección de correo electrónico 'testuser@test.com'.

```
CALL SYSPROC.ADMIN_CMD
('ADD CONTACT testuser TYPE EMAIL ADDRESS testuser@test.com')
```

Notas de uso

El DAS ya se debe haber creado y estar en ejecución.

El estado de la ejecución del mandato se devuelve en el área de comunicaciones SQL (SQLCA) obtenida a partir de la sentencia CALL.

Mandato ADD CONTACTGROUP utilizando el procedimiento ADMIN_CMD

Añade un nuevo grupo de contactos a la lista de los grupos definidos en el sistema local. Un grupo de contacto es una lista de usuarios y grupos a los que los procesos de supervisión, tales como el Planificador y el Supervisor de salud, pueden enviar mensajes. El valor del parámetro de configuración **contact_host** del Servidor de administración de bases de datos (DAS) determina si la lista es local o global.

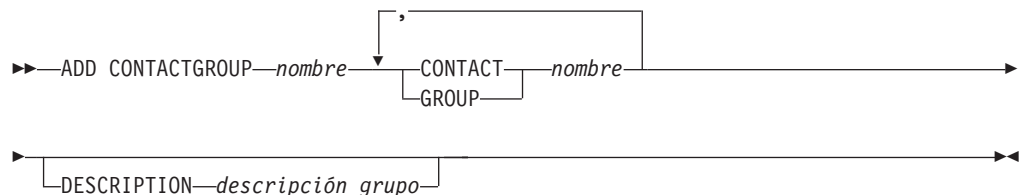
Autorización

Ninguna

Conexión necesaria

Base de datos. El DAS debe estar en ejecución.

Sintaxis del mandato



Parámetros del mandato

ADD CONTACTGROUP *nombre*

Nombre del nuevo grupo de contacto, que debe ser exclusivo entre el conjunto de grupos del sistema.

CONTACT *nombre*

Nombre del contacto que es miembro del grupo. Se puede definir un contacto con el mandato ADD CONTACT después de haberlo añadido a un grupo.

GROUP *nombre*

Nombre del grupo de contactos del que este grupo es miembro.

DESCRIPTION *descripción grupo*

Opcional. Texto descriptivo del grupo de contacto.

Ejemplo

Crear un grupo de contactos llamado 'gname1' que contenga dos contactos: 'cname1' y 'cname2'.

```
CALL SYSPROC.ADMIN_CMD( 'add contactgroup gname1 contact cname1, contact cname2' )
```

Notas de uso

El DAS ya se debe haber creado y estar en ejecución.

El estado de la ejecución del mandato se devuelve en el área de comunicaciones SQL (SQLCA) obtenida a partir de la sentencia CALL.

Mandato AUTOCONFIGURE utilizando el procedimiento ADMIN_CMD

Calcula y visualiza los valores iniciales para los parámetros del tamaño de agrupación de almacenamientos intermedios, de configuración de base de datos y de configuración de gestor de bases de datos, con la opción de aplicar estos valores recomendados.

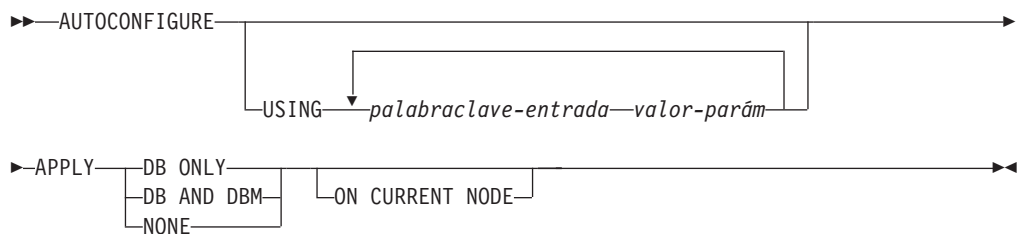
Autorización

sysadm

Conexión necesaria

Base de datos

Sintaxis del mandato



Parámetros del mandato

USING *palabraclave-entrada valor-parám*

Tabla 27. Palabras claves de entrada y valores de parámetro válidos

| Palabra clave | Valores válidos | Valor por omisión | Explicación |
|---------------|-----------------|-------------------|--|
| mem_percent | 1-100 | 25 | Porcentaje inicial de memoria de instancia que se va a dedicar. El Asesor de configuración ajustará este valor para las bases de datos posteriores en función del uso de memoria actual del sistema. |

Tabla 27. Palabras claves de entrada y valores de parámetro válidos (continuación)

| Palabra clave | Valores válidos | Valor por omisión | Explicación |
|-----------------|-----------------------------|-------------------|--|
| workload_type | simple, mixed, complex | mixed | Las cargas de trabajo sencillas tienden a tener una actividad intensiva de E/S y principalmente de transacciones, mientras que las cargas de trabajo complejas tienden a tener una actividad intensiva de CPU y principalmente de consultas. |
| num_stmts | 1-1 000 000 | 10 | Número de sentencias por unidad de trabajo |
| tpm | 1-200 000 | 60 | Transacciones por minuto |
| admin_priority | performance, recovery, both | both | Optimizar para obtener un rendimiento mejor (más transacciones por minuto) o un tiempo de recuperación mejor |
| is_populated | yes, no | yes | ¿Hay datos en la base de datos? |
| num_local_apps | 0-5 000 | 0 | Número de aplicaciones locales conectadas |
| num_remote_apps | 0-5 000 | 10 | Número de aplicaciones remotas conectadas |

Tabla 27. Palabras claves de entrada y valores de parámetro válidos (continuación)

| Palabra clave | Valores válidos | Valor por omisión | Explicación |
|---------------|-----------------|-------------------|---|
| isolation | RR, RS, CS, UR | RR | Nivel máximo de aislamiento de aplicaciones que se conectan con esta base de datos (RR: Lectura repetible; RS: Estabilidad de lectura; CS: Estabilidad de cursor; UR: Lectura no confirmada). Solo se usa para determinar los valores de otros parámetros de configuración. No se establece nada para restringir las aplicaciones para que tengan un determinado nivel de aislamiento y no hay problema en utilizar el valor por omisión. |
| bp_resizeable | yes, no | yes | ¿Puede modificarse el tamaño de las agrupaciones de almacenamientos intermedios? |

APPLY

DB ONLY

Muestra los valores recomendados para la configuración de la base de datos y los valores de la agrupación de almacenamientos intermedios según la configuración actual del gestor de bases de datos. Aplica los cambios recomendados a los valores de configuración de la base de datos y de la agrupación de almacenamientos intermedios.

DB AND DBM

Visualiza y aplica los cambios recomendados a la configuración del gestor de bases de datos, la configuración de la base de datos y los valores de la agrupación de almacenamientos intermedios.

NONE

Visualiza los cambios recomendados, pero no los aplica.

ON CURRENT NODE

En la característica de particionamiento de base de datos (DPF), el asesor de configuración actualiza por omisión la configuración de base de datos en todos los nodos. Si se ejecuta con la opción ON CURRENT NODE, el asesor solo aplica la configuración de base de datos recomendada al nodo (conexión) del coordinador.

Los cambios de la agrupación de almacenamientos intermedios siempre se aplican a los catálogos del sistema. Por lo tanto, todos los nodos se ven

afectados. La opción ON CURRENT NODE no tiene importancia en las recomendaciones de agrupación de almacenamientos intermedios.

Ejemplo

Invocar autoconfigure en una base de datos mediante el procedimiento almacenado ADMIN_CMD.

```
CALL SYSPROC.ADMIN_CMD( 'AUTOCONFIGURE APPLY NONE' )
```

En el siguiente ejemplo se muestra el conjunto de resultados devuelto por el mandato.

| NIVEL | NOMBRE | VALOR | VALOR RECOMENDADO | TIPO DE DATOS |
|-------|-----------------|-------|-------------------|---------------|
| DBM | ASLHEAPSZ | 15 | 15 | BIGINT |
| DBM | FCM_NUM_BUFFERS | 512 | 512 | BIGINT |
| ... | | | | |
| DB | APP_CTL_HEAP_SZ | 128 | 144 | INTEGER |
| DB | APPGROUP_MEM_SZ | 20000 | 14559 | BIGINT |
| ... | | | | |
| BP | IBMDEFAULTBP | 1000 | 164182 | BIGINT |

Notas de uso

- En los sistemas con varias particiones lógicas, el parámetro **mem_percent** hace referencia al porcentaje de la memoria que van a utilizar todas las particiones lógicas. Por ejemplo, si DB2 utiliza un 25% de la memoria del sistema, especifique 25% sin tener en cuenta el número de particiones lógicas. Sin embargo, las recomendaciones de configuración de la base de datos efectuadas se ajustarán para una partición lógica.
- Este mandato efectúa recomendaciones de configuración para la base de datos conectada en la actualidad, suponiendo que la base de datos sea la única base de datos activa en el sistema. Si hay más de una base de datos activa en el sistema, ajuste el parámetro **mem_percent** para que refleje la participación en la memoria de la base de datos actual. Por ejemplo, si la base de datos DB2 utiliza un 80% de la memoria del sistema y hay dos bases de datos activas en el sistema que deberían compartir los recursos por igual, especifique 40% (80% dividido por 2 bases de datos) para el parámetro **mem_percent**.
- Al invocar explícitamente el asesor de configuración con el mandato AUTOCONFIGURE, no se tendrá en cuenta el valor de la variable de registro DB2_ENABLE_AUTOCONFIG_DEFAULT.
- La ejecución del mandato AUTOCONFIGURE en una base de datos recomendará la habilitación del gestor de memoria de ajuste automático. Sin embargo, si el mandato AUTOCONFIGURE se ejecuta en una base de datos en una instancia donde SHEAPTHRES no es cero, la opción de ajuste de memoria de clasificación (SORTHEAP) no se habilitará automáticamente. Para habilitar el ajuste de memoria de clasificación (SORTHEAP), debe establecer que SHEAPTHRES sea igual a cero con el mandato UPDATE DATABASE MANAGER CONFIGURATION. Tenga en cuenta que el hecho de cambiar el valor de SHEAPTHRES puede afectar al uso de memoria de clasificación en las bases de datos que existían con anterioridad.
- El estado de la ejecución del mandato se devuelve en el área de comunicaciones SQL (SQLCA) obtenida a partir de la sentencia CALL.
- Query Patroller supervisa el SQL ejecutado en el procedimiento ADMIN_CMD en nombre del mandato AUTOCONFIGURE.

- El mandato AUTOCONFIGURE emite una sentencia COMMIT al final de su ejecución. En el caso de las conexiones de tipo 2, ello hará que el procedimiento ADMIN_CMD devuelva SQL30090N con el código de razón 2.

Información de conjunto de resultados

El estado de la ejecución del mandato se devuelve en el área de comunicaciones SQL (SQLCA) obtenida a partir de la sentencia CALL. Si la ejecución es satisfactoria, el mandato devuelve información adicional en el siguiente conjunto de resultados:

Tabla 28. Conjunto de resultados devuelto por el mandato AUTOCONFIGURE

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| LEVEL | VARCHAR(3) | Nivel del parámetro y uno de estos valores: <ul style="list-style-type: none"> • BP para nivel de agrupación de almacenamientos intermedios • DBM para nivel de gestor de bases de datos • DB para nivel de base de datos |
| NAME | VARCHAR(128) | <ul style="list-style-type: none"> • Si LEVEL es DB o DBM, este valor contiene la palabra clave de parámetro de configuración. • Si LEVEL es BP, este valor contiene el nombre de agrupación de almacenamientos intermedios. |
| VALUE | VARCHAR(256) | <ul style="list-style-type: none"> • Si LEVEL es DB o DBM, y se aplicaron los valores recomendados, esta columna contiene el valor del parámetro de configuración identificado en la columna NAME antes de aplicar el valor recomendado (es decir, contiene el valor antiguo). Si el cambio no se aplicó, esta columna contiene el valor en disco actual (valor diferido) del parámetro de configuración identificado. • Si LEVEL es BP, y se aplicaron los valores recomendados, esta columna contiene el tamaño (en páginas) de la agrupación de almacenamientos intermedios identificada en la columna NAME antes de aplicar el valor recomendado (es decir, contiene el tamaño antiguo). Si el cambio no se aplicó, esta columna contiene el tamaño actual (en páginas) de la agrupación de almacenamientos intermedios identificada. |

Tabla 28. Conjunto de resultados devuelto por el mandato AUTOCONFIGURE (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| RECOMMENDED_VALUE | VARCHAR(256) | <ul style="list-style-type: none"> • Si LEVEL es DB o DBM, esta columna contiene el valor recomendado (o aplicado) del parámetro de configuración identificado en la columna del parámetro. • Si el tipo es BP, esta columna contiene el tamaño recomendado (o aplicado), en páginas, de la agrupación de almacenamientos intermedios identificada en la columna del parámetro. |
| DATATYPE | VARCHAR(128) | Tipo de datos del parámetro. |

Mandato BACKUP DATABASE utilizando el procedimiento ADMIN_CMD

Crea una copia de seguridad de una base de datos o de un espacio de tablas.

Para obtener información sobre las operaciones de copia de seguridad soportadas por sistemas de base de datos de DB2 entre diferentes sistemas operativos y plataformas de hardware, consulte “Operaciones de copia de seguridad y restauración entre diferentes sistemas operativos y plataformas de hardware.”

Ámbito

En un entorno de bases de datos particionadas, si no se especifica ninguna partición de base de datos, este mandato sólo afecta a la partición de base de datos en la que se ejecuta.

Si se especifica la opción de realizar una copia de seguridad particionada, sólo se puede llamar al mandato en el nodo de catálogo. Si la opción especifica que se debe hacer copia de seguridad de todos los servidores de particiones de base de datos, afecta a todos los servidores de particiones de base de datos listados en el archivo db2nodes.cfg. De lo contrario, afecta a los servidores de particiones de base de datos que se especifican en el mandato.

Autorización

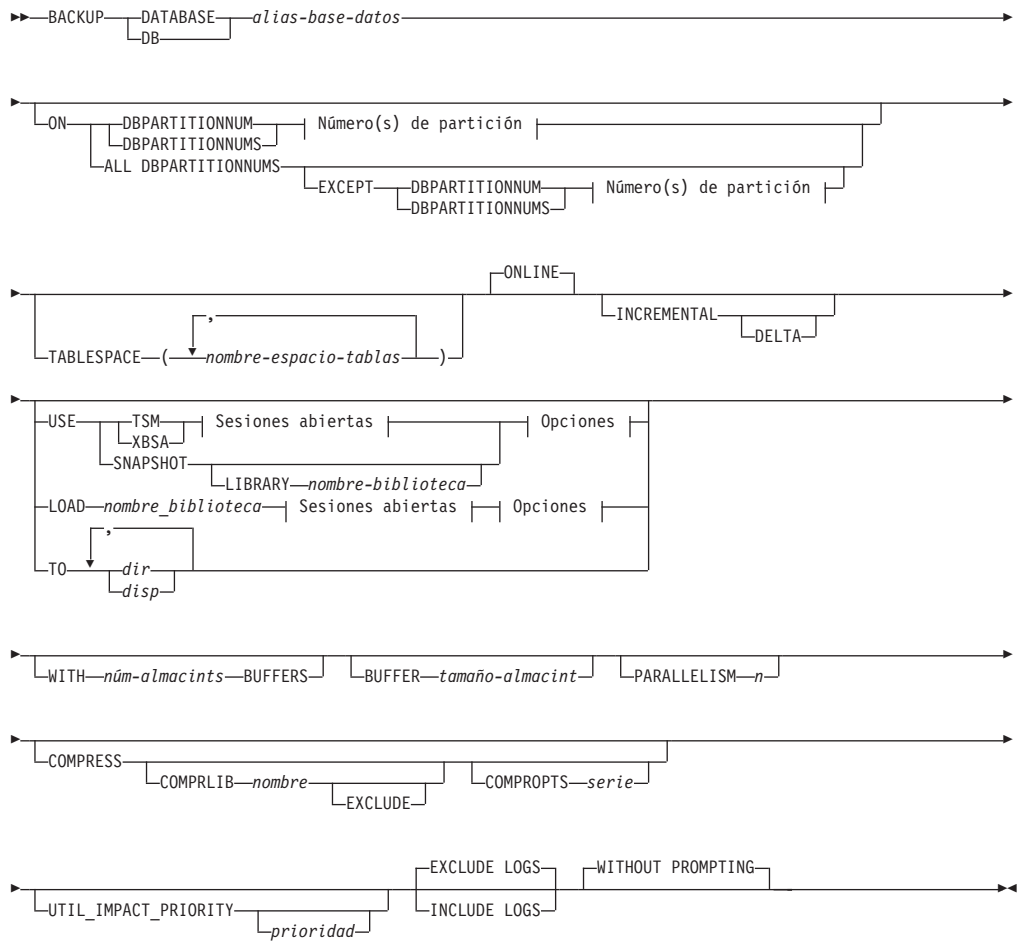
Una de las siguientes:

- *sysadm*
- *sysctrl*
- *sysmaint*

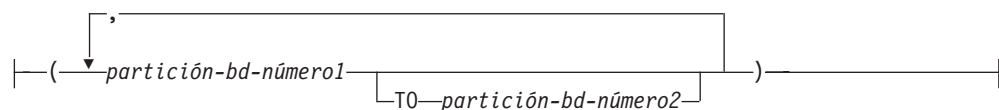
Conexión necesaria

Base de datos. La conexión de base de datos existente terminará después de que termine la operación de copia de seguridad.

Sintaxis del mandato



Número(s) de partición:



Sesiones abiertas:



Opciones:



Parámetros del mandato

DATABASE | DB *alias-base-datos*

Especifica el alias de la base de datos de la que se debe hacer copia de seguridad. El alias debe ser una base de datos local definida en el servidor y debe ser el nombre de la base de datos a la que el usuario está conectado actualmente. Si el alias-base-datos no es aquel al que el usuario está conectado, se devuelve un error SQL20322N.

ON Realiza la copia de seguridad de la base de datos en un conjunto de particiones de base de datos. Esta cláusula debe especificarse en la partición de catálogo.

DBPARTITIONNUM *número1-partición-bd*

Especifica un número de partición de base de datos en la lista de particiones de base de datos.

DBPARTITIONNUMS *número1-partición-bd TO número2-partición-bd*

Especifica un rango de números de particiones de base de datos para que todas las particiones desde *número1-partición-bd1* hasta *número2-partición-bd* inclusive se incluyan en la lista de particiones de base de datos.

ALL DBPARTITIONNUMS

Especifica que la operación de copia de seguridad de la base de datos ha de realizarse en todas las particiones especificadas en el archivo `db2nodes.cfg`.

EXCEPT

Especifica que la operación de copia de seguridad de la base de datos ha de realizarse en todas las particiones especificadas en el archivo `db2nodes.cfg`, excepto aquellas especificadas en la lista de particiones de base de datos.

DBPARTITIONNUM *número1-partición-bd*

Especifica un número de partición de base de datos en la lista de particiones de base de datos.

DBPARTITIONNUMS *número1-partición-bd TO número2-partición-bd*

Especifica un rango de números de particiones de base de datos para que todas las particiones desde *número1-partición-bd1* hasta *número2-partición-bd* inclusive se incluyan en la lista de particiones de base de datos.

TABLESPACE *nombre-espacio-tablas*

Lista de nombres utilizados para especificar los espacios de tablas de los que se debe hacer copia de seguridad.

ONLINE

Especifica copia de seguridad en línea. Es la única modalidad soportada y es el valor por omisión. No es necesario especificar la cláusula **ONLINE**.

INCREMENTAL

Especifica una imagen de copia de seguridad acumulativa (incremental). Una imagen de copia de seguridad incremental es una copia de todos los datos de base de datos que han cambiado desde la operación de copia de seguridad completa más reciente que ha resultado satisfactoria.

DELTA

Especifica una imagen de copia de seguridad (delta) no acumulativa. Una imagen de copia de seguridad delta es una copia de todos los datos de la base de datos que se han modificado desde la operación de copia de seguridad de cualquier tipo más reciente que ha resultado satisfactoria.

USE

TSM Especifica que la copia de seguridad debe utilizar salida de Tivoli Storage Manager (TSM).

XBSA Especifica que debe utilizarse la interfaz XBSA. Las API de servicios de copia de seguridad (XBSA) se encuentran en una interfaz de programación de aplicaciones abierta para aplicaciones o recursos que necesitan gestión de almacenamiento de datos para finalidades de copia de seguridad o archivo.

SNAPSHOT

Especifica que se debe realizar una copia de seguridad instantánea.

No se puede utilizar el parámetro SNAPSHOT con ninguno de los siguientes parámetros:

- TABLESPACE
- INCREMENTAL
- WITH *núm-almacints* BUFFERS
- BUFFER
- PARALLELISM
- COMPRESS
- UTIL_IMPACT_PRIORITY
- SESSIONS

El comportamiento por omisión para una copia de seguridad instantánea es una copia de seguridad FULL DATABASE OFFLINE de todas las vías de acceso que componen la base de datos incluyendo todos los contenedores, el directorio de volúmenes local, la vía de acceso de base de datos (DBPATH) y las vías de acceso de anotaciones cronológicas primaria y de reflejos (INCLUDE LOGS es el valor por omisión de todas las copias de seguridad instantáneas a menos que se indique EXCLUDE LOGS explícitamente).

LIBRARY *nombre-biblioteca*

En IBM Data Server se integra un controlador de API ACS de DB2 para el hardware de almacenamiento siguiente:

- IBM TotalStorage SAN Volume Controller
- IBM Enterprise Storage Server Model 800
- IBM System Storage DS6000
- IBM System Storage DS8000
- IBM System Storage N Series
- NetApp V-series
- NetApp FAS

Si tiene otro hardware de almacenamiento y un controlador de API ACS de DB2 para ese hardware de almacenamiento,

puede utilizar el parámetro LIBRARY para especificar el controlador de API ACS de DB2.

El valor del parámetro LIBRARY es un nombre de archivo de biblioteca completamente calificado.

OPTIONS

"opciones-serie"

Especifica opciones que se deben utilizar para la operación de copia de seguridad. La serie se pasará al controlador de API ACS de DB2 exactamente como se ha especificado, sin las comillas dobles. No se puede utilizar el parámetro de configuración de la base de datos **VENDOROPT** para especificar opciones específicas del proveedor para operaciones de copia de seguridad instantánea. En su lugar, debe utilizar el parámetro OPTIONS de los programas de utilidad de copia de seguridad.

@nombre-archivo

Especifica que las opciones que se deben utilizar para la operación de copia de seguridad están incluidas en un archivo situado en el servidor de DB2. La serie se pasará a la biblioteca de soporte del proveedor. El archivo debe ser un nombre de archivo totalmente calificado.

OPEN *núm-sesiones* SESSIONS

El número de sesiones de E/S que se deben crear entre DB2 y TSM u otro producto del proveedor de copia de seguridad. Este parámetro no tiene ningún efecto cuando se hace la copia de seguridad en cinta, en disco o en otro dispositivo local.

TO *dir* | *disp*

Lista de nombres de directorio o dispositivo de cinta. Se debe especificar la vía de acceso completa en la que reside el directorio. Este directorio o dispositivo de destino debe existir en el servidor de bases de datos.

En una base de datos particionada, el directorio o el dispositivo de destino deben existir en todas las particiones de base de datos, y opcionalmente pueden ser una vía de acceso compartida. El nombre de directorio o de destino puede especificarse utilizando una expresión de partición de base de datos. Para obtener más información sobre las expresiones de partición de base de datos, consulte *Almacenamiento automático de bases de datos*.

Este parámetro puede repetirse para especificar los directorios y los dispositivos de destino que fragmentará la imagen de copia de seguridad. Si se especifica más de un destino (por ejemplo destino1, destino2 y destino3), se abrirá primero destino1. La cabecera del soporte y los archivos especiales (incluidos el archivo de configuración, la tabla de espacios de tablas y el archivo histórico) se colocarán en destino1. Todos los demás destinos se abren y, a continuación, se utilizan en paralelo durante la operación de copia de seguridad. Dado que no existe ningún soporte de cinta general en los sistemas operativos Windows®, cada tipo de dispositivo de cinta requiere un controlador de dispositivo exclusivo.

El uso de dispositivos de cinta o disquetes podría generar mensajes de petición y la interacción del usuario, lo que dará como resultado que se devuelva un error.

Si el sistema de la cinta no soporta la capacidad de hacer referencia exclusivamente a una imagen de copia de seguridad, se recomienda no guardar varias copias de seguridad de la misma base de datos en la misma cinta.

LOAD *nombre_biblioteca*

El nombre de la biblioteca compartida (DLL en sistemas operativos Windows) que contiene las funciones de E/S de copia de seguridad y restauración del proveedor que se debe utilizar. Puede contener la vía de acceso completa. Si no se proporciona la vía de acceso completa, tomará por omisión la vía de acceso en la que reside el programa de salida de usuario.

WITH *núm-almacins* **BUFFERS**

Número de almacenamientos intermedios a utilizar. DB2 seleccionará automáticamente un valor óptimo para este parámetro a menos que especifique explícitamente un valor. Sin embargo, cuando se crea una copia de seguridad en múltiples ubicaciones, se puede utilizar un número mayor de almacenamientos intermedios para mejorar el rendimiento.

BUFFER *tamaño-almacint*

Tamaño, en páginas de 4 KB, del almacenamiento intermedio utilizado al crear la imagen de copia de seguridad. DB2 seleccionará automáticamente un valor óptimo para este parámetro a menos que especifique explícitamente un valor. El valor mínimo para este parámetro es de 8 páginas.

Si utiliza una cinta con tamaños de bloque variables, reduzca el tamaño de almacenamiento intermedio a un rango soportado por el dispositivo de cinta. De lo contrario, la operación de copia de seguridad podría ser satisfactoria, pero podría ser que la imagen resultante no se pudiese recuperar.

En la mayoría de versiones de Linux®, la utilización del tamaño del almacenamiento intermedio por omisión de DB2 para operaciones de copia de seguridad en un dispositivo de cinta SCSI genera un error SQL2025N, código de razón 75. Para evitar el desbordamiento de almacenamientos intermedios Linux utilice esta fórmula:

$$\text{bufferpages} \leq \text{ST_MAX_BUFFERS} * \text{ST_BUFFER_BLOCKS} / 4$$

donde *bufferpages* es el valor que desea utilizar con el parámetro BUFFER y ST_MAX_BUFFERS y ST_BUFFER_BLOCKS se definen en el kernel de Linux debajo del directorio drivers/scsi.

PARALLELISM *n*

Determina el número de espacios de tablas que puede leer en paralelo el programa de utilidad de copia de seguridad. DB2 seleccionará automáticamente un valor óptimo para este parámetro a menos que especifique explícitamente un valor.

UTIL_IMPACT_PRIORITY *prioridad*

Especifica que la copia de seguridad se ejecutará en modalidad disminuida, con la prioridad especificada. La disminución le permite regular el impacto en el rendimiento de la operación de copia de seguridad. La prioridad puede ser cualquier número entre 1 y 100, con 1 representando la prioridad más baja, y 100 representando la prioridad más alta. Si se especifica la palabra clave UTIL_IMPACT_PRIORITY sin ninguna prioridad, la copia de seguridad se ejecutará con la prioridad por omisión de 50. Si no se ha especificado UTIL_IMPACT_PRIORITY, la copia de

seguridad se ejecutará en modalidad no disminuida. Se debe definir una política de impacto estableciendo el parámetro de configuración *util_impact_lim* para que una copia de seguridad se ejecute en modalidad disminuida.

COMPRESS

Indica que la copia de seguridad se debe comprimir.

COMPRLIB *nombre*

Indica el nombre de la biblioteca que se debe utilizar para realizar la compresión (por ejemplo, *db2compr.dll* para Windows; *libdb2compr.so* para sistemas Linux/UNIX). El nombre debe ser una vía de acceso totalmente calificada que haga referencia a un archivo del servidor. Si no se especifica este parámetro, se utilizará la biblioteca de compresión de DB2 por omisión. Si la biblioteca especificada no se puede cargar, la copia de seguridad fallará.

EXCLUDE

Indica que la biblioteca de compresión no se almacenará en la imagen de copia de seguridad.

COMPROPTS *serie*

Describe un bloque de datos binarios que se pasará a la rutina de inicialización en la biblioteca de compresión. DB2 pasará esta serie directamente del cliente al servidor, de modo que los posibles problemas de inversión de bytes o de conversión de páginas de códigos los deberá manejar la biblioteca de compresión. Si el primer carácter del bloque de datos es '@', DB2 interpretará los datos restantes como el nombre de un archivo que se encuentra en el servidor. A continuación, DB2 sustituirá el contenido de la serie por el contenido de este archivo y pasará en su lugar este nuevo valor a la rutina de inicialización. La longitud máxima de *serie* es de 1024 bytes.

EXCLUDE LOGS

Especifica que la imagen de copia de seguridad no debe incluir ningún archivo de anotaciones cronológicas. Al realizar una operación de copia de seguridad fuera de línea, las anotaciones cronológicas se excluyen tanto si se especifica esta opción como si no, con la excepción de las copias de seguridad instantáneas.

INCLUDE LOGS

Especifica que la imagen de copia de seguridad debe incluir el rango de archivos de anotaciones cronológicas necesarios para restaurar y avanzar esta imagen hasta un punto en el tiempo coherente. Esta opción no es válida para una copia de seguridad fuera de línea, con la excepción de copias de seguridad instantáneas donde esta opción es el valor por omisión a menos que se haya indicado explícitamente que se excluya.

WITHOUT PROMPTING

Especifica que la copia de seguridad se ejecutará desatendida y que las acciones que normalmente requieren la intervención del usuario devolverán un mensaje de error. Es el valor por omisión.

Ejemplos

A continuación, se muestra una estrategia de copia de seguridad incremental semanal para una base de datos recuperable. Incluye una operación de copia de seguridad de base de datos completa semanal, una operación de copia de

seguridad no acumulativa (delta) diaria y una operación de copia de seguridad acumulativa (incremental) de mitad de semana:

```
(Sun) CALL SYSPROC.ADMIN_CMD('backup db sample online use tsm')
(Mon) CALL SYSPROC.ADMIN_CMD
      ('backup db sample online incremental delta use tsm')
(Tue) CALL SYSPROC.ADMIN_CMD
      ('backup db sample online incremental delta use tsm')
(Wed) CALL SYSPROC.ADMIN_CMD
      ('backup db sample online incremental use tsm')
(Thu) CALL SYSPROC.ADMIN_CMD
      ('backup db sample online incremental delta use tsm')
(Fri) CALL SYSPROC.ADMIN_CMD
      ('backup db sample online incremental delta use tsm')
(Sat) CALL SYSPROC.ADMIN_CMD
      ('backup db sample online incremental use tsm')
```

Notas de uso

Los datos de una copia de seguridad no pueden ir protegidos por el servidor de bases de datos. Asegúrese de que las copias de seguridad estén protegidas correctamente, en especial, si la copia de seguridad contiene datos protegidos para LBAC.

Al hacer copia de seguridad en cinta, el uso del tamaño de bloque variable no está soportado actualmente. Si debe usar esta opción, asegúrese de que tiene implantados procedimientos bien probados que le permitan hacer una recuperación satisfactoria, utilizando imágenes de copia de seguridad que se crearon con un tamaño de bloque variable.

Al utilizar un tamaño de bloque variable, debe especificar un tamaño de almacenamiento intermedio de copia de seguridad que sea menor o igual que el límite máximo de los dispositivos de cinta que esté utilizando. Para optimizar el rendimiento, el tamaño del almacenamiento intermedio debe ser igual al límite de tamaño de bloque máximo del dispositivo que se utilice.

Las copias de seguridad instantáneas se deben complementar con copias de seguridad de disco regulares en caso de anomalía en el sistema del archivador/almacenamiento.

Puesto que deberá realizar una copia de seguridad de la base de datos regularmente, es posible que acumule imágenes de copia de seguridad de base de datos de gran tamaño, numerosas anotaciones cronológicas de base de datos e imágenes de copia de carga que pueden ocupar una gran cantidad de espacio en disco. Consulte la tarea “Gestión de objetos de recuperación” para obtener información sobre cómo gestionar estos objetos de recuperación.

Información de conjunto de resultados

El estado de la ejecución del mandato se devuelve en el área de comunicaciones SQL (SQLCA) obtenida a partir de la sentencia CALL. Si la ejecución es satisfactoria, el mandato devuelve información adicional. La operación de copia de seguridad devolverá un conjunto de resultados, que comprende una fila por partición de base de datos que ha participado en la copia de seguridad.

Tabla 29. Conjunto de resultados para una operación de copia de seguridad

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| BACKUP_TIME | VARCHAR(14) | Corresponde a la serie de indicación de fecha y hora que se utiliza para nombrar la imagen de copia de seguridad. |
| DBPARTITIONNUM | SMALLINT | El número de la partición de base de datos en el que el agente ha ejecutado la operación de copia de seguridad. |
| SQLCODE | INTEGER | SQLCODE final como resultado del proceso de copia de seguridad en la partición de la base de datos especificada. |
| SQLERRMC | VARCHAR(70) | SQLERRMC final como resultado del proceso de copia de seguridad en la partición de base de datos especificada. |
| SQLERRML | SMALLINT | SQLERRML final como resultado del proceso de copia de seguridad en la partición de base de datos especificada. |

Si se realiza una copia de seguridad de una base de datos no particionada, o si se realiza una copia de seguridad de una base de datos particionada utilizando la sintaxis de una única partición tradicional, el conjunto de resultados comprenderá una sola fila. DBPARTITIONNUM contendrá el número de identificador de la partición de la base de datos de la que se está realizando una copia de seguridad.

SQLCODE, SQLERRMC y SQLERRML hacen referencia a los miembros de nombre equivalente del SQLCA que la copia de seguridad devuelve en la partición de la base de datos especificada.

Mandato DESCRIBE utilizando el procedimiento ADMIN_CMD

Utilice el mandato DESCRIBE para visualizar información sobre cualquiera de los elementos siguientes:

- Salida de una sentencia SELECT, CALL o XQuery
- Columnas de una tabla o de una vista
- Índices de una tabla o una vista
- Particiones de datos de una tabla o vista

Autorización

La autorización necesaria depende del tipo de información que desea mostrar al utilizar el mandato DESCRIBE.

- Si existe el espacio de tablas SYSTOOLSTMPSPACE, es necesaria una de las autorizaciones que se muestran en la siguiente tabla.

| Objeto sobre el que se muestra información | Privilegios o autorizaciones necesarias |
|---|---|
| Salida de una sentencia SELECT o sentencia XQuery | <p>Cualquiera de los privilegios o de las autorizaciones siguientes para cada tabla o vista a la que se hace referencia en la sentencia SELECT:</p> <ul style="list-style-type: none"> • Privilegio SELECT • Autorización DATAACCESS • Autorización DBADM • Autorización SQLADM • Autorización EXPLAIN |
| Salida de una sentencia CALL | <p>Cualquiera de los privilegios o de las autorizaciones siguientes:</p> <ul style="list-style-type: none"> • Autorización DATAACCESS • Privilegio EXECUTE para el procedimiento almacenado |
| Columnas de una tabla o de una vista | <p>Cualquiera de los siguientes privilegios o autorizaciones de la tabla de catálogo del sistema SYSCAT.COLUMNS:</p> <ul style="list-style-type: none"> • Privilegio SELECT • Autorización ACCESSCTRL • Autorización DATAACCESS • Autorización DBADM • Autorización SECADM • Autorización SQLADM <p>Si desea utilizar el parámetro SHOW DETAIL, también puede requerir cualquiera de estos privilegios o autorizaciones en la tabla de catálogo SYSCAT.DATAPARTITIONEXPRESSION.</p> <p>Puesto que PUBLIC tiene todos los privilegios sobre las tablas temporales declaradas, puede utilizar el mandato para mostrar información sobre cualquier tabla temporal declarada que exista en la conexión.</p> |

| Objeto sobre el que se muestra información | Privilegios o autorizaciones necesarias |
|--|--|
| Índices de una tabla o una vista | <p>Cualquiera de los siguientes privilegios o autorizaciones de la tabla de catálogo del sistema SYSCAT.INDEXES:</p> <ul style="list-style-type: none"> • Privilegio SELECT • Autorización ACCESSCTRL • Autorización DATAACCESS • Autorización DBADM • Autorización SECADM • Autorización SQLADM <p>Si desea utilizar el parámetro SHOW DETAIL, también requiere privilegio EXECUTE en GET_INDEX_COLNAMES() UDF.</p> <p>Puesto que PUBLIC tiene todos los privilegios sobre las tablas temporales declaradas, puede utilizar el mandato para mostrar información sobre cualquier tabla temporal declarada que exista en la conexión.</p> |
| Particiones de datos de una tabla o vista | <p>Cualquiera de los siguientes privilegios o autorizaciones de la tabla de catálogo del sistema SYSCAT.DATAPARTITIONS:</p> <ul style="list-style-type: none"> • Privilegio SELECT • Autorización ACCESSCTRL • Autorización DATAACCESS • Autorización DBADM • Autorización SECADM • Autorización SQLADM <p>Puesto que PUBLIC tiene todos los privilegios sobre las tablas temporales declaradas, puede utilizar el mandato para mostrar información sobre cualquier tabla temporal declarada que exista en la conexión.</p> |

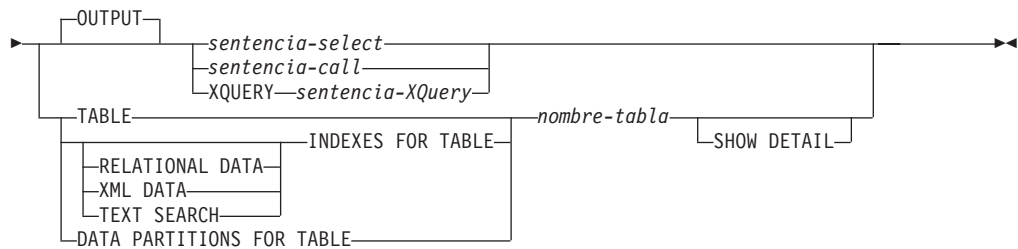
- Si el espacio de tablas SYSTOOLSTMPSPACE no existe, se necesitará la autorización SYSADM o SYSCTRL además de una de las autorizaciones anteriores.

Conexión necesaria

Base de datos

Sintaxis del mandato

►►—DESCRIBE—►



Parámetros del mandato

OUTPUT

Indica que debe describirse la salida de la sentencia. Esta palabra clave es opcional.

sentencia-select | *sentencia-call* | **XQUERY** *sentencia-XQuery*

Identifica la sentencia sobre la que se desea información. CLP prepara automáticamente la sentencia. Para identificar una sentencia XQuery, hay que poner antes la palabra clave XQUERY. Una sentencia DESCRIBE OUTPUT sólo devuelve información sobre una columna oculta implícita si la columna se especifica de forma explícita como parte de la lista SELECT de la tabla de resultados final de la consulta descrita.

TABLE *nombre-tabla*

Especifica la tabla o vista que se debe describir. Se debe utilizar el nombre totalmente calificado con el formato *esquema.nombre-tabla*. No se puede utilizar un alias para la tabla en lugar de la tabla misma. Se devuelve información sobre columnas ocultas de forma implícita.

El mandato DESCRIBE TABLE lista la siguiente información sobre cada columna:

- Nombre de columna
- Esquema de tipo
- Nombre de tipo
- Longitud
- Escala
- Nulos (sí/no)

INDEXES FOR TABLE *nombre-tabla*

Especifica la tabla o la vista para la que es necesario describir índices. Puede utilizar el nombre totalmente calificado con el formato *esquema.nombre-tabla* o simplemente puede especificar el *nombre-tabla* y se utilizará automáticamente el esquema por omisión. No se puede utilizar un alias para la tabla en lugar de la tabla misma.

El mandato DESCRIBE INDEXES FOR TABLE lista la siguiente información acerca de cada índice de la tabla o vista:

- Esquema de índice
- Nombre de índice
- Norma exclusiva
- Número de columnas
- Tipo de índice

Si el mandato DESCRIBE INDEXES FOR TABLE se ha especificado con la opción SHOW DETAIL, el nombre de índice se trunca cuando sobrepasa

los 18 bytes. Si no se ha especificado ninguna opción de tipo de índice, aparece la información correspondiente a todos los tipos de índice: índice de datos relacionales, índice sobre datos XML e índice de Text Search. La salida incluye la siguiente información adicional:

- ID de índice para un índice de datos relacionales, un índice de vía de acceso XML, un índice de regiones de XML o un índice sobre datos XML
- Tipo de datos para un índice sobre datos XML
- Generado aleatoriamente (hashed) para un índice sobre datos XML
- Longitud máxima de VARCHAR para un índice sobre datos XML
- Patrón XML especificado para un índice sobre datos XML
- Página de códigos para un índice de búsqueda de texto
- Idioma para un índice de búsqueda de texto
- Formato especificado para un índice de búsqueda de texto
- Actualización mínima para un índice de búsqueda de texto
- Frecuencia de actualización para un índice de búsqueda de texto
- Directorio de la colección para un índice de búsqueda de texto
- Nombres de columna

Especifique un tipo de índice para que se muestre información sólo de un tipo de índice específico. No se admite la especificación de varios tipos de índice.

RELATIONAL DATA

Si la opción de tipo de índice RELATIONAL DATA se especifica sin la opción SHOW DETAIL, sólo aparece la información siguiente:

- Esquema de índice
- Nombre de índice
- Norma exclusiva
- Número de columnas

Si se especifica SHOW DETAIL, también aparece la información de nombres de columnas.

XML DATA

Si la opción de tipo de índice XML DATA se especifica sin la opción SHOW DETAIL, sólo aparece la información siguiente:

- Esquema de índice
- Nombre de índice
- Norma exclusiva
- Número de columnas
- Tipo de índice

Si se especifica SHOW DETAIL, la información siguiente de un índice sobre datos XML también aparece en la lista:

- ID de índice
- Tipo de datos
- Generado aleatoriamente (hashed)
- Longitud máxima de Varchar
- Patrón XML
- Nombres de columna

TEXT SEARCH

Si la opción de tipo de índice TEXT SEARCH se especifica sin la opción SHOW DETAIL, sólo aparece la información siguiente:

- Esquema de índice
- Nombre de índice

Si se especifica SHOW DETAIL, también aparece la información de índice de búsqueda de texto siguiente:

- Nombre de columna
- Página de códigos
- Lenguaje
- Formato
- Actualización mínima
- Frecuencia de actualización
- Directorio de la colección

Si se ha especificado la opción TEXT SEARCH y no hay una opción de búsqueda de texto instalada o no está bien configurada, se devuelve un error (SQLSTATE 42724).

Consulte DB2 Text Search para obtener la información que aparece en las columnas.

DATA PARTITIONS FOR TABLE *nombre-tabla*

Especifica la tabla o la vista para la que es necesario describir particiones de datos. La información visualizada para cada partición de datos en la tabla es el identificador de la partición y los intervalos de particionamiento. Los resultados se ordenan según la secuencia de identificadores de las particiones. Se debe utilizar el nombre totalmente calificado con el formato *esquema.nombre-tabla*. No se puede utilizar un alias para la tabla en lugar de la tabla misma. El *esquema* es el nombre de usuario bajo el que se ha creado la tabla o la vista.

En el caso del mandato DESCRIBE DATA PARTITIONS FOR TABLE, especifica que la salida incluya una segunda tabla con la siguiente incluir adicional:

- Identificador de secuencia particiones de datos
- Expresión de particiones de datos en SQL

SHOW DETAIL

En el caso del mandato DESCRIBE TABLE, especifica que en la salida se incluya la siguiente información adicional así como un segundo conjunto de resultados que contiene las expresiones de particiones de datos de tabla (puede devolver 0 filas si la tabla no tiene particiones de datos):

- Si se ha definido una columna CHARACTER, VARCHAR o LONG VARCHAR como FOR BIT DATA
- Número de columna
- Secuencia de claves de distribución
- Página de códigos
- Valor por omisión
- Tipo de particionamiento de tabla (en el caso de tablas particionadas por rango, esta salida aparece más abajo de la salida original)

- Columnas de clave de particionamiento (en el caso de tablas particionadas por rango, esta salida aparece más abajo de la salida original)
- Identificador del espacio de tablas utilizado para el índice

Ejemplos

Descripción de la salida de una sentencia SELECT

El ejemplo siguiente muestra cómo describir una sentencia SELECT:

```
CALL SYSPROC.ADMIN_CMD('describe select * from emp_photo')
```

El siguiente ejemplo muestra una salida de esta sentencia SELECT.

Conjunto de resultados 1

| SQLTYPE_ID | SQLTYPE | SQLLENGTH | SQLSCALE | SQLNAME_DATA | ... |
|------------|-----------|-----------|----------|--------------|-----|
| 452 | CHARACTER | 6 | 0 | EMPNO | ... |
| 448 | VARCHAR | 10 | 0 | PHOTO_FORMAT | ... |
| 405 | BLOB | 102400 | 0 | PICTURE | ... |

3 registro(s) seleccionado(s).
Estado de devolución = 0

Salida de esta sentencia SELECT (continuación).

| ... | SQLNAME_LENGTH | SQLDATATYPENAME_DATA | SQLDATATYPENAME_LENGTH |
|-----|----------------|----------------------|------------------------|
| ... | 5 | SYSIBM .CHARACTER | 18 |
| ... | 12 | SYSIBM .VARCHAR | 16 |
| ... | 7 | SYSIBM .BLOB | 13 |

Descripción de una tabla

Describir una tabla no particionada.

```
CALL SYSPROC.ADMIN_CMD('describe table org show detail')
```

El siguiente ejemplo muestra una salida de esta sentencia CALL.

Conjunto de resultados 1

| COLNAME | TYPESHEMA | TYPENAME | FOR_BINARY_DATA | ... |
|----------|-----------|----------|-----------------|-----|
| DEPTNUMB | SYSIBM | SMALLINT | N | ... |
| DEPTNAME | SYSIBM | VARCHAR | N | ... |
| MANAGER | SYSIBM | SMALLINT | N | ... |
| DIVISION | SYSIBM | VARCHAR | N | ... |
| LOCATION | SYSIBM | VARCHAR | N | ... |

5 registro(s) seleccionado(s).

Salida de esta sentencia CALL (continuación).

| ... | LENGTH | SCALE | NULLABLE | COLNO | PARTKEYSEQ | CODEPAGE | DEFAULT |
|-----|--------|-------|----------|-------|------------|----------|---------|
| ... | 2 | 0 | N | 0 | 1 | 0 | - |
| ... | 14 | 0 | Y | 1 | 0 | 1208 | - |
| ... | 2 | 0 | Y | 2 | 0 | 0 | - |
| ... | 10 | 0 | Y | 3 | 0 | 1208 | - |
| ... | 13 | 0 | Y | 4 | 0 | 1208 | - |

Salida de esta sentencia CALL (continuación).

Conjunto de resultados 2

```
-----  
DATA_PARTITION_KEY_SEQ DATA_PARTITION_EXPRESSION  
-----
```

0 registro(s) seleccionado(s).
Estado de devolución = 0

Describir una tabla particionada.

```
CALL SYSPROC.ADMIN_CMD('describe table part_table1 show detail')
```

El siguiente ejemplo muestra una salida de esta sentencia CALL.

Conjunto de resultados 1

```
-----  
COLNAME      TYPESCHEMA      TYPENAME FOR_BINARY_DATA ...  
-----...- - - - -  
COL1         SYSIBM         INTEGER N          ...
```

1 registro(s) seleccionado(s).

Salida de esta sentencia CALL (continuación).

```
... LENGTH SCALE NULLABLE COLNO PARTKEYSEQ CODEPAGE DEFAULT  
... -----  
...      4      0 N          0          1          0      -
```

Salida de esta sentencia CALL (continuación).

Conjunto de resultados 2

```
-----  
DATA_PARTITION_KEY_SEQ DATA_PARTITION_EXPRESSION  
-----  
1 COL1
```

1 registro(s) seleccionado(s)

Descripción de un índice de tabla

En el siguiente ejemplo se muestra cómo describir un índice de tabla. Esta llamada enumera dos índices de datos relacionales, seis índices de datos XML, dos índices de búsqueda de texto y los índices del sistema:

```
CALL SYSPROC.ADMIN_CMD('describe indexes for table user1.department')
```

El siguiente ejemplo muestra una salida de esta sentencia CALL.

Conjunto de resultados 1

```
-----  
INDSCHEMA      INDNAME          UNIQUE_RULE  
-----...- - - - -  
SYSIBM         SQL070531145253450  DUPLICATES_ALLOWED  
SYSIBM         SQL070531145253620  UNIQUE_ENTRIES_ONLY  
USER1          RELIDX1            DUPLICATES_ALLOWED  
USER1          RELIDX2            DUPLICATES_ALLOWED  
SYSIBM         SQL070531145253650  PRIMARY_INDEX  
USER1          XMLIDX1            DUPLICATES_ALLOWED  
SYSIBM         SQL070531154625650  DUPLICATES_ALLOWED  
USER1          XMLIDX2            DUPLICATES_ALLOWED  
SYSIBM         SQL070531154626000  DUPLICATES_ALLOWED  
USER1          XMLIDX3            DUPLICATES_ALLOWED  
SYSIBM         SQL070531154626090  DUPLICATES_ALLOWED  
USER1          XMLIDX4            DUPLICATES_ALLOWED  
SYSIBM         SQL070531154626190  DUPLICATES_ALLOWED  
USER1          XMLIDX5            DUPLICATES_ALLOWED  
SYSIBM         SQL070531154626290  DUPLICATES_ALLOWED
```

```

USER1          XMLIDX6          DUPLICATES_ALLOWED
SYSIBM        SQL070531154626400  DUPLICATES_ALLOWED
USER1          TXTIDX1           -
USER1          TXTIDX2           -

```

19 registros(s) seleccionado(s).

Estado de devolución = 0

Salida de esta sentencia CALL (continuación).

```

... COLCOUNT  INDEXTYPE
... -----  -
...          -  XML_DATA_REGIONS
...          1  XML_DATA_PATH
...          1  RELATIONAL_DATA
...          2  RELATIONAL_DATA
...          1  RELATIONAL_DATA
...          1  XML_DATA_VALUES_LOGICAL
...          1  XML_DATA_VALUES_PHYSICAL
...          1  XML_DATA_VALUES_LOGICAL
...          1  XML_DATA_VALUES_PHYSICAL
...          1  XML_DATA_VALUES_LOGICAL
...          1  XML_DATA_VALUES_PHYSICAL
...          1  XML_DATA_VALUES_LOGICAL
...          1  XML_DATA_VALUES_PHYSICAL
...          1  XML_DATA_VALUES_LOGICAL
...          1  XML_DATA_VALUES_PHYSICAL
...          1  XML_DATA_VALUES_LOGICAL
...          1  XML_DATA_VALUES_PHYSICAL
...          1  XML_DATA_VALUES_LOGICAL
...          1  XML_DATA_VALUES_PHYSICAL
...          1  XML_DATA_VALUES_LOGICAL
...          1  XML_DATA_VALUES_PHYSICAL
...          1  XML_DATA_VALUES_LOGICAL
...          1  XML_DATA_VALUES_PHYSICAL
...          1  XML_DATA_VALUES_LOGICAL
...          1  XML_DATA_VALUES_PHYSICAL
...          1  XML_DATA_VALUES_LOGICAL
...          1  XML_DATA_VALUES_PHYSICAL
...          1  TEXT_SEARCH
...          1  TEXT_SEARCH

```

Descripción de una partición de datos

En el siguiente ejemplo se muestra cómo describir particiones de datos.

```
CALL SYSPROC.ADMIN_CMD('describe data partitions for table part_table2')
```

El siguiente ejemplo muestra una salida de esta sentencia CALL.

Conjunto de resultados 1

```

-----
DATA_PARTITION_ID LOW_KEY_INCLUSIVE LOW_KEY_VALUE ...
-----
                0 Y                1                ...
                1 Y                10               ...
                2 Y                20               ...

```

3 registro(s) seleccionado(s).

Salida de esta sentencia CALL (continuación).

```

... HIGH_KEY_INCLUSIVE HIGH_KEY_VALUE
... -----
... N                    10
... N                    20
... N                    40

```

En el siguiente ejemplo se muestra cómo describir particiones de datos con la cláusula 'SHOW DETAIL'.

```
CALL SYSPROC.ADMIN_CMD('describe data partitions for table part_table2 show detail')
```

El siguiente ejemplo muestra una salida de esta sentencia CALL.

Conjunto de resultados 1

```
-----  
DATA_PARTITION_ID LOW_KEY_INCLUSIVE LOW_KEY_VALUE ...  
-----  
0 Y 1 ...  
1 Y 10 ...  
2 Y 20 ...
```

3 registro(s) seleccionado(s).

Estado de devolución = 0

Salida de esta sentencia CALL (continuación).

```
... HIGH_KEY_INCLUSIVE HIGH_KEY_VALUE  
... -----  
... N 10  
... N 20  
... N 40
```

Salida de esta sentencia CALL (continuación).

Conjunto de resultados 2

```
-----  
DATA_PARTITION_ID DATA_PARTITION_NAME TBSPID ...  
-----  
0 PART0 3 ...  
1 PART1 3 ...  
2 PART2 3 ...
```

3 registro(s) seleccionado(s).

Estado de devolución = 0

Salida de esta sentencia CALL (continuación).

```
... PARTITION_OBJECT_ID LONG_TBSPID ACCESSMODE STATUS  
... -----  
... 15 3 FULL_ACCESS  
... 16 3 FULL_ACCESS  
... 17 3 FULL_ACCESS
```

Nota sobre uso

Si el mandato DESCRIBE intenta crear una tabla temporal y no lo consigue, se intenta crear SYSTOOLSTMPSPACE y a continuación, se intenta crear de nuevo la tabla temporal, esta vez en SYSTOOLSTMPSPACE. Es necesaria autorización SYSCTRL o SYSADM para crear el espacio de tablas SYSTOOLSTMPSPACE.

Información de conjunto de resultados

El estado de la ejecución del mandato se devuelve en el área de comunicaciones SQL (SQLCA) obtenida a partir de la sentencia CALL. Si la ejecución es satisfactoria, los mandatos devuelven información adicional en conjuntos de resultados, de la siguiente manera:

- Tabla 30 en la página 65: mandatos DESCRIBE *sentencia-select*, DESCRIBE *sentencia-call* y DESCRIBE XQUERY *sentencia-XQuery*
- Tabla 31 en la página 65: Conjunto de resultados 1 del mandato DESCRIBE TABLE
- Tabla 32 en la página 66: Conjunto de resultados 2 del mandato DESCRIBE TABLE

- Tabla 33 en la página 66: Mandato DESCRIBE INDEXES FOR TABLE
- Tabla 34 en la página 68: Conjunto de resultados 1 del mandato DESCRIBE DATA PARTITIONS FOR TABLE
- Tabla 35 en la página 69: Conjunto de resultados 2 del mandato DESCRIBE DATA PARTITIONS FOR TABLE

Tabla 30. Conjunto de resultados devuelto por los mandatos DESCRIBE sentencia-select, DESCRIBE sentencia-call y DESCRIBE XQUERY sentencia-XQuery

| Nombre de columna | Tipo de datos | Sólo LOB ¹ | Descripción |
|-------------------|---------------|-----------------------|---|
| SQLTYPE_ID | SMALLINT | No | Tipo de datos de la columna, tal como aparece en el campo SQLTYPE del área de descriptores SQL (SQLDA). |
| SQLTYPE | VARCHAR (257) | No | Tipo de datos correspondiente al valor SQLTYPE_ID. |
| SQLEN | INTEGER | No | Atributo longitud de la columna, tal como aparece en el campo SQLEN del SQLDA. |
| SQLSCALE | SMALLINT | No | Número de dígitos de la parte fraccionaria de un valor decimal; 0 en el caso de otros tipos de datos. |
| SQLNAME_DATA | VARCHAR (128) | No | Nombre de la columna. |
| SQLNAME_LENGTH | SMALLINT | No | Longitud del nombre de la columna. |
| SQLDATA_TYPSHEMA | VARCHAR (128) | Sí | Nombre del esquema de tipo de datos. |
| SQLDATA_TYPENAME | VARCHAR (128) | Sí | Nombre del tipo de datos. |

Nota: ¹: el valor Sí (Yes) indica que sólo se devuelven valores no-nulos cuando se describen datos LOB.

Tabla 31. Conjunto de resultados 1 devuelto por el mandato DESCRIBE TABLE

| Nombre de columna | Tipo de datos | Detalle ² | Descripción |
|-------------------|---------------|----------------------|--|
| COLNAME | VARCHAR (128) | No | Nombre de la columna. |
| TYPESHEMA | VARCHAR (128) | No | Si el nombre de la columna es distinto, se devuelve el nombre del esquema; en caso contrario, se devuelve 'SYSIBM'. |
| TYPENAME | VARCHAR (128) | No | Nombre del tipo de columna. |
| FOR_BINARY_DATA | CHAR (1) | Sí | Devuelve 'Y' si la columna es de tipo CHAR, VARCHAR o LONG VARCHAR y está definida como FOR BIT DATA; de lo contrario, devuelve 'N'. |
| LENGTH | INTEGER | No | Longitud máxima de los datos. En el caso de datos DECIMAL, esto indica la precisión. En el caso de tipos distintos, se devuelve 0. |
| SCALE | SMALLINT | No | En el caso de datos DECIMAL, esto indica la escala. Par todos los demás tipos, se devuelve 0. |

Tabla 31. Conjunto de resultados 1 devuelto por el mandato DESCRIBE TABLE (continuación)

| Nombre de columna | Tipo de datos | Detalle ² | Descripción |
|-------------------|---------------|----------------------|--|
| NULLABLE | CHAR (1) | No | Uno de los siguientes: <ul style="list-style-type: none"> • 'Y' si la columna tiene posibilidad de nulos • 'N' si la columna no tiene posibilidad de nulos |
| COLNO | SMALLINT | Sí | Ordinal de la columna. |
| PARTKEYSEQ | SMALLINT | Sí | Ordinal de la columna dentro de la clave de particionamiento de la tabla. Se devuelve NULL o 0 si la columna no forma parte de la clave de particionamiento; es NULL para las subtablas y las tablas de la jerarquía. |
| CODEPAGE | SMALLINT | Sí | Página de códigos de la columna y uno de estos valores: <ul style="list-style-type: none"> • Valor de la página de códigos de base de datos de columnas no definidas con FOR BIT DATA. • Valor de la página de códigos DBCS para columnas gráficas. • De lo contrario, 0. |
| DEFAULT | VARCHAR (254) | Sí | Valor por omisión de la columna de una tabla expresada como constante, registro especial o función de conversión apropiada para el tipo de datos de la columna. También podría ser NULL. |

Nota: ²: el valor Sí (Yes) indica que sólo se devuelven valores no-nulos cuando se utiliza la cláusula SHOW DETAIL.

Tabla 32. Conjunto de resultados 2 devuelto por el mandato DESCRIBE TABLE cuando se utiliza la cláusula SHOW DETAIL.

| Nombre de columna | Tipo de datos | Descripción |
|---------------------------|---------------|--|
| DATA_PARTITION_KEY_SEQ | INTEGER | Número de clave de partición de datos, por ejemplo, 1 para la primera expresión de partición de datos y 2 para la segunda expresión de partición de datos. |
| DATA_PARTITION_EXPRESSION | CLOB (32K) | Expresión para esta clave de partición de datos en sintaxis SQL |

Tabla 33. Conjunto de resultados devuelto por el mandato DESCRIBE INDEXES FOR TABLE

| Nombre de columna | Tipo de datos | Detalles ³ | Opción de tipo de índice ^{4, 5} | Descripción |
|-------------------|---------------|-----------------------|--|-------------------------------|
| INDSCHEMA | VARCHAR (128) | No | RELATIONAL DATA XML DATA TEXT SEARCH | Nombre del esquema de índice. |
| INDNAME | VARCHAR (128) | No | RELATIONAL DATA XML DATA TEXT SEARCH | Nombre del índice. |

Tabla 33. Conjunto de resultados devuelto por el mandato DESCRIBE INDEXES FOR TABLE (continuación)

| Nombre de columna | Tipo de datos | Detalles ³ | Opción de tipo de índice ^{4, 5} | Descripción |
|--------------------|---------------|-----------------------|--|--|
| UNIQUE_RULE | VARCHAR (30) | No | RELATIONAL DATA XML DATA | Uno de los valores siguientes: <ul style="list-style-type: none"> • DUPLICATES_ALLOWED • PRIMARY_INDEX • UNIQUE_ENTRIES_ONLY |
| INDEX_PARTITIONING | CHAR(1) | No | N/D | Identifica la característica de particionamiento del índice. Los valores posibles son: <ul style="list-style-type: none"> • N= índice no particionado • P= índice particionado • Blanco = el índice no está en una tabla particionada |
| COLCOUNT | SMALLINT | No | RELATIONAL DATA XML DATA | Número de columnas de la clave, más el número de columnas de inclusión, si hay alguna. |
| INDEX_TYPE | VARCHAR (30) | No | RELATIONAL DATA XML DATA TEXT SEARCH | Tipo de índice: <ul style="list-style-type: none"> • RELATIONAL_DATA • TEXT_SEARCH • XML_DATA_REGIONS • XML_DATA_PATH • XML_DATA_VALUES_LOGICAL • XML_DATA_VALUES_PHYSICAL |
| INDEX_ID | SMALLINT | Sí | RELATIONAL DATA XML DATA | ID de índice para un índice de datos relacionales, un índice de vía de acceso XML, un índice de regiones de XML o un índice sobre datos XML |
| DATA_TYPE | VARCHAR (128) | Sí | XML DATA | Tipo de datos SQL especificado para un índice sobre datos XML. Uno de los valores siguientes: <ul style="list-style-type: none"> • VARCHAR • DOUBLE • DATE • TIMESTAMP |
| HASHED | CHAR (1) | Sí | XML DATA | Indica si el valor de un índice sobre datos XML se ha generado de forma aleatoria. <ul style="list-style-type: none"> • 'Y' si el valor se ha generado de forma aleatoria (hashed). • 'N' si el valor no se ha generado de forma aleatoria (hashed). |
| LENGTH | SMALLINT | Sí | XML DATA | Para un índice sobre datos XML, la longitud de VARCHAR (entero); de lo contrario, 0. |
| PATTERN | CLOB(2M) | Sí | XML DATA | Patrón XML especificado para un índice sobre datos XML |
| CODEPAGE | INTEGER | Sí | TEXT SEARCH | Página de códigos del documento especificada para el índice de búsqueda de texto |

Tabla 33. Conjunto de resultados devuelto por el mandato DESCRIBE INDEXES FOR TABLE (continuación)

| Nombre de columna | Tipo de datos | Detalles ³ | Opción de tipo de índice ^{4, 5} | Descripción |
|----------------------|----------------|-----------------------|--|---|
| LANGUAGE | VARCHAR(5) | Sí | TEXT SEARCH | Idioma del documento especificado para el índice de búsqueda de texto |
| FORMAT | VARCHAR (30) | Sí | TEXT SEARCH | Formato del documento especificado para un índice de búsqueda de texto |
| UPDATEMINIMUM | INTEGER | Sí | TEXT SEARCH | Número mínimo de entradas en la tabla de anotaciones cronológicas de búsqueda de texto antes de llevar a cabo la actualización incremental |
| UPDATEFREQUENCY | VARCHAR(300) | Sí | TEXT SEARCH | Criterio de activador especificado para aplicar actualizaciones al índice de texto |
| COLLECTION DIRECTORY | VARCHAR (512) | Sí | TEXT SEARCH | Directorio especificado para los archivos de índice de búsqueda de texto |
| COLNAMES | VARCHAR (2048) | Sí | RELATIONAL DATA XML DATA TEXT SEARCH | Lista de los nombres de columnas, cada uno de ellos precedido con un signo + para indicar el orden ascendente o con un signo - para indicar el orden descendente. |

Nota: ³: Sí (Yes) indica que los valores sólo se devuelven cuando la cláusula SHOW DETAIL se utiliza sin especificar una opción de tipo de índice. Los valores pueden ser NULL.

Nota: ⁴: indica los valores devueltos cuando se utiliza DESCRIBE *tipo-índice* INDEXES FOR TABLE. Por ejemplo, los valores de INDEX_ID no se devuelven si se especifica TEXT SEARCH como *tipo-índice*. Los valores de INDEX_ID se devuelven si se especifica RELATIONAL DATA o bien XML DATA.

Nota: ⁵: cuando se utiliza DESCRIBE *tipo-índice* INDEXES FOR TABLE SHOW DETAIL, los valores sólo se devuelven cuando se enumera el tipo de índice. Por ejemplo, los valores de DATA_TYPE se devuelven si se especifica XML DATA *tipo-índice*. Los valores de DATA_TYPE no se devuelven si se especifica TEXT SEARCH o bien RELATIONAL DATA como *tipo-índice*.

Tabla 34. Conjunto de resultados 1 devuelto por el mandato DESCRIBE DATA PARTITIONS FOR TABLE

| Nombre de columna | Tipo de datos | Detalle ² | Descripción |
|--------------------|---------------|----------------------|---|
| DATA_PARTITION_ID | INTEGER | No | Identificador de partición de datos |
| LOW_KEY_INCLUSIVE | CHAR (1) | No | 'Y' si el valor bajo de clave es inclusive; de lo contrario, 'N'. |
| LOW_KEY_VALUE | VARCHAR (512) | No | Valor bajo de clave para esta partición de datos. |
| HIGH_KEY_INCLUSIVE | CHAR (1) | No | 'Y' si el valor alto de clave es inclusive; de lo contrario, 'N'. |
| HIGH_KEY_VALUE | VARCHAR (512) | No | Valor alto de clave para esta partición de datos. |

Nota: ²: el valor Sí (Yes) indica que sólo se devuelven valores no-nulos cuando se utiliza la cláusula SHOW DETAIL.

Tabla 35. Conjunto de resultados 2 devuelto por el mandato DESCRIBE DATA PARTITIONS FOR TABLE cuando se utiliza la cláusula SHOW DETAIL.

| Nombre de columna | Tipo de datos | Descripción |
|---------------------|---------------|--|
| DATA_PARTITION_ID | INTEGER | Identificador de partición de datos |
| DATA_PARTITION_NAME | VARCHAR (128) | Nombre de la partición de datos. |
| TBSPID | INTEGER | Identificador del espacio de tablas en el que se almacena la partición de datos. |
| PARTITION_OBJECT_ID | INTEGER | Identificador del objeto DMS en el que se almacena la partición de datos. |
| LONG_TBSPID | INTEGER | Identificador del espacio de tablas en el que se almacenan los datos largos. |
| INDEX_TBSPID | INTEGER | Identificador del espacio de tablas en el que se almacenan los datos de índice. |
| ACCESSMODE | VARCHAR (20) | Define la accesibilidad de la partición de datos y uno de estos valores: <ul style="list-style-type: none"> • FULL_ACCESS • NO_ACCESS • NO_DATA_MOVEMENT • READ_ONLY |
| STATUS | VARCHAR(64) | Estado de la partición de datos y puede ser uno de estos valores: <ul style="list-style-type: none"> • NEWLY_ATTACHED • NEWLY_DETACHED: se necesita mantenimiento de MQT. • INDEX_CLEANUP_PENDING: partición de datos desenlazada cuya tupla en SYSDATAPARTITIONS sólo se mantiene para la limpieza del índice. Esta tupla se elimina cuando todos los registros de índice que hacen referencia a la partición de datos desenlazada se han eliminado. De lo contrario, la columna está en blanco. |

Mandato DROP CONTACT utilizando el procedimiento ADMIN_CMD

Elimina un contacto de la lista de contactos definidos en el sistema local. Un contacto es un usuario al que el Planificador y el Supervisor de salud envían mensajes. El valor del parámetro de configuración **contact_host** del Servidor de administración de bases de datos (DAS) determina si la lista es local o global.

Autorización

Ninguna

Conexión necesaria

Base de datos. El DAS debe estar en ejecución.

Sintaxis del mandato

►►—DROP CONTACT—*nombre*—◄◄

Parámetros del mandato

CONTACT *nombre*

Nombre del contacto que se descartará del sistema local.

Ejemplo

Descartar el contacto 'testuser' de la lista de contactos existente en el sistema servidor.

```
CALL SYSPROC.ADMIN_CMD( 'drop contact testuser' )
```

Notas de uso

El DAS ya se debe haber creado y estar en ejecución.

El estado de la ejecución del mandato se devuelve en el área de comunicaciones SQL (SQLCA) obtenida a partir de la sentencia CALL.

Mandato DROP CONTACTGROUP utilizando el procedimiento ADMIN_CMD

Elimina un grupo de contactos de la lista de contactos definidos en el sistema local. Un grupo de contactos contiene una lista de usuarios a los que el Planificador y el Supervisor de salud envían mensajes. El valor del parámetro de configuración **contact_host** del Servidor de administración de bases de datos (DAS) determina si la lista es local o global.

Autorización

Ninguna

Conexión necesaria

Base de datos. El DAS debe estar en ejecución.

Sintaxis del mandato

►►—DROP CONTACTGROUP—*nombre*—◄◄

Parámetros del mandato

CONTACTGROUP *nombre*

Nombre del grupo de contactos que se descartará del sistema local.

Ejemplo

Descartar el grupo de contactos llamado 'gname1'.

```
CALL SYSPROC.ADMIN_CMD( 'drop contactgroup gname1' )
```

Notas de uso

El DAS ya se debe haber creado y estar en ejecución.

El estado de la ejecución del mandato se devuelve en el área de comunicaciones SQL (SQLCA) obtenida a partir de la sentencia CALL.

Mandato EXPORT utilizando el procedimiento ADMIN_CMD

Exporta datos de una base de datos a uno de varios formatos de archivo externos. El usuario especifica los datos que se deben exportar proporcionando una sentencia SELECT de SQL o proporcionando información jerárquica para tablas de tipo. Los datos solo se exportan al servidor.

Enlace rápido con “Modificadores de tipo de archivo para el programa de utilidad de exportación” en la página 77.

Autorización

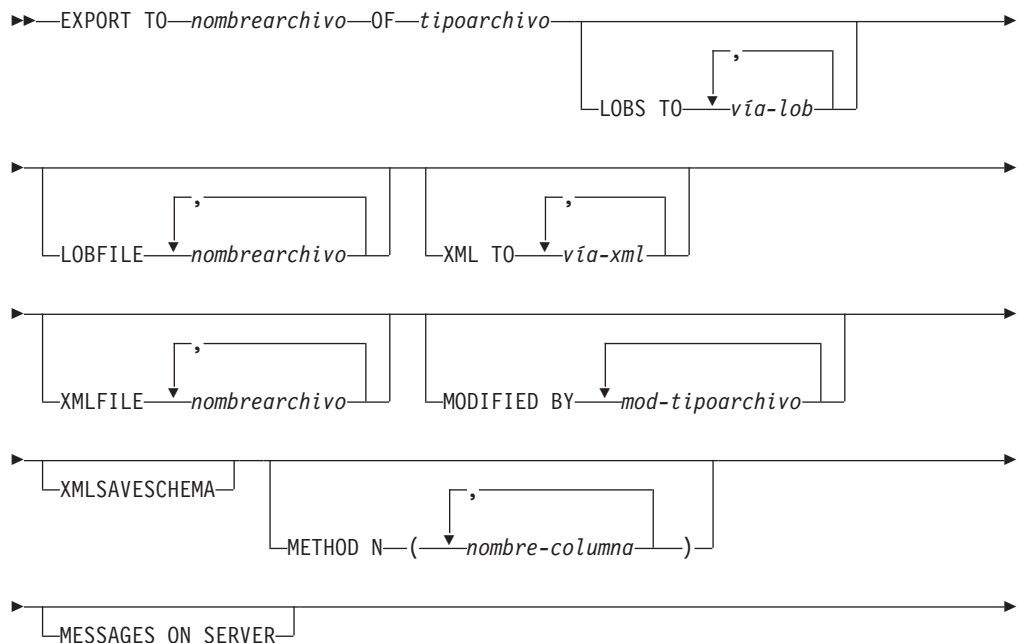
Una de las siguientes:

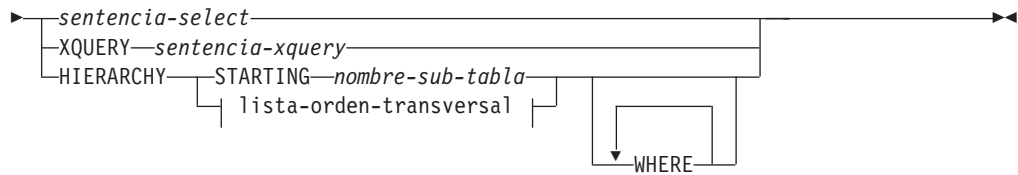
- Autorización *dataaccess*
- Privilegio CONTROL o SELECT en cada tabla o vista participante

Conexión necesaria

Base de datos. El acceso del programa de utilidad a servidores de bases de datos Linux, UNIX o Windows desde clientes Linux, UNIX o Windows debe ser una conexión directa a través del motor y no a través de un entorno de pasarela o de bucle de retorno de DB2 Connect.

Sintaxis del mandato





lista-orden-transversal:



Parámetros del mandato

HIERARCHY *lista-orden-transversal*

Exporta una subjerarquía utilizando el orden transversal especificado. Todas las subtablas deben listarse de modo PRE-ORDER. El primer nombre de subtabla se utiliza como nombre de tabla de destino para la sentencia SELECT.

HIERARCHY STARTING *nombre-subtabla*

Mediante el uso del orden transversal por omisión (orden OUTER para archivos ASC, DEL o WSF o el orden almacenado en los archivos de datos PC/IXF), exporta una subjerarquía empezando desde *nombre-subtabla*.

LOBBFILE *nombrearchivo*

Especifica uno o más nombres de archivo base para los archivos LOB. Cuando se ha agotado el espacio de nombres para el primer nombre, se utiliza el segundo nombre y así sucesivamente. Esto activará implícitamente el comportamiento de LOBSINFILE.

Cuando se crean archivos LOB durante una operación de exportación, los nombres de archivo se construyen añadiendo el nombre base actual de esta lista a la vía de acceso actual (de *vía-lob*) y a continuación, añadiendo un número de secuencia de 3 dígitos de inicio y el identificador de tres caracteres lob. Por ejemplo, si la vía de acceso de LOB actual es el directorio /u/foo/lob/path/ y el nombre de archivo LOB actual es bar, los archivos LOB creados serán /u/foo/lob/path/bar.001.lob, /u/foo/lob/path/bar.002.lob y así sucesivamente. El número de secuencia de 3 dígitos del nombre de archivo LOB aumentará a 4 dígitos después de utilizarse 999, los 4 dígitos aumentarán a 5 dígitos después de utilizarse 9999, etc.

LOBS TO *vía-lob*

Especifica una o más vías de acceso a los directorios en los que deben almacenarse los archivos LOB. La vía o vías de acceso deben existir en la partición coordinadora del servidor y deben estar totalmente calificadas. Habrá por lo menos un archivo por vía de acceso de LOB y cada archivo contendrá por lo menos un LOB. El número máximo de vías de acceso que se puede especificar es 999. Esto activará implícitamente el comportamiento de LOBSINFILE.

MESSAGES ON SERVER

Especifica que hay que guardar el archivo de mensajes creado en el servidor mediante el mandato EXPORT. El conjunto de resultados devuelto incluirá las dos columnas siguientes: MSG_RETRIEVAL, que es la sentencia de SQL necesaria para recuperar todos los mensajes de aviso y de error

producidos durante esta operación, y MSG_REMOVAL, que es la sentencia de SQL necesaria para hacer limpieza de los mensajes.

Si no se especifica esta cláusula, el archivo de mensajes se suprimirá cuando el procedimiento ADMIN_CMD retorne al llamador. La columna MSG_RETRIEVAL y la columna MSG_REMOVAL del conjunto de resultados contendrán valores nulos.

Observe que, con cláusula o sin ella, el ID de usuario delimitado debe poseer autorización para crear archivos bajo el directorio indicado por la variable de registro DB2_UTIL_MSGPATH, así como bajo el directorio al que hay que exportar los datos.

METHOD N *nombre-columna*

Especifica uno o más nombres de columna que se deben utilizar en el archivo de salida. Si no se especifica este parámetro, se utilizan los nombres de columna de la tabla. Este parámetro solo es válido para archivos WSF e IXF, pero no es válido al exportar datos jerárquicos.

MODIFIED BY *mod-tipoarchivo*

Especifica opciones de modificador de tipo de archivo. Véase “Modificadores de tipo de archivo para el programa de utilidad de exportación” en la página 77.

OF *tipoarchivo*

Especifica el formato de los datos del archivo de salida:

- DEL (formato ASCII delimitado), utilizado por diversos programas del gestor de bases de datos y gestores de archivos.
- WSF (formato de hoja de trabajo), utilizado por programas tales como:
 - Lotus 1-2-3
 - Lotus Symphony

Al exportar datos BIGINT o DECIMAL, solo se pueden exportar de forma precisa los valores que están en el rango de tipo DOUBLE. Aunque los valores que no están en este rango también se exportan, si se vuelven a importar o a cargar dichos valores, los datos resultantes podrían ser incorrectos, según el sistema operativo.

Nota: El soporte para el formato de archivo WSF ha quedado en desuso y se puede eliminar en futuros releases. Se recomienda empezar a usar un formato de archivo compatible en lugar de archivos WSF antes de que se elimine el soporte.

- IXF (Integration Exchange Format, versión PC) es un formato binario con propietario.

sentencia-select

Especifica la sentencia SELECT o XQUERY que devolverá los datos que se deben exportar. Si la sentencia provoca un error, se grabará un mensaje en el archivo de mensajes (o en la salida estándar). Si el código de error es uno de SQL0012W, SQL0347W, SQL0360W, SQL0437W o SQL1824W, la operación de exportación continuará; de lo contrario, se detendrá.

TO *nombreakchivo*

Especifica el nombre del archivo al que hay que exportar datos en el servidor. Debe ser una vía de acceso totalmente calificada y debe existir en la partición coordinadora del servidor.

Si se especifica el nombre de un archivo que ya existe, el programa de utilidad de exportación grabará encima del contenido del archivo; no añadirá la información.

XMLFILE *nombreamarchivo*

Especifica uno o más nombres de archivo base para los archivos XML. Cuando se ha agotado el espacio de nombres para el primer nombre, se utiliza el segundo nombre y así sucesivamente.

Cuando se crean archivos XML durante una operación de exportación, los nombres de archivo se construyen añadiendo el nombre base actual de esta lista a la vía de acceso actual (de *vía-xml*), añadiendo un número de secuencia de 3 dígitos y luego el identificador de tres caracteres xml. Por ejemplo, si la vía de acceso de XML actual es el directorio /u/foo/xml/path/ y el nombre de archivo XML actual es bar, los archivos XML creados serán /u/foo/xml/path/bar.001.xml, /u/foo/xml/path/bar.002.xml y así sucesivamente.

XML TO *vía-xml*

Especifica una o más vías de acceso a los directorios en los que deben almacenarse los archivos XML. Habrá como mínimo un archivo por cada vía de acceso de XML, y cada archivo contendrá como mínimo una instancia de modelo de datos XQuery (XDM). Si se especifica más de una vía de acceso, las instancias de XDM se distribuyen uniformemente entre las vías de acceso.

XMLSAVESCHEMA

Especifica que hay que guardar información de esquema XML para todas las columnas XML. Para cada documento XML exportado que se haya validado con respecto a un esquema XML en el momento de insertarlo, se almacenará el identificador SQL totalmente calificado de ese esquema en forma de atributo (SCH) dentro del especificador de datos XML correspondiente (XDS). Si el documento exportado no se ha validado con respecto a un esquema XML, no se incluirá un atributo SCH en el correspondiente XDS.

Las partes de esquema y nombre del identificador SQL se almacenan como valores "OBJECTSCHEMA" y "OBJECTNAME" en la fila de la tabla de catálogo SYSCAT.XSROBJECTS correspondiente al esquema XML.

La opción XMLSAVESCHEMA no es compatible con las secuencias XQuery que no producen documentos XML bien formados.

Ejemplo

El siguiente ejemplo muestra cómo exportar información de la tabla STAFF de la base de datos SAMPLE al archivo myfile.ixf. La salida estará en formato IXF. Debe estar conectado a la base de datos SAMPLE antes de emitir el mandato.

```
CALL SYSPROC.ADMIN_CMD ('EXPORT to /home/user1/data/myfile.ixf
OF ixf MESSAGES ON SERVER select * from staff')
```

Notas de uso

- Las vías de acceso que se utilicen en el mandato EXPORT deben ser válidas en el servidor y estar totalmente calificadas.
- Si una tabla contiene columnas LOB, hay que especificar como mínimo una vía de acceso de LOB totalmente calificada y un nombre de LOB, utilizando para ello las cláusulas LOBS TO y LOBFILE.

- El programa de utilidad de exportación emite una sentencia COMMIT al principio de la operación; esta sentencia, en el caso de las conexiones de tipo 2, hace que el procedimiento devuelva SQL30090N con el código de razón 2.
- Al exportar desde una base de datos UCS-2 a un archivo ASCII delimitado (DEL), todos los datos de tipo carácter se convierten a la página de códigos que está en vigor en el lugar en el que se ejecuta el procedimiento. Los datos de tipo serie y los de tipo gráfico se convierten a la misma página de códigos SBCS o MBCS del servidor.
- Asegúrese de completar todas las operaciones de tabla y de liberar todos los bloqueos antes de iniciar una operación de exportación. Esto puede realizarse emitiendo un mandato COMMIT después de cerrar todos los cursores abiertos WITH HOLD o emitiendo un mandato ROLLBACK.
- Se pueden utilizar alias de tabla en la sentencia SELECT.
- Los mensajes colocados en el archivo de mensajes incluyen la información devuelta del servicio de recuperación de mensajes. Cada mensaje empieza en una línea nueva.
- El programa de utilidad de exportación produce un mensaje de aviso siempre que se selecciona una columna de caracteres con una longitud de más de 254 para la exportación a archivos de formato DEL.
- Se deberá utilizar la importación PC/IXF para mover datos entre bases de datos. Si los datos de tipo carácter que contienen separadores de filas se exportan a un archivo ASCII delimitado (DEL) y se procesan con un programa de transferencia de texto, los campos que contengan separadores de filas se acortarán o se ampliarán.
- El paso de copia de archivo no es necesario si se puede acceder a las bases de datos fuente y destino desde el mismo cliente.
- Se puede utilizar DB2 Connect para exportar tablas de servidores DRDA, como por ejemplo, DB2 para OS/390, DB2 para VM y VSE, y DB2 para OS/400. Sólo se soporta la exportación PC/IXF.
- Cuando se exporta al formato IXF, si los identificadores superan el tamaño máximo al que da soporte el formato IXF, la exportación se realizará correctamente pero el archivo de datos resultante sólo se podrá utilizar en una operación de importación siguiente utilizando la modalidad CREATE. Se devolverá SQL27984W.
- Cuando se exporta a un disquete en Windows, y si la tabla contiene más datos de los que caben en un solo disquete, el sistema solicitará otro disquete y los archivos PC/IXF de varias partes (también denominados archivos PC/IXF multivolumen o archivos PC/IXF divididos lógicamente) se generarán y se almacenarán en distintos disquetes. En cada archivo, con la extensión del último, hay un REGISTRO DE CONTINUACIÓN DE DB2 CONTINUATION (registro "AC" abreviado) que indica que los archivos están lógicamente divididos y dónde encontrar el siguiente archivo. Luego los archivos se pueden transferir a un sistema AIX para que los lean los programas de utilidad de importación y de carga. El programa de utilidad de exportación no creará archivos PC/IXF de múltiples componentes cuando se invoque desde un sistema AIX. Para ver detalles sobre el uso, consulte el mandato IMPORT o el mandato LOAD.
- El programa de utilidad de exportación almacenará el atributo NOT NULL WITH DEFAULT de la tabla en un archivo IXF si la sentencia SELECT proporcionada está en el formato SELECT * FROM nombretabla.
- Cuando se exportan tablas de tipo, solo se pueden expresar sentencias de subselección especificando el nombre de tabla de destino y la cláusula WHERE. La selección completa y la *sentencia-select* no se pueden especificar cuando se exporta una jerarquía.

- Para formatos de archivos distintos de IXF, se recomienda especificar la lista de orden transversal porque indica a DB2 cómo atravesar la jerarquía y qué subtablas se deben exportar. Si no se especifica dicha lista, se exportan todas las tablas de la jerarquía y el orden por omisión es el orden de OUTER. La alternativa consiste en utilizar el orden por omisión, que es el orden proporcionado por la función OUTER.
- Utilice el mismo orden transversal durante una operación de importación. El programa de utilidad de carga no soporta la carga de jerarquías o de subjerarquías.
- Al exportar datos de una tabla que tenga filas protegidas, las credenciales LBAC retenidas por el ID de autorización de sesión podrían limitar las filas que se exportan. Las filas a las que el ID de autorización de sesión no tiene acceso de lectura no se exportarán. No se proporciona ningún código de error ni de aviso.
- Si las credenciales LBAC retenidas por el ID de autorización de sesión no permiten leer una o más columnas protegidas incluidas en la exportación, la exportación falla y se devuelve un error (SQLSTATE 42512).
- Los paquetes de exportación se enlazan con el formato DATETIME ISO, por lo tanto, todos los valores de fecha/hora/indicación de fecha y hora se convierten al formato ISO cuando se convierte a una representación de serie. Puesto que los paquetes de CLP se enlazan con el formato DATETIME LOC (formato específico del entorno local), tal vez detecte un comportamiento incoherente entre CLP y la exportación, si el formato CLP DATETIME es diferente de ISO. Por ejemplo, la siguiente sentencia SELECT puede devolver resultados imprevistos:

```
db2 select col2 from tab1 where char(col2)='05/10/2005';
COL2
-----
05/10/2005
05/10/2005
05/10/2005
3 registro(s) seleccionado(s).
```

Pero un mandato de exportación con la misma cláusula select no realizará lo siguiente:

```
db2 export to test.del of del select col2 from test
where char(col2)='05/10/2005';
Número de filas exportadas: 0
```

Ahora, si se sustituye el formato de fecha LOCALE por el formato ISO, se producen los resultados previstos:

```
db2 export to test.del of del select col2 from test
where char(col2)='2005-05-10';
Número de filas exportadas: 3
```

Información de conjunto de resultados

El estado de la ejecución del mandato se devuelve en el área de comunicaciones SQL (SQLCA) obtenida a partir de la sentencia CALL. Si la ejecución es satisfactoria, el mandato devuelve información adicional en conjuntos de resultados, de la siguiente manera:

Tabla 36. Conjunto de resultados devuelto por el mandato EXPORT

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|-----------------------------------|
| ROWS_EXPORTED | BIGINT | Número total de filas exportadas. |

Tabla 36. Conjunto de resultados devuelto por el mandato EXPORT (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| MSG_RETRIEVAL | VARCHAR(512) | Sentencia de SQL que sirve para recuperar los mensajes creados por este programa de utilidad. Por ejemplo: <pre>SELECT SQLCODE, MSG FROM TABLE (SYSPROC.ADMIN_GET_MSGS ('3203498_txu')) AS MSG</pre> |
| MSG_REMOVAL | VARCHAR(512) | Sentencia de SQL que sirve para hacer limpieza de los mensajes creados por este programa de utilidad. Por ejemplo: <pre>CALL SYSPROC.ADMIN_REMOVE_MSGS ('3203498_txu')</pre> |

Modificadores de tipo de archivo para el programa de utilidad de exportación

Tabla 37. Modificadores de tipo de archivo válidos para el programa de utilidad de exportación: Todos los formatos de archivo

| Modificador | Descripción |
|------------------|---|
| lobsinfile | <p><i>vía-lob</i> especifica la vía de acceso a los archivos que contienen datos de LOB.</p> <p>Cada vía de acceso contiene por lo menos un archivo que contiene por lo menos un LOB al que apunta un Especificador de ubicación de LOB (LLS) en el archivo de datos. El LLS es una representación de serie de la ubicación de un LOB en un archivo almacenado en la vía de acceso del archivo LOB. El formato de un LLS es <i>nombarch.ext.nnn.mmm/</i>, donde <i>nombarch.ext</i> es el nombre del archivo que contiene el LOB, <i>nnn</i> es el desplazamiento en bytes del LOB dentro del archivo y <i>mmm</i> es la longitud del LOB en bytes. Por ejemplo, si se almacena la serie <i>db2exp.001.123.456/</i> en el archivo de datos, el LOB está ubicado en el desplazamiento 123 dentro del archivo <i>db2exp.001</i> tiene una longitud de 456 bytes.</p> <p>Si se especifica el modificador "lobsinfile" cuando se utiliza EXPORT, los datos de LOB se encuentran en las ubicaciones especificadas por la cláusula LOBS TO. De lo contrario, se envían datos LOB al directorio del archivo de datos. La cláusula LOBS TO especifica una o más vías de acceso a los directorios en los que deben almacenarse los archivos LOB. Habrá por lo menos un archivo por vía de acceso de LOB y cada archivo contendrá por lo menos un LOB. Las opciones LOBS TO o LOBFILE activarán implícitamente el comportamiento de LOBSINFILE.</p> <p>Para indicar un LOB nulo, entre el tamaño como -1. Si el tamaño se especifica como 0, se trata como un LOB de longitud 0. Para los LOB nulos de longitud -1, se pasan por alto el desplazamiento y el nombre de archivo. Por ejemplo, el LLS de un LOB nulo puede ser <i>db2exp.001.7.-1/</i>.</p> |
| xmlinsefiles | Cada instancia XQuery Data Model (XDM) se graba en un archivo aparte. Por omisión, se concatenan varios valores juntos en el mismo archivo. |
| lobsinsefiles | Cada valor LOB se graba en un archivo aparte. Por omisión, se concatenan varios valores juntos en el mismo archivo. |
| xmlnodeclaration | Las instancias XDM se graban sin un identificador de declaración de XML. Por omisión, las instancias XDM se exportan con un identificador de declaración de XML al principio que incluye un atributo de codificación. |
| xmlchar | Las instancias XDM se graban en la página de códigos de caracteres. Tenga en cuenta que la página de códigos es el valor especificado por el modificador de tipo de archivo codepage o la página de códigos de la aplicación, si no se ha especificado. Por omisión, las instancias de XDM se escriben en Unicode. |

Tabla 37. Modificadores de tipo de archivo válidos para el programa de utilidad de exportación: Todos los formatos de archivo (continuación)

| Modificador | Descripción |
|-------------|---|
| xmlgraphic | Si se especifica el modificador <code>xmlgraphic</code> con el mandato EXPORT, el documento XML exportado se codificará con la página de códigos UTF-16 independientemente de la página de códigos de la aplicación o el modificador de tipo de archivo <code>codepage</code> . |

Tabla 38. Modificadores de tipo de archivo válidos para el programa de utilidad de exportación: Formato de archivo DEL (ASCII delimitado)

| Modificador | Descripción |
|--------------|--|
| chardelx | <p><code>x</code> es un delimitador de serie de un solo carácter. El valor por omisión son las comillas dobles (""). El carácter especificado se utiliza en lugar de las comillas dobles para delimitar una serie.² Si desea especificar explícitamente las comillas dobles como delimitador de serie de caracteres, debe especificarse de la manera siguiente:</p> <pre>modified by chardel""</pre> <p>También se pueden especificar las comillas simples (') como delimitador de serie de caracteres del modo siguiente:</p> <pre>modified by chardel''</pre> |
| codepage=x | <p><code>x</code> es una serie de caracteres ASCII. El valor se interpreta como la página de códigos de los datos del archivo de salida. Convierte datos de tipo carácter a esta página de códigos desde la página de códigos de la aplicación durante la operación de exportación.</p> <p>Para DBCS (gráfico) puro, DBCS mixto y EUC, los delimitadores están restringidos al rango de <code>x00</code> a <code>x3F</code>, inclusive. El modificador <code>codepage</code> no se puede utilizar con el modificador <code>lobsinfile</code>.</p> |
| coldelx | <p><code>x</code> es un delimitador de columna de un solo carácter. El valor por omisión es una coma (.). Se utiliza el carácter especificado en lugar de una coma para indicar el final de una columna.²</p> <p>En el ejemplo siguiente, <code>coldel;</code> hace que el programa de utilidad de exportación utilice el carácter de punto y coma (;) como un delimitador de columna para los datos exportados:</p> <pre>db2 "export to temp of del modified by coldel; select * from staff where dept = 20"</pre> |
| decplusblank | Carácter de signo más. Hace que a los valores decimales positivos se les ponga un espacio en blanco como prefijo en lugar de un signo más (+). La acción por omisión es poner a los valores decimales positivos un signo más como prefijo. |
| decptx | <code>x</code> es un sustituto de un solo carácter del punto como carácter de coma decimal. El valor por omisión es un punto (.). Se utiliza el carácter especificado como carácter de coma decimal. ² |
| nochardel | <p>Los datos de columna no estarán rodeados por delimitadores de caracteres. No se debe especificar esta opción si los datos se van a importar o cargar utilizando DB2. Se proporciona para dar soporte a archivos de datos de proveedores que no tienen delimitadores de caracteres. El uso incorrecto podría causar la pérdida o la corrupción de los datos.</p> <p>Esta opción no se puede especificar con <code>chardelx</code> o <code>nodoubledel</code>. Son opciones que se excluyen mutuamente.</p> |
| nodoubledel | Suprime el reconocimiento de los delimitadores de caracteres dobles. ² |

Tabla 38. Modificadores de tipo de archivo válidos para el programa de utilidad de exportación: Formato de archivo DEL (ASCII delimitado) (continuación)

| Modificador | Descripción |
|---------------------|---|
| timestampformat="x" | <p>x es el formato de la indicación de fecha y hora en el archivo fuente.⁴ Los elementos de indicación de fecha y hora son:</p> <ul style="list-style-type: none"> AAAA - Año (cuatro dígitos entre 0000 y 9999) M - Mes (uno o dos dígitos entre 1 y 12) MM - Mes (dos dígitos entre 01 y 12; se excluye mutuamente con M y MMM) MMM - Mes (abreviación de tres letras no sensible a mayúsculas y minúsculas para el nombre del mes; se excluye mutuamente con M y MM) D - Día (uno o dos dígitos entre 1 y 31) DD - Día (dos dígitos entre 1 y 31; se excluye mutuamente con D) DDD - Día del año (tres dígitos entre 001 y 366; se excluye mutuamente con otros elementos de día o mes) H - Hora (uno o dos dígitos entre 0 y 12 para un sistema de 12 horas y entre 0 y 24 para un sistema de 24 horas) HH - Hora (dos dígitos entre 0 y 12 para un sistema de 12 horas y entre 0 y 24 para un sistema de 24 horas; se excluye mutuamente con H) M - Minuto (uno o dos dígitos entre 0 y 59) MM - Minuto (dos dígitos entre 0 y 59; se excluye mutuamente con M, minuto) S - Segundo (uno o dos dígitos entre 0 y 59) SS - Segundo (dos dígitos entre 0 y 59; se excluye mutuamente con S) SSSSS - Segundo del día después de medianoche (5 dígitos entre 00000 y 86399; se excluye mutuamente con otros elementos de hora) U (de 1 a 12 veces) <ul style="list-style-type: none"> - Segundos fraccionados (el número de casos de U representa el número de dígitos con cada dígito comprendido entre 0 y 9) TT - Indicador de meridiano (AM o PM) <p>A continuación se muestra un ejemplo de un formato de indicación de la hora: "AAAA/MM/DD HH:MM:SS.UUUUUU"</p> <p>El elemento MMM producirá los valores siguientes: 'Ene', 'Feb', 'Mar', 'Abr', 'May', 'Jun', 'Jul', 'Ago', 'Sep', 'Oct', 'Nov', y 'Dic'. 'Ene' es igual al mes 1, y 'Dic' es igual al mes 12.</p> <p>El ejemplo siguiente muestra un ejemplo de cómo exportar datos que contengan formatos de indicación de fecha y hora definidos por el usuario desde una tabla denominada 'schedule':</p> <pre>db2 export to delfile2 of del modified by timestampformat="aaaa.mm.dd hh:mm tt" select * from schedule</pre> |

Tabla 39. Modificadores de tipo de archivo válidos para el programa de utilidad de exportación: Formato de archivo IXF

| Modificador | Descripción |
|-------------|---|
| codepage=x | <p>x es una serie de caracteres ASCII. El valor se interpreta como la página de códigos de los datos del archivo de salida. Convierte los datos de tipo carácter de esta página de códigos a la página de códigos de la aplicación durante la operación de exportación.</p> <p>Para DBCS (gráfico) puro, DBCS mixto y EUC, los delimitadores están restringidos al rango de x00 a x3F, inclusive. El modificador codepage no se puede utilizar con el modificador lobsinfile.</p> |

Tabla 40. Modificadores de tipo de archivo válidos para el programa de utilidad de exportación: Formato de archivo WSF⁶

| Modificador | Descripción |
|-------------|--|
| 1 | Crea un archivo WSF compatible con Lotus 1-2-3 Release 1 o Lotus 1-2-3 Release 1a. ⁵ Es el valor por omisión. |
| 2 | Crea un archivo WSF compatible con Lotus Symphony Release 1.0. ⁵ |
| 3 | Crea un archivo WSF compatible con Lotus 1-2-3 Versión 2 o Lotus Symphony Release 1.1. ⁵ |
| 4 | Crea un archivo WSF que contiene caracteres DBCS. |

Nota:

1. El programa de utilidad de exportación no emite ningún aviso si se intentan utilizar tipos de archivo no soportados con la opción MODIFIED BY. Si se intenta realizar dicha acción, la operación de exportación falla y se devuelve un código de error.
2. En la sección *Consideraciones sobre el delimitador para mover datos* se muestran las restricciones que se aplican a los caracteres que se pueden utilizar como alteración temporal de los delimitadores.
3. El programa de utilidad de exportación graba normalmente los
 - datos de fecha en formato AAAAMMDD
 - datos char(fecha) en formato "AAAA-MM-DD"
 - datos de hora en formato "HH.MM.SS"
 - datos de la indicación de fecha y hora en formato "AAAA-MM-DD-HH.MM.SS.aaaaaa"

Los datos contenidos en cualquier columna de fecha y hora especificada en la sentencia SELECT para la operación de exportación también tendrán estos formatos.

4. Para formatos de indicación de la hora, hay que tener cuidado de evitar la ambigüedad entre los descriptores de mes y de minuto, dado que ambos utilizan la letra M. Un campo de mes debe estar junto a otros campos de fecha. Un campo de minuto debe ser adyacente a otros campos de hora. He aquí unos formatos ambiguos de indicación de fecha y hora:

"M" (puede ser mes o minuto)
 "M:M" (¿Cuál es cada uno?)
 "M:AAAA:M" (Ambos se interpretan como mes.)
 "S:M:AAAA" (adyacente a un valor de hora y un valor de fecha)

En los casos ambiguos, el programa de utilidad informará con un mensaje de error y la operación fallará.

He aquí unos formatos no ambiguos de indicación de fecha y hora:

"M:AAAA" (Mes)
"S:M" (Minuto)
"M:AAAA:S:M" (Mes...Minuto)
"M:H:AAAA:M:D" (Minuto...Mes)

5. Estos archivos también pueden dirigirse a un producto determinado especificando una L para Lotus 1-2-3 o una S para Symphony en la serie de parámetros *mod-tipoarchivo*. Sólo se puede especificar un valor o indicador de producto. El soporte para el formato de archivo WSF ha quedado en desuso y se puede eliminar en futuros releases. Se recomienda empezar a usar un formato de archivo compatible en lugar de archivos WSF antes de que se elimine el soporte.
6. El formato de archivo WSF no está soportado para columnas de XML. El soporte para este formato de archivo ha quedado en desuso y se puede eliminar en futuros releases. Se recomienda empezar a usar un formato de archivo compatible en lugar de archivos WSF antes de que se elimine el soporte.
7. Todas las instancias de XDM se graban en archivos XML aparte del archivo de datos principal, incluso si no se ha especificado la cláusula XMLFILE ni XMLTO. Por omisión, los archivos XML se graban en la vía de acceso del archivo de datos exportado. El nombre base por omisión para los archivos XML es el nombre del archivo de datos exportado con la extensión ".xml" añadida.
8. Todas las instancias XDM se graban con una declaración de XML al principio que incluye un atributo de codificación, a menos que se especifique el modificador de tipo de archivo XMLNODECLARATION.
9. Por omisión, todas las instancias de XDM se escriben en Unicode, a menos que se especifique el modificador de tipo de archivo XMLCHAR o XMLGRAPHIC.
10. La vía de acceso por omisión para datos XML y datos LOB es la vía de acceso del archivo de datos principal. El nombre base del archivo XML por omisión es el archivo de datos principal. El nombre base del archivo LOB por omisión es el archivo de datos principal. Por ejemplo, si el archivo de datos principal es:

```
/mypath/myfile.del
```

, la vía de acceso por omisión para datos XML y datos LOB es:

```
/mypath"
```

el nombre base del archivo XML por omisión es:

```
myfile.del
```

y el nombre base del archivo LOB por omisión es:

```
myfile.del
```

Debe especificarse el modificador de tipo de archivo LOBSINFILE para poder generar los archivos LOB.

11. El programa de utilidad de exportación añade un identificador numérico a cada archivo LOB o archivo XML. El identificador comienza como un valor de secuencia de 3 dígitos, rellenado con 0, que empieza en:
.001

Después del archivo LOB o archivo XML 999, el identificador ya no estará rellenado con ceros (por ejemplo el archivo LOG o archivo XML 1000 tendrá una extensión de:

.1000

A continuación del identificador numérico hay un identificador de tipo de tres caracteres que representa el tipo de datos, ya sea:

.lob

o

.xml

Por ejemplo, un archivo LOB generado tendría un nombre con formato:

myfile.del.001.lob

y un archivo XML generado tendría un nombre con formato:

myfile.del.001.xml

12. Es posible hacer que el programa de utilidad de exportación exporte las instancias de XDM que no sean documentos con formato correcto especificando una XQuery. No obstante, no podrá importar ni cargar estos documentos exportados directamente a una columna XML, ya que las columnas XML solamente pueden contener documentos completos.

Mandato FORCE APPLICATION utilizando el procedimiento ADMIN_CMD

Fuerza a las aplicaciones o usuarios locales o remotos a salir del sistema para permitir tareas de mantenimiento en un servidor.

Atención: Si se fuerza una operación que no se puede interrumpir (por ejemplo RESTORE DATABASE), se deberá volver a ejecutar satisfactoriamente la operación para que la base de datos quede disponible.

Ámbito

Este mandato afecta a todas las particiones de base de datos que se listan en el archivo `$HOME/sql11ib/db2nodes.cfg`.

En un entorno de bases de datos particionadas, este mandato no tiene que emitirse desde la partición de base de datos coordinadora de la aplicación que se está forzando. Puede emitirse desde cualquier nodo (servidor de particiones de base de datos) del entorno de bases de datos particionadas.

Autorización

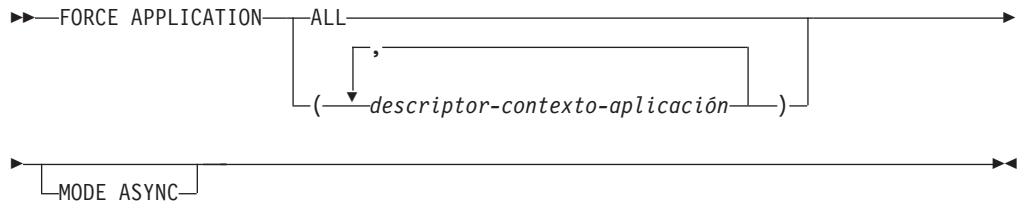
Una de las siguientes:

- *sysadm*
- *sysctrl*
- *sysmaint*

Conexión necesaria

Base de datos

Sintaxis del mandato



Parámetros del mandato

FORCE APPLICATION

ALL Se desconectarán todas las aplicaciones de la base de datos. Esto podría cerrar la conexión en la que se ejecuta el procedimiento ADMIN_CMD, lo que provoca la devolución de un error SQL1224N en relación al procedimiento ADMIN_CMD una vez completada satisfactoriamente la operación de forzar.

descriptor-aplicación

Especifica el agente que se debe terminar. Liste los valores mediante el mandato LIST APPLICATIONS.

MODE ASYNC

El mandato no espera a que todos los usuarios especificados terminen para volver; vuelve tan pronto como se ha emitido satisfactoriamente la función o se ha descubierto un error (por ejemplo sintaxis no válida).

Esta es la única modalidad que se soporta actualmente.

Ejemplos

El ejemplo siguiente fuerza a dos usuarios, con valores de *application-handle* igual a 41408 y 55458, a desconectarse de la base de datos:

```
CALL SYSPROC.ADMIN_CMD( 'force application ( 41408, 55458 )' )
```

Notas de uso

El gestor de bases de datos permanece activo para que las operaciones del gestor de bases de datos subsiguientes puedan manejarse sin necesidad de ejecutar db2start.

Para conservar la integridad de la base de datos, solo se pueden interrumpir los usuarios que están desocupados o que están ejecutando operaciones de base de datos que se pueden interrumpir.

Los siguientes tipos de usuario y aplicaciones no se pueden forzar.

- usuarios que crean una base de datos
- aplicaciones del sistema

Para poder forzar a estos tipos de usuarios y aplicaciones de manera satisfactoria, se debe desactivar la base de datos y/o reiniciar la instancia.

Después de haber emitido un FORCE APPLICATION, la base de datos seguirá aceptando peticiones de conexión. Podrían ser necesarias operaciones de forzar adicionales para forzar a salir completamente a todos los usuarios.

El estado de la ejecución del mandato se devuelve en el área de comunicaciones SQL (SQLCA) obtenida a partir de la sentencia CALL.

Mandato GET STMM TUNING DBPARTITIONNUM mediante el procedimiento ADMIN_CMD

Se utiliza para leer tablas de catálogos para informar acerca del número de partición de base de datos de ajuste STMM (Self Tuning Memory Manager) preferido por el usuario y el número de partición de base de datos de ajuste STMM actual.

Autorización

Los privilegios que posee el ID de autorización de la sentencia deben incluir al menos uno de los privilegios o autorizaciones siguientes:

- DBADM
- SECADM
- SQLADM
- ACCESSCTRL
- DATAACCESS
- SELECT en SYSIBM.SYSTUNINGINFO

Conexión necesaria

Base de datos

Sintaxis del mandato

►► GET STMM TUNING DBPARTITIONNUM ◀◀

Ejemplo

```
CALL SYSPROC.ADMIN_CMD( 'get stmm tuning dbpartitionnum' )
```

El ejemplo siguiente muestra los datos de salida de la consulta.

Conjunto de resultados 1

```
-----  
USER_PREFERRED_NUMBER CURRENT_NUMBER  
-----  
2 2
```

1 registro(s) seleccionado(s).

Estado de devolución = 0

Notas de uso

El número de partición de base de datos de ajuste STMM (Self Tuning Memory Manager) preferido por el usuario (USER_PREFERRED_NUMBER) lo establece el usuario y especifica la partición de base de datos en la que el usuario desea ejecutar el ajustador de memoria. Mientras se ejecuta la base de datos, cada hora se actualiza la partición de ajuste de forma asíncrona un número de veces. Como resultado, es posible que el valor de CURRENT_NUMBER y USER_PREFERRED_NUMBER devuelto no se haya sincronizado después de una

actualización del número de partición STMM preferido por el usuario. Para solucionarlo, espere a que se actualice CURRENT_NUMBER de forma asíncrona o detenga e inicie la base de datos para forzar la actualización de CURRENT_NUMBER.

Información de conjunto de resultados

El estado de la ejecución del mandato se devuelve en el área de comunicaciones SQL (SQLCA) obtenida a partir de la sentencia CALL. Si la ejecución es correcta, el mandato devuelve información adicional en el conjunto de resultados siguiente:

Tabla 41. Conjunto de resultados devuelto por el mandato GET STMM TUNING DBPARTITIONNUM

| Nombre de columna | Tipo de datos | Descripción |
|-----------------------|---------------|--|
| USER_PREFERRED_NUMBER | INTEGER | Número de partición de base de datos de ajuste STMM (Self Tuning Memory Manager) preferido por el usuario. Un valor de -1 indica que se utiliza la partición de base de datos por omisión. |
| CURRENT_NUMBER | INTEGER | Número de partición de base de datos de ajuste STM actual. Un valor de -1 indica que se utiliza la partición de base de datos por omisión. |

Mandato IMPORT utilizando el procedimiento ADMIN_CMD

Inserta datos de un archivo externo con un formato de archivo soportado en una tabla, una jerarquía, vista o apodo. LOAD es una alternativa más rápida, pero el programa de utilidad de carga no da soporte a la carga de datos en el nivel de la jerarquía.

Enlace rápido con “Modificadores de tipo de archivo para el programa de utilidad de importación” en la página 102.

Autorización

- IMPORT utilizando la opción **INSERT** requiere una de las autorizaciones siguientes:
 - Autorización *dataaccess*
 - Privilegio CONTROL en cada tabla, vista o apodo participante
 - Privilegio INSERT y SELECT en cada tabla o vista participante
- IMPORT en una tabla existente utilizando la opción **INSERT_UPDATE**, necesita una de las autorizaciones siguientes:
 - Autorización *dataaccess*
 - Privilegio CONTROL en cada tabla, vista o apodo participante
 - Privilegio INSERT, SELECT, UPDATE y DELETE en cada tabla o vista participante
- IMPORT en una tabla existente utilizando la opción **REPLACE** o **REPLACE_CREATE**, requiere una de las autorizaciones siguientes:
 - Autorización *dataaccess*
 - Privilegio CONTROL sobre la tabla o vista

- Privilegio INSERT, SELECT y DELETE sobre la tabla o vista
- IMPORT en una tabla nueva utilizando la opción **CREATE** o **REPLACE_CREATE**, requiere una de las autorizaciones siguientes:
 - Autorización *dbadm*
 - Autorización CREATETAB para la base de datos y privilegio USE para el espacio de tablas, y también uno de los elementos siguientes:
 - Autorización IMPLICIT_SCHEMA para la base de datos, si el nombre de esquema implícito o explícito de la tabla no existe
 - Privilegio CREATEIN sobre el esquema, si el nombre esquema de la tabla hace referencia a un esquema existente.
- IMPORT en una jerarquía que no existe utilizando la opción **CREATE**, o la opción **REPLACE_CREATE**, requiere una de las autorizaciones siguientes:
 - Autorización *dbadm*
 - Autorización CREATETAB para la base de datos y privilegio USE para el espacio de tablas, y también uno de los elementos siguientes:
 - Autorización IMPLICIT_SCHEMA sobre la base de datos, si no existe el nombre de esquema de la tabla
 - Privilegio CREATEIN sobre el esquema, si existe el esquema de la tabla
 - Privilegio CONTROL en cada subtabla de la jerarquía, si se utiliza la opción **REPLACE_CREATE** en la jerarquía entera
- IMPORT en una jerarquía existente utilizando la opción **REPLACE** requiere una de las autorizaciones siguientes:
 - Autorización *dataaccess*
 - Privilegio CONTROL en cada subtabla de la jerarquía
- Para importar datos a una tabla que tenga columnas protegidas, el ID de autorización de sesión debe tener credenciales LBAC que permitan el acceso de grabación a todas las columnas protegidas de la tabla. En caso contrario, la importación falla y se devuelve un error (SQLSTATE 42512).
- Para importar datos a una tabla que tiene filas protegidas, el ID de autorización de sesión debe tener credenciales LBAC que satisfagan estos criterios:
 - Formar parte de la política de seguridad que protege la tabla
 - Haber sido otorgadas al ID de autorización de sesión para acceso de grabación

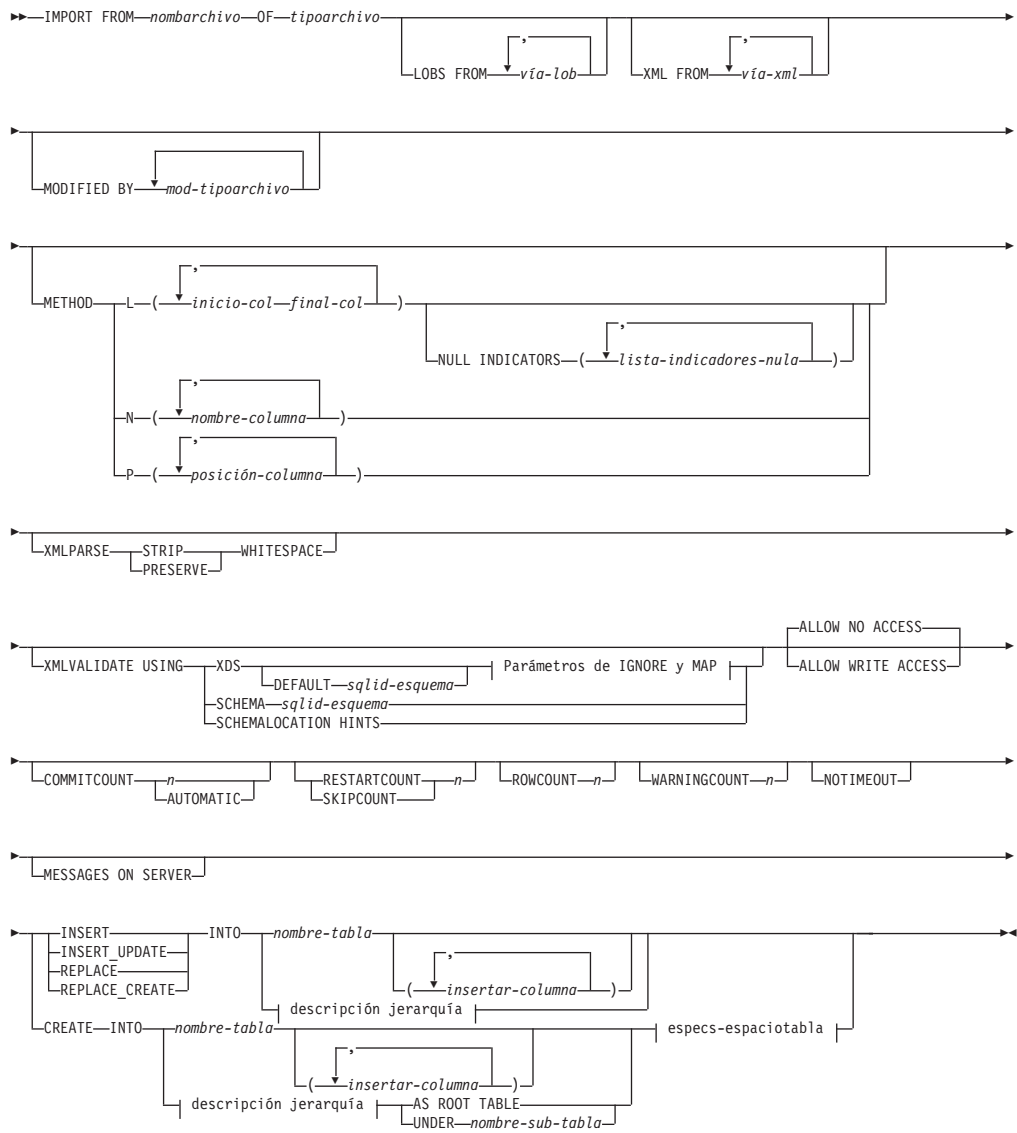
La etiqueta de la fila que se debe insertar, las credenciales LBAC del usuario, la definición de política de seguridad y las normas de LBAC determinan la etiqueta en la fila.

- Si se especifica la opción **REPLACE** o **REPLACE_CREATE**, el ID de autorización de la sesión debe tener la autorización para descartar la tabla.
- Para importar datos en un apodo, el ID de autorización de sesión debe tener el privilegio para acceder y utilizar una fuente de datos especificada en la modalidad de paso a través.

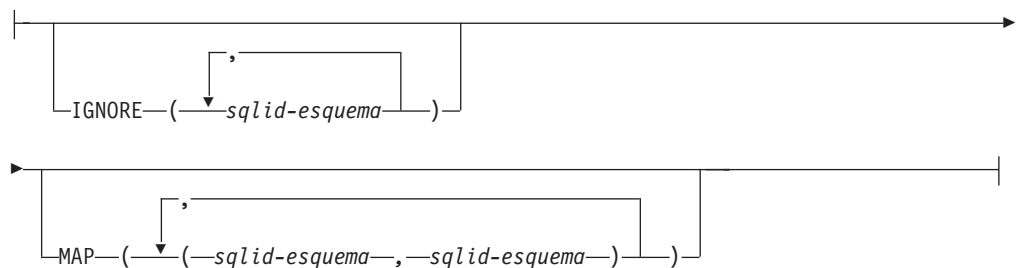
Conexión necesaria

Base de datos. El acceso del programa de utilidad a servidores de bases de datos Linux, UNIX o Windows desde clientes Linux, UNIX o Windows debe ser una conexión directa a través del motor y no a través de un entorno de pasarela o de bucle de retorno de DB2 Connect.

Sintaxis del mandato



Parámetros de IGNORE y de MAP:



parámetro es necesario cuando se importa a un apodo y hay que especificar **COMMITCOUNT** con un número válido (**AUTOMATIC** no se considera una opción válida).

AS ROOT TABLE

Crea una o más subtablas como jerarquía de tablas autónoma.

COMMITCOUNT *n* | AUTOMATIC

Realiza una operación **COMMIT** tras la importación de cada *n* registros. Cuando se especifica un número *n*, realiza una operación **COMMIT** tras la importación de cada *n* registros. Cuando se utilizan inserciones compuestas, se redondea al alza una frecuencia de confirmación especificada por el usuario de *n* hasta el primer entero múltiplo del valor de cuenta compuesto. Cuando se especifica **AUTOMATIC**, la importación determina internamente cuándo se debe realizar una confirmación. El programa de utilidad realizará una confirmación por uno de los dos motivos siguientes:

- para evitar el agotamiento del espacio de anotación cronológica activa
- para evitar escalas de bloqueo del nivel de fila al nivel de tabla

Si se especifica la opción **ALLOW WRITE ACCESS** y no se especifica la opción **COMMITCOUNT**, el programa de utilidad de importación realizará confirmaciones como si se hubiera especificado **COMMITCOUNT AUTOMATIC**.

La capacidad de la operación de importación de evitar que se agote el espacio de anotaciones cronológicas activas mediante la variable de registro de DB2 **DB2_FORCE_APP_ON_MAX_LOG**:

- Si **DB2_FORCE_APP_ON_MAX_LOG** está establecido en **FALSE** y se especifica la opción del mandato **COMMITCOUNT AUTOMATIC**, el programa de utilidad de importación podrá evitar automáticamente que se agote el espacio de anotaciones cronológicas activas.
- Si **DB2_FORCE_APP_ON_MAX_LOG** está establecido en **FALSE** y se especifica la opción del mandato **COMMITCOUNT *n***, el programa de utilidad de importación intentará resolver la condición de anotaciones cronológicas llenas si encuentra un mensaje **SQL0964C** (Anotaciones cronológicas llenas) al insertar o actualizar un registro. Realizará una confirmación no condicional y a continuación volverá a intentar insertar o actualizar el registro. Si esto no resuelve el problema (por ejemplo, cuando el archivo de anotaciones lleno se atribuyese a otra actividad de la base de datos), el mandato **IMPORT** fallará, como cabe esperar, pero el número de filas confirmadas podría no ser un múltiplo del valor **COMMITCOUNT *n***. Para evitar procesar las filas que ya estaban confirmadas al volver a intentar la operación, utilice los parámetros del mandato **RESTARTCOUNT** o **SKIPCOUNT**.
- Si **DB2_FORCE_APP_ON_MAX_LOG** está establecido en **TRUE** (que es el valor por omisión), la operación de importación fallará si encuentra un **SQL0964C** al insertar o actualizar un registro. Esto se puede producir independientemente de si especifica **COMMITCOUNT AUTOMATIC** o **COMMITCOUNT *n***.

La aplicación se fuerza fuera de la base de datos y la unidad de trabajo actual se retrotrae. Para evitar procesar las filas que ya estaban confirmadas al volver a intentar la operación, utilice los parámetros del mandato **RESTARTCOUNT** o **SKIPCOUNT**.

CREATE

Nota: El parámetro **CREATE** ha quedado obsoleto y se puede eliminar en un futuro release. Para obtener información detallada, consulte “Las opciones del mandato **IMPORT**, **CREATE** y **REPLACE_CREATE** están en desuso”.

Crea la definición de la tabla y el contenido de las filas en la página de códigos de la base de datos. Si los datos se han exportado de una tabla, subtabla o jerarquía de DB2 se crean índices. Si esta opción opera en una jerarquía y los datos se han exportado desde DB2, también se creará una jerarquía de tipos. Esta opción solo se puede utilizar con archivos IXF.

Este parámetro no es válido cuando se importa a un apodo.

Nota: Si los datos se han exportado desde una base de datos de sistema principal MVS y contienen campos **LONGVAR** cuyas longitudes, calculadas en el tamaño de página son superiores a 254, **CREATE** podría fallar porque las filas son demasiado largas. Consulte “Volver a crear una tabla importada” para obtener una lista de restricciones. En este caso, se deberá crear manualmente la tabla y se deberá invocar **IMPORT** con **INSERT** o, alternativamente, se deberá utilizar el mandato **LOAD**.

DEFAULT *sqlid-esquema*

Esta opción sólo se puede utilizar cuando se especifica el parámetro **USING XDS**. El esquema especificado mediante la cláusula **DEFAULT** identifica un esquema que hay que utilizar para la validación cuando el Especificador de datos XML (XDS) de un documento XML importado no contiene un atributo **SCH** que identifique un esquema XML.

La cláusula **DEFAULT** tiene prioridad sobre las cláusulas **IGNORE** y **MAP**. Si un XDS satisface la cláusula **DEFAULT**, se omitirán las especificaciones **IGNORE** y **MAP**.

FROM *nombrarchivo*

Especifica el nombre del archivo que contiene los datos que se deben importar. Debe ser una vía de acceso totalmente calificada, y el archivo ya debe existir en el servidor de bases de datos.

HIERARCHY

Especifica que deben importarse datos jerárquicos.

IGNORE *sqlid-esquema*

Esta opción sólo se puede utilizar cuando se especifica el parámetro **USING XDS**. La cláusula **IGNORE** especifica una lista de uno o más esquemas que hay que ignorar si están identificados mediante un atributo **SCH**. Si un atributo **SCH** existe en el Especificador de datos XML para un documento XML importado, y el esquema identificado por el atributo **SCH** se incluye en la lista de esquemas que se deben ignorar, no se producirá ninguna validación de esquemas para el documento XML importado.

Si se especifica un esquema en la cláusula **IGNORE**, tampoco puede estar presente en el lado izquierdo de un par de esquemas en la cláusula **MAP**.

La cláusula **IGNORE** sólo se aplica al XDS. Un esquema que está correlacionado por la cláusula **MAP** no se omitirá posteriormente si se ha especificado en la cláusula **IGNORE**.

IN *nombre-espacio-tablas*

Identifica el espacio de tablas en el que se va a crear la tabla. El espacio de tablas debe existir y debe ser un espacio de tablas **REGULAR**. Si no se especifica ningún otro espacio de tablas, todas las partes de tabla se almacenan en este espacio de tablas. Si no se especifica esta cláusula, la

tabla se crea en un espacio de tablas creado por el ID de autorización. Si no se encuentra ninguno, se coloca la tabla en el espacio de tablas por omisión USERSPACE1. Si se ha descartado USERSPACE1, fallará la creación de tabla.

INDEX IN *nombre-espacio-tabla*

Identifica el espacio de tablas donde se crearán todos los índices de la tabla. Esta opción sólo está permitida cuando el espacio de tablas primario que se ha especificado en la cláusula **IN** es un espacio de tablas DMS. El espacio de tablas especificado debe existir y debe ser un espacio de tablas REGULAR o LARGE DMS.

Nota: La especificación de qué espacio de tablas contendrá un índice solo se puede utilizar al crear la tabla.

columna-inserción

Especifica el nombre de una columna de la tabla o la vista en la que deben insertarse datos.

INSERT

Añade los datos importados a la tabla sin cambiar los datos de tabla existentes.

INSERT_UPDATE

Añade filas de datos importados a la tabla de destino o actualiza las filas existentes (de la tabla de destino) con las claves primarias que coinciden.

INTO *nombre_tabla*

Especifica la tabla de base de datos a la que se deberán importar los datos. Esta tabla no puede ser una tabla de sistema, una tabla temporal creada, una tabla temporal declarada o una tabla de resumen.

Uno puede utilizar un alias para **INSERT**, **INSERT_UPDATE**, o **REPLACE**, excepto en el caso de un servidor de nivel inferior, en que debe utilizar un nombre de tabla totalmente calificado o no calificado. Un nombre de tabla calificado tiene el formato *esquema.nombretabla*. El *esquema* es el nombre de usuario bajo el que se ha creado la tabla.

LOBS FROM *vía-lob*

Especifica una o más vías de acceso totalmente calificado que almacenan archivos LOB. Las vías de acceso deben existir en la partición de coordinador del servidor de base de datos. Los nombres de los archivos de datos LOB se almacenan en el archivo de datos principal (ASC, DEL o IXF), en la columna que se cargará en la columna LOB. El número máximo de vías de acceso que se puede especificar es 999. Esto activará implícitamente el comportamiento de LOBSINFILE.

Este parámetro no es válido cuando se importa a un apodo.

LONG IN *nombre-espacio-tablas*

Identifica el espacio de tablas en el que se almacenarán los valores de todas las columnas largas (tipos de datos LONG VARCHAR, LONG VARGRAPHIC, LOB o los tipos diferenciados con alguno de estos como tipos de fuente). Esta opción sólo está permitida si el espacio de tablas primarias especificado en la cláusula **IN** es un espacio de tablas DMS. El espacio de tablas debe existir y debe ser un espacio de tablas LARGE DMS.

MAP *sqlid-esquema*

Esta opción sólo se puede utilizar cuando se especifica el parámetro **USING XDS**. Utilice la cláusula **MAP** para especificar esquemas alternativos que deben utilizarse en lugar de los especificados por el

atributo SCH de un especificador de datos XML (XDS) para cada documento XML importado. La cláusula **MAP** especifica una lista de uno o más pares de esquemas, donde cada par representa una correlación de un esquema con otro. El primer esquema del par representa un esquema al que hace referencia un atributo SCH de un XDS. El segundo esquema del par representa el esquema que debe utilizarse para llevar a cabo la validación de esquemas.

Si un esquema está presente en el lado izquierdo de un par de esquemas en la cláusula **MAP**, tampoco puede especificarse en la cláusula **IGNORE**.

Una vez aplicada una correlación de par de esquemas, el resultado es final. La operación de correlación no es transitiva y, por lo tanto, el esquema elegido no se aplicará ulteriormente a otra correlación de par de esquemas.

Un esquema no se puede correlacionar más de una vez, es decir, no puede aparecer en la parte izquierda de más de un par.

MESSAGES ON SERVER

Especifica que hay que guardar el archivo de mensajes creado en el servidor mediante el mandato **IMPORT**. El conjunto de resultados devuelto incluirá las dos columnas siguientes: **MSG_RETRIEVAL**, que es la sentencia de SQL necesaria para recuperar todos los mensajes de aviso y de error producidos durante esta operación, y **MSG_REMOVAL**, que es la sentencia de SQL necesaria para hacer limpieza de los mensajes.

Si no se especifica esta cláusula, el archivo de mensajes se suprimirá cuando el procedimiento **ADMIN_CMD** retorne al llamador. La columna **MSG_RETRIEVAL** y la columna **MSG_REMOVAL** del conjunto de resultados contendrán valores nulos.

Observe que, con cláusula o sin ella, el ID de usuario delimitado debe poseer autorización para crear archivos bajo el directorio indicado por la variable de registro **DB2_UTIL_MSGPATH**, así como bajo el directorio al que hay que exportar los datos.

METHOD

L Especifica los números de columna inicial y final de los que se deben importar datos. Un número de columna es un desplazamiento de bytes respecto al principio de una fila de datos. Se numera empezando por 1.

Nota: Este método solo se puede utilizar con archivos **ASC** y es la única opción válida para ese tipo de archivo.

N Especifica los nombres de las columnas del archivo de datos que se deben importar. Las mayúsculas y minúsculas de estos nombres de columna deben coincidir con las mayúsculas y minúsculas de los nombres correspondientes en los catálogos del sistema. Cada columna de tabla que no sea anulable debe tener una entrada correspondiente en la lista **METHOD N**. Por ejemplo, dados los campos de datos **F1**, **F2**, **F3**, **F4**, **F5** y **F6** y las columnas de tabla **C1 INT**, **C2 INT NOT NULL**, **C3 INT NOT NULL** y **C4 INT**, el método **N (F2, F1, F4, F3)** es una petición válida, mientras que el método **N (F2, F1)** no es válido.

Nota: Este método solo se puede utilizar con archivos **IXF**.

P Especifica los números de campo de los campos de datos de entrada que se van a importar.

Nota: Este método solo se puede utilizar con archivos IXF o DEL y es la única opción válida para el tipo de archivo DEL.

MODIFIED BY *mod-tipoarchivo*

Especifica opciones de modificador de tipo de archivo. Consulte "Modificadores de tipo de archivo para el programa de utilidad de importación" en la página 102.

NOTIMEOUT

Especifica que el programa de utilidad de importación no superará el tiempo de espera mientras espere bloqueos. Esta opción sustituye al parámetro de configuración de base de datos **locktimeout**. No se verá afectada ninguna otra aplicación.

NULL INDICATORS *lista-indicadores-nulo*

Esta opción sólo se puede utilizar cuando se especifica el parámetro **METHOD L**. Es decir, el archivo de entrada es un archivo ASC. Una lista de indicadores nulos es una lista de enteros positivos separados por comas que especifican el número de columna de cada campo de indicador nulo. El número de columna es el desplazamiento de bytes del campo de indicador nulo respecto al principio de una fila de datos. Debe haber una entrada en la lista de indicadores nulos para cada campo de datos definido en el parámetro **METHOD L**. Un número de columna de cero indica que el campo de datos correspondiente siempre contiene datos.

Un valor de Y en la columna de indicador NULL especifica que los datos de la columna son NULL. Cualquier carácter *distinto de Y* en la columna del indicador NULL especifica que los datos de columna no son NULL y se cargarán los datos de columna especificados por la opción **METHOD L**.

El carácter de indicador NULL se puede cambiar utilizando la opción **MODIFIED BY**, con el modificador de tipo de archivo **nullindchar**.

OF *tipoarchivo*

Especifica el formato de los datos del archivo de entrada:

- ASC (formato ASCII no delimitado)
- DEL (formato ASCII delimitado) utilizado por diversos programas del gestor de bases de datos y gestores de archivos.
- WSF (formato de hoja de trabajo), utilizado por programas tales como:
 - Lotus 1-2-3
 - Lotus Symphony
- IXF (Integration Exchange Format, versión PC) es un formato binario que utiliza exclusivamente DB2.

Importante: El soporte para el formato de archivo WSF ha quedado en desuso y se puede eliminar en futuros releases. Se recomienda empezar a usar un formato de archivo compatible en lugar de archivos WSF antes de que se elimine el soporte.

El tipo de archivo WSF no está soportado cuando se importa a un apodo.

REPLACE

Suprime todos los datos existentes de la tabla truncando el objeto de datos e inserta los datos importados. La definición de tabla y las definiciones de índice no se modifican. Esta opción solo se puede utilizar si existe la tabla. Si se utiliza esta opción al mover datos entre jerarquías, sólo se pueden sustituir los datos de una jerarquía entera, no de subtablas individuales.

Este parámetro no es válido cuando se importa a un apodo.

Esta opción no respeta la cláusula NOT LOGGED INITIALLY (NLI) de la sentencia CREATE TABLE, ni la cláusula ACTIVE NOT LOGGED INITIALLY de la sentencia ALTER TABLE.

Si se realiza una importación con la opción **REPLACE** dentro de la misma transacción que una sentencia CREATE TABLE o ALTER TABLE en la que se invoque la cláusula NLI, la importación no respetará la cláusula NLI. Todas las inserciones se anotarán.

Corrección 1

Suprima el contenido de la tabla con la sentencia DELETE y luego invoque la importación con la sentencia INSERT

Corrección 2

Descarte la tabla, vuelva a crearla y luego invoque la importación con la sentencia INSERT.

Esta limitación se aplica a DB2 Universal Database Versión 7 y DB2 UDB Versión 8

REPLACE_CREATE

Nota: El parámetro **REPLACE_CREATE** se ha quedado obsoleto y se puede eliminar en un futuro release. Para obtener información detallada, consulte “Las opciones del mandato IMPORT, CREATE y REPLACE_CREATE, están en desuso”.

Si la tabla existe, suprime todos los datos existentes de la tabla truncando el objeto de datos e inserta los datos importados sin cambiar la definición de tabla o las definiciones de índice.

Si la tabla no existe, crea las definiciones de tabla y de índice, así como el contenido de filas, en la página de códigos de la base de datos. Consulte *Volver a crear una tabla importada* para obtener una lista de restricciones.

Esta opción solo se puede utilizar con archivos IXF. Si se utiliza esta opción al mover datos entre jerarquías, sólo se pueden sustituir los datos de una jerarquía entera, no de subtablas individuales.

Este parámetro no es válido cuando se importa a un apodo.

RESTARTCOUNT *n*

Especifica que se debe iniciar una operación de importación en el registro $n + 1$. Los primeros n registros se saltan. Esta opción es funcionalmente equivalente a **SKIPCOUNT**. **RESTARTCOUNT** y **SKIPCOUNT** se excluyen mutuamente.

ROWCOUNT *n*

Especifica el número n de registros físicos del archivo que se deben importar (insertar o actualizar). Permite a un usuario importar únicamente n filas de un archivo, empezando por el registro determinado por las opciones **SKIPCOUNT** o **RESTARTCOUNT**. Si no se especifican las opciones **SKIPCOUNT** o **RESTARTCOUNT**, se importan las primeras n filas. Si se especifica **SKIPCOUNT *m*** o **RESTARTCOUNT *m***, se importan las filas $m+1$ a $m+n$. Cuando se utilizan inserciones compuestas, se redondea al alza el número de filas **ROWCOUNT *n*** hasta el primer múltiplo entero del valor de cuenta compuesto.

SKIPCOUNT *n*

Especifica que se debe iniciar una operación de importación en el registro n

+ 1. Los primeros *n* registros se saltan. Esta opción es funcionalmente equivalente a **RESTARTCOUNT**. **SKIPCOUNT** y **RESTARTCOUNT** se excluyen mutuamente.

STARTING *nombre-sub-tabla*

Palabra clave para jerarquía solamente, que solicita el orden por omisión, empezando a partir de *nombre-subtabla*. Para los archivos PC/IXF, el orden por omisión es el orden almacenado en el archivo de entrada. El orden por omisión es el único orden válido para el formato de archivos PC/IXF.

lista-subtablas

Para tablas de tipo con la opción **INSERT** o **INSERT_UPDATE**, se utiliza una lista de nombres de subtabla para indicar las subtablas en las que se importarán los datos.

lista-orden-transversal

Para tablas de tipo con la opción **INSERT**, **INSERT_UPDATE**, o **REPLACE**, se utiliza una lista de nombres de subtablas para indicar el orden transversal de las subtablas de importación de la jerarquía.

UNDER *nombre-sub-tabla*

Especifica una tabla padre para crear una o más subtablas.

WARNINGCOUNT *n*

Detiene la operación de importación después de *n* avisos. Establezca este parámetro si no se esperan avisos, pero se desea que se verifique si se están utilizando el archivo y la tabla correctos. Si el archivo de importación o la tabla de destino se especifican de modo incorrecto, el programa de utilidad de importación generará un aviso por cada fila que intente importar, lo que hará que la importación falle. Si *n* es cero o no se especifica esta opción, la operación de importación continuará independientemente del número de avisos emitidos.

XML FROM *vía-xml*

Especifica una o más vías de acceso que contienen los archivos XML.

XMLPARSE

Especifica cómo se analizan los documentos XML. Si no se especifica esta opción, el comportamiento de análisis de los documentos XML vendrá determinado por el valor del registro especial **CURRENT XMLPARSE OPTION**.

STRIP WHITESPACE

Especifica que hay que eliminar el espacio en blanco al analizar el documento XML.

PRESERVE WHITESPACE

Especifica que no hay que eliminar el espacio en blanco al analizar el documento XML.

XMLVALIDATE

Especifica que los documentos XML se validan con respecto a un esquema, cuando sea pertinente.

USING XDS

Los documentos XML se validan con respecto al esquema XML identificado por el especificador de datos XML (XDS) en el documento de datos principal. Por omisión, si se invoca la opción **XMLVALIDATE** con la cláusula **USING XDS**, el esquema que se utiliza para realizar la validación estará determinado por el atributo SCH del XDS. Si no existe un atributo SCH en el XDS, no

se producirá ningún esquema de validación a menos que la cláusula **DEFAULT** especifique un esquema por omisión.

Las cláusulas **DEFAULT**, **IGNORE** y **MAP** se pueden utilizar para modificar el comportamiento de determinación del esquema. Estas tres cláusulas opcionales se aplican directamente a las especificaciones del XDS, no entre ellas. Por ejemplo, si se selecciona un esquema porque está especificado por la cláusula **DEFAULT**, no se omitirá si también está especificado por la cláusula **IGNORE**. Asimismo, si se selecciona un esquema como la primera parte de un par en la cláusula **MAP**, no se volverá a correlacionar si también se ha especificado en la segunda parte de otro par de cláusulas **MAP**.

USING SCHEMA *sqlid-esquema*

Los documentos XML se validan con respecto al esquema XML que tiene el identificador SQL especificado. En este caso, el atributo SCH del especificador de datos XML (XDS) se ignorará para todas las columnas XML.

USING SCHEMALOCATION HINTS

Los documentos XML se validan con respecto a los esquemas identificados por las sugerencias de ubicación de esquemas XML en los documentos XML fuente. Si no se encuentra un atributo schemaLocation en el documento XML, no se producirá ninguna validación. Cuando se especifica la cláusula **USING SCHEMALOCATION HINTS**, el atributo SCH del Especificador de datos XML (XDS) se omitirá para todas las columnas XML.

Vea los ejemplos de la opción **XMLVALIDATE** que figuran más abajo.

Ejemplo

En el siguiente ejemplo se muestra cómo importar información del archivo myfile.ixf a la tabla STAFF de la base de datos SAMPLE.

```
CALL SYSPROC.ADMIN_CMD
  ('IMPORT FROM /home/userid/data/myfile.ixf
  OF IXF MESSAGES ON SERVER INSERT INTO STAFF')
```

Notas de uso

Las vías de acceso que se utilizan en el mandato **IMPORT** deben ser vías de acceso totalmente calificadas válidas en el nodo coordinador del servidor.

Si se especifican las opciones **ALLOW WRITE ACCESS** o **COMMITCOUNT**, se realizará una confirmación por parte del programa de utilidad de importación. Esto hace que el procedimiento **ADMIN_CMD** devuelva un error **SQL30090N** con el código de razón 1 en el caso de las conexiones de tipo 2.

Si el valor que hay que asignar a una columna de un conjunto de resultados del procedimiento **ADMIN_CMD** es mayor que el valor máximo del tipo de datos de la columna, se asigna el valor máximo del tipo de datos y se devuelve el mensaje de aviso **SQL1155W**.

Asegúrese de completar todas las operaciones de tabla y de liberar todos los bloqueos antes de iniciar una operación de importación. Esto puede realizarse emitiendo un mandato **COMMIT** después de cerrar todos los cursores abiertos **WITH HOLD** o emitiendo un mandato **ROLLBACK**.

El programa de utilidad de importación añade filas a la tabla de destino utilizando la sentencia INSERT de SQL. El programa de utilidad emite una sentencia INSERT para cada fila de datos del archivo de entrada. Si falla una sentencia INSERT, se produce una de dos acciones:

- Si es probable que las sentencias INSERT subsiguientes puedan ser satisfactorias, se graba un mensaje de aviso en el archivo de mensajes y el proceso continúa.
- Si es probable que las sentencias INSERT subsiguientes fallen y existen posibilidades de que se dañe la base de datos, se graba un mensaje de error en el archivo de mensajes y el proceso se detiene.

El programa de utilidad realiza una operación COMMIT automática después de que se supriman las filas antiguas durante una operación **REPLACE** o **REPLACE_CREATE**. Por consiguiente, si el sistema falla o la aplicación interrumpe el gestor de bases de datos después de que se trunque el objeto de tabla, se perderán todos los datos anteriores. Antes de utilizar estas opciones, asegúrese de que los datos anteriores ya no son necesarios.

Si la anotación cronológica se llena durante una operación **CREATE**, **REPLACE**, o **REPLACE_CREATE**, el programa de utilidad realiza una operación COMMIT automática en los registros insertados. Si el sistema falla o la aplicación interrumpe el gestor de bases de datos después de un COMMIT automático, permanece en la base de datos una tabla con datos parciales. Utilice la opción **REPLACE** o **REPLACE_CREATE** para volver a ejecutar toda la operación de importación, o bien utilice **INSERT** con el parámetro **RESTARTCOUNT** establecido en el número de filas importadas satisfactoriamente.

Por omisión, las operaciones COMMIT automáticas no se realizan para la opción **INSERT** o **INSERT_UPDATE**. No obstante, se realizan si el parámetro **COMMITCOUNT** es distinto de cero. Si no se realizan operaciones COMMIT automáticas, una anotación cronológica llena producirá una retrotracción (ROLLBACK).

La importación fuera de línea no realiza operaciones COMMIT automáticas si se da alguna de las condiciones siguientes:

- el destino es una vista, no una tabla
- se utilizan inserciones compuestas
- se utilizan inserciones en almacenamiento intermedio

Por omisión, la importación en línea realiza operaciones COMMIT automáticas para liberar tanto el espacio de anotación cronológica activa como la lista de bloqueos. No se realizan operaciones COMMIT automáticas, únicamente si se especifica un valor de cero para **COMMITCOUNT**.

Siempre que el programa de utilidad de importación realiza un COMMIT, se graban dos mensajes en el archivo de mensajes: uno indica el número de registros que se deben confirmar y el otro se graba después de una operación COMMIT satisfactoria. Cuando reinicie la operación de importación después de una anomalía, especifique el número de registros que se deben saltar, según se determine en el último COMMIT satisfactorio.

El programa de utilidad de importación acepta datos de entrada con problemas de incompatibilidad menores (por ejemplo, se pueden importar datos de tipo carácter utilizando relleno o truncamiento y se pueden importar datos numéricos con un tipo de datos numéricos diferente), pero no se aceptan datos con problemas de incompatibilidad más importantes.

No se puede utilizar **REPLACE** o **REPLACE_CREATE** con una tabla de objetos si ésta tiene otros dependientes distintos de ella misma o con una vista de objeto si la tabla base tiene dependientes (incluida ella misma). Para sustituir una tabla o una vista de este tipo, realice lo siguiente:

1. Descarte todas las claves foráneas en las que la tabla es padre.
2. Ejecute el programa de utilidad de importación.
3. Modifique la tabla para volver a crear las claves foráneas.

Si se produce un error al volver a crear las claves foráneas, modifique los datos para mantener la integridad referencial.

Las restricciones de referencia y las definiciones de claves foráneas no se conservan al volver a crear tablas a partir de archivos PC/IX. (Las definiciones de claves primarias *se conservan* si se han exportado anteriormente los datos utilizando **SELECT ***.)

La importación a una base de datos remota requiere suficiente espacio de disco en el servidor para una copia del archivo de datos de entrada, para el archivo de mensajes de salida y para el aumento potencial de tamaño de la base de datos.

Si se ejecuta una operación de importación en una base de datos remota, y el archivo de mensajes de salida es muy grande (más de 60 KB), puede que el archivo de mensajes devuelto al usuario en el cliente no incluya mensajes que se hayan producido durante la operación de importación. Los primeros y los últimos 30 KB de información de mensajes siempre se retienen.

La importación de archivos PC/IXF a una base de datos remota es mucho más rápida si el archivo PC/IXF está en una unidad de disco duro en lugar de estar en disquetes.

Antes de poder importar datos en los formatos de archivo **ASC**, **DEL** o **WSF**, debe existir la tabla o la jerarquía de base de dato; no obstante si la tabla no existe, las opciones **IMPORT CREATE** o **IMPORT REPLACE_CREATE** crean la tabla al importar datos desde un archivo PC/IXF. Para las tablas de tipo, **IMPORT CREATE** puede crear también la jerarquía de tipos y la jerarquía de tablas.

La importación PC/IXF deberá utilizarse para mover datos (incluidos datos jerárquicos) entre bases de datos. Si los datos de tipo carácter que contienen separadores de filas se exportan a un archivo ASCII delimitado (**DEL**) y se procesan con un programa de transferencia de texto, los campos que contengan separadores de filas se acortarán o se ampliarán. El paso de copia de archivo no es necesario si se puede acceder a las bases de datos fuente y destino desde el mismo cliente.

Se supone que los datos de los archivos **ASC** y **DEL** están en la página de códigos de la aplicación cliente que realiza la importación. Se recomiendan los archivos PC/IXF, que permiten diferentes páginas de códigos, al importar datos de páginas de códigos diferentes. Si el archivo PC/IXF y el programa de utilidad de importación están en la misma página de códigos, el proceso se produce como para una aplicación normal. Si las dos páginas son diferentes y se especifica la opción **FORCEIN**, el programa de utilidad de importación presupone que los datos del archivo PC/IXF tienen la misma página de códigos que la aplicación que realiza la importación. Esto se produce incluso si existe una tabla de conversión para las dos páginas de códigos. Si las dos son diferentes, no se especifica la opción **FORCEIN** y hay una tabla de conversión, todos los datos del archivo

PC/IXF se convertirán de la página de códigos del archivo a la página de códigos de la aplicación. Si las dos son diferentes, no se especifica la opción **FORCEIN** y no hay ninguna tabla de conversión, la operación de importación fallará. Esto se aplica únicamente a archivos PC/IXF en clientes de DB2 en el sistema operativo AIX.

Para objetos de tabla en una página de 8 KG que estén cerca del límite de 1012 columnas, la importación de archivos de datos PC/IXF podría hacer que DB2 devuelva un error porque se ha excedido el tamaño máximo de una sentencia de SQL. Esta situación sólo se puede producir si las columnas son de tipo CHAR, VARCHAR o CLOB. La restricción no se aplica a la importación de los archivos **DEL** o **ASC**. Si se están utilizando archivos PC/IXF para crear una nueva tabla, una alternativa consiste en utilizar db2look para volcar la sentencia DDL que creó la tabla y, a continuación, emitir esa sentencia a través del CLP.

Se puede utilizar DB2 Connect para importar datos a servidores DRDA como por ejemplo, DB2 para OS/390, DB2 para VM y VSE, y DB2 para OS/400. Sólo está soportada la importación PC/IXF (opción **INSERT**). El parámetro **RESTARTCOUNT** también está soportado, pero no así el parámetro **COMMITCOUNT**.

Cuando utilice la opción **CREATE** con tablas de tipos, cree cada subtabla definida en el archivo PC/IXF; las definiciones de subtabla no se pueden modificar. Cuando utilice opciones distintas de **CREATE** con tablas de tipo, la lista de orden transversal permite especificar el orden transversal; por lo tanto, la lista de orden transversal debe coincidir con la utilizada durante la operación de exportación. Para el formato de archivo PC/IXF, solo es necesario especificar el nombre de subtabla de destino y utilizar el orden transversal almacenado en el archivo.

El programa de utilidad de importación se puede utilizar para recuperar una tabla exportada anteriormente a un archivo PC/IXF. La tabla vuelve al estado en el que estaba al exportarse.

Los datos no se pueden importar a una tabla de sistema, una tabla temporal creada, una tabla temporal declarada o una tabla de resumen.

No se pueden crear vistas mediante el programa de utilidad de importación.

La importación de un archivo PC/IX con varias partes cuyas partes individuales se copian desde un sistema Windows hasta un sistema AIX está soportada. Sólo debe especificarse el nombre del primer archivo en el mandato **IMPORT**. Por ejemplo, **IMPORT FROM data.ixf OF IXF INSERT INTO TABLE1**. El archivo **data.002**, etc. debe estar disponible en el mismo directorio que **data.ixf**.

En el sistema operativo Windows:

- No se soporta la importación de archivos PC/IXF subdivididos de forma lógica.
- No se soporta la importación de archivos PC/IXF o WSF de formato incorrecto.

En las etiquetas de seguridad, cuando están en formato interno, puede haber caracteres de nueva línea. Si importa el archivo utilizando el formato de archivo **DEL**, los caracteres de nueva línea se podrían malinterpretar como delimitadores. Si surge este problema, utilice la prioridad por omisión más antigua para los delimitadores especificando el modificador de tipo de archivo **delprioritychar** en el mandato **IMPORT**.

Consideraciones federadas

Cuando se utiliza el mandato `IMPORT` y los parámetros de mandato `INSERT`, `UPDATE` o `INSERT_UPDATE`, debe asegurarse de que tiene privilegio `CONGROL` en el apodo participante. Debe asegurarse de que el apodo que desea utiliza cuando realiza una operación de importación ya exista. Hay varias restricciones que debería tener en cuenta tal y como se muestra en la sección de parámetros del mandato `IMPORT`.

Algunas fuentes de datos, como `ODBC`, no dan soporte a la importación a apodos.

Información de conjunto de resultados

El estado de la ejecución del mandato se devuelve en el área de comunicaciones SQL (`SQLCA`) obtenida a partir de la sentencia `CALL`. Si la ejecución es satisfactoria, el mandato devuelve información adicional en conjuntos de resultados, de la siguiente manera:

Tabla 42. Conjunto de resultados devuelto por el mandato `IMPORT`

| Nombre de columna | Tipo de datos | Descripción |
|-----------------------------|---------------------------|---|
| <code>ROWS_READ</code> | <code>BIGINT</code> | Número de registros leídos en el archivo durante la importación. |
| <code>ROWS_SKIPPED</code> | <code>BIGINT</code> | Número de registros que se han saltado antes de que se empiece a insertar o actualizar. |
| <code>ROWS_INSERTED</code> | <code>BIGINT</code> | Número de filas insertadas en la tabla de destino. |
| <code>ROWS_UPDATED</code> | <code>BIGINT</code> | Número de filas de la tabla de destino actualizadas con información de los registros importados (registros cuyo valor de clave primaria ya existe en la tabla). |
| <code>ROWS_REJECTED</code> | <code>BIGINT</code> | Número de registros que no se han podido importar. |
| <code>ROWS_COMMITTED</code> | <code>BIGINT</code> | Número de registros importados satisfactoriamente y confirmados en la base de datos. |
| <code>MSG_RETRIEVAL</code> | <code>VARCHAR(512)</code> | Sentencia de SQL que sirve para recuperar los mensajes creados por este programa de utilidad. Por ejemplo: <pre>SELECT SQLCODE, MSG FROM TABLE (SYSPROC.ADMIN_GET_MSGS ('1203498_txu')) AS MSG</pre> |
| <code>MSG_REMOVAL</code> | <code>VARCHAR(512)</code> | Sentencia de SQL que sirve para hacer limpieza de los mensajes creados por este programa de utilidad. Por ejemplo: <pre>CALL SYSPROC.ADMIN_REMOVE_MSGS ('1203498_txu')</pre> |

Modificadores de tipo de archivo para el programa de utilidad de importación

Tabla 43. Modificadores de tipo de archivo válidos para el programa de utilidad de importación: Todos los formatos de archivo

| Modificador | Descripción |
|--------------------|---|
| compound= <i>x</i> | <p><i>x</i> es un número entre 1 y 100 inclusive. Utiliza SQL compuesto no atómico para insertar los datos y se intentarán cada vez <i>x</i> sentencias.</p> <p>Si se especifica este modificador y la anotación cronológica de transacciones no es suficientemente grande, la operación de importación fallará. La anotación cronológica de transacciones debe ser suficientemente grande para dar cabida al número de filas especificadas por COMMITCOUNT, o el número de filas del archivo de datos si no se especifica COMMITCOUNT. Por lo tanto, se recomienda especificar la opción COMMITCOUNT para evitar el desbordamiento de la anotación cronológica de transacciones.</p> <p>Este modificador es incompatible con la modalidad INSERT_UPDATE, las tablas jerárquicas y los siguientes modificadores: usedefaults, identitymissing, identityignore, generatedmissing y generatedignore.</p> |
| generatedignore | <p>Este modificador informa al programa de utilidad que los datos para todas las columnas generadas existen en el archivo de datos pero que se deberán ignorar. Esto hace que el programa de utilidad genere todos los valores para las columnas generadas. Este modificador no se puede utilizar con el modificador generatedmissing.</p> |
| generatedmissing | <p>Si se especifica este modificador, el programa de utilidad supone que el archivo de datos de entrada no contiene datos para las columnas generadas (ni siquiera ningún NULL) y, por consiguiente, generará un valor para cada fila. Este modificador no se puede utilizar con el modificador generatedignore.</p> |
| identityignore | <p>Este modificador informa al programa de utilidad que los datos para la columna de identidad existen en el archivo de datos pero que se deberán ignorar. Esto hace que el programa de utilidad genere todos los valores de identidad. El comportamiento será el mismo para las columnas de identidad GENERATED ALWAYS y GENERATED BY DEFAULT. Esto significa que, para las columnas GENERATED ALWAYS, no se rechazará ninguna fila. Este modificador no se puede utilizar con el modificador identitymissing.</p> |
| identitymissing | <p>Si se especifica este modificador, el programa de utilidad supone que el archivo de datos de entrada no contiene datos para la columna de identidad (ni siquiera ningún NULL) y, por consiguiente, generará un valor para cada fila. El comportamiento será el mismo para las columnas de identidad GENERATED ALWAYS y GENERATED BY DEFAULT. Este modificador no se puede utilizar con el modificador identityignore.</p> |

Tabla 43. Modificadores de tipo de archivo válidos para el programa de utilidad de importación: Todos los formatos de archivo (continuación)

| Modificador | Descripción |
|---------------------------|---|
| lobsinfile | <p><i>vía-lob</i> especifica la vía de acceso a los archivos que contienen datos de LOB.</p> <p>Cada vía de acceso contiene por lo menos un archivo que contiene por lo menos un LOB al que apunta un Especificador de ubicación de LOB (LLS) en el archivo de datos. El LLS es una representación de serie de la ubicación de un LOB en un archivo almacenado en la vía de acceso del archivo LOB. El formato de un LLS es <i>filename.ext.nnn.mmm/</i>, donde <i>filename.ext</i> es el nombre del archivo que contiene el LOB, <i>nnn</i> es el desplazamiento en bytes del LOB dentro del archivo y <i>mmm</i> es la longitud del LOB en bytes. Por ejemplo, si se almacena la serie <i>db2exp.001.123.456/</i> en el archivo de datos, el LOB está ubicado en el desplazamiento 123 dentro del archivo <i>db2exp.001</i> tiene una longitud de 456 bytes.</p> <p>La cláusula LOBS FROM especifica dónde están situados los archivos LOB cuando se utiliza el modificador "lobsinfile". La cláusula LOBS FROM activará implícitamente el comportamiento de LOBSINFILE. La cláusula LOBS FROM lleva al programa de utilidad IMPORT la lista de vías de acceso para buscar los archivos LOB durante la importación de los datos.</p> <p>Para indicar un LOB nulo, entre el tamaño como -1. Si el tamaño se especifica como 0, se trata como un LOB de longitud 0. Para los LOB nulos de longitud -1, se pasan por alto el desplazamiento y el nombre de archivo. Por ejemplo, el LLS de un LOB nulo puede ser <i>db2exp.001.7.-1/</i>.</p> |
| no_type_id | Sólo es válido cuando se está realizando la importación a una sola subtabla. El uso típico consiste en exportar datos de una tabla normal y luego invocar una operación de importación (utilizando este modificador) para convertir los datos en una sola subtabla. |
| nodefaults | <p>Si no se especifica explícitamente una columna fuente para una columna de tabla de destino y la columna de tabla no es anulable, no se cargan los valores por omisión. Sin esta opción, si no se especifica explícitamente una columna fuente para una de las columnas de tabla de destino, se produce una de las siguientes acciones:</p> <ul style="list-style-type: none"> • Si se puede especificar un valor por omisión para una columna, se carga el valor por omisión • Si la columna es anulable y no se puede especificar un valor por omisión para dicha columna, se carga un NULL • Si la columna no es anulable y no se puede especificar un valor por omisión, se devuelve un error y el programa de utilidad deja de procesarse. |
| norowwarnings | Suprime todos los avisos sobre filas rechazadas. |
| rowchangetimestampignore | Este modificador informa al programa de utilidad que los datos de la columna de indicación de fecha y hora de cambio de fila existen en el archivo de datos, pero deben omitirse. Esto hace que el programa de utilidad genere toda la columna ROW CHANGE TIMESTAMP (indicación de fecha y hora de cambio de fila). El comportamiento será el mismo tanto para las columnas GENERATED ALWAYS como para GENERATED BY DEFAULT. Esto significa que, para las columnas GENERATED ALWAYS, no se rechazará ninguna fila. Este modificador no se puede utilizar con el modificador rowchangetimestampmissing: |
| rowchangetimestampmissing | Si se especifica este modificador, el programa de utilidad presupone que el archivo de datos de entrada no contiene ningún dato para la columna de indicación de fecha y hora de cambio de fila (ni siquiera ningún NULL) y por lo tanto, generará un valor para cada fila. El comportamiento será el mismo tanto para las columnas GENERATED ALWAYS como para GENERATED BY DEFAULT. Este modificador no se puede utilizar con el modificador rowchangetimestampignore. |

Tabla 43. Modificadores de tipo de archivo válidos para el programa de utilidad de importación: Todos los formatos de archivo (continuación)

| Modificador | Descripción |
|--------------|--|
| seclabelchar | <p>Indica que las etiquetas de seguridad en el archivo fuente de entrada están en formato de serie para los valores de etiqueta de seguridad en lugar de en el formato numérico codificado por omisión. IMPORT convierte cada etiqueta de seguridad al formato interno al cargarse. Si una serie no tiene el formato adecuado, la fila no se carga y se devuelve un aviso (SQLSTATE 01H53). Si la serie no representa una etiqueta de seguridad válida que forma parte de la política de seguridad que protege la tabla, la fila no se carga y se devuelve un aviso (SQLSTATE 01H53, SQLCODE SQL3243W).</p> <p>Este modificador no puede especificarse si se especifica el modificador seclabelname, de lo contrario la importación falla y se devuelve un error (SQLCODE SQL3525N).</p> |
| seclabelname | <p>Indica que las etiquetas de seguridad en el archivo fuente de entrada están indicadas por su nombre en lugar de en el formato numérico codificado por omisión. IMPORT convertirá el nombre a la etiqueta de seguridad adecuada, si existe. Si no existe ninguna etiqueta de seguridad con el nombre indicado para la política de seguridad que protege la tabla, la fila no se carga y se devuelve un aviso (SQLSTATE 01H53, SQLCODE SQL3244W).</p> <p>Este modificador no puede especificarse si se especifica el modificador seclabelchar, de lo contrario la importación falla y se devuelve un error (SQLCODE SQL3525N).</p> <p>Nota: Si el tipo de archivo es ASC, los espacios a continuación del nombre de la etiqueta de seguridad se interpretarán como parte del nombre. Para evitarlo utilice el modificador de tipo de archivo striptblanks para asegurarse de que se eliminan los espacios.</p> |
| usedefaults | <p>Si se ha especificado una columna fuente para una columna de tabla de destino, pero dicha columna fuente no contiene datos para una o más instancias de fila, se cargan los valores por omisión. He aquí unos ejemplos de datos que faltan:</p> <ul style="list-style-type: none"> • Para archivos DEL: se especifican dos delimitadores de columna adyacentes (",,") o dos delimitadores de columnas adyacentes separados por un número arbitrario de espacios (" , ") para un valor de columna. • Para archivos DEL/ASC/WSF: una fila que no tiene suficientes columnas o que no es suficientemente larga para la especificación original. <p>Nota: Para archivos ASC, los valores de columna NULL no se considera que falten de modo explícito y no se sustituirán los valores de columna NULL por valores por omisión. Los valores de columna NULL se representan mediante todos los caracteres de espacio para columnas numéricas, de fecha, hora e indicación de fecha y hora o utilizando NULL INDICATOR para una columna de cualquier tipo para indicar que la columna es NULL.</p> <p>Sin esta opción, si la columna fuente no contiene datos para una instancia de fila, se produce una de las acciones siguientes:</p> <ul style="list-style-type: none"> • Para archivos DEL/ASC/WSF: Si la columna es anulable, se carga un NULL. Si la columna no es anulable, el programa de utilidad rechaza la fila. |

Tabla 44. Modificadores de tipo de archivo válidos para el programa de utilidad de importación: Formatos de archivo ASCII (ASC/DEL)

| Modificador | Descripción |
|-------------------------|---|
| codepage= <i>x</i> | <p><i>x</i> es una serie de caracteres ASCII. El valor se interpreta como la página de códigos de los datos del archivo de entrada. Convierte los datos de tipo carácter de esta página de códigos a la página de códigos de la aplicación durante la operación de importación.</p> <p>Se aplican las siguientes normas:</p> <ul style="list-style-type: none"> • Para DBCS (gráfico) puro, DBCS mixto y EUC, los delimitadores están restringidos al rango de x00 a x3F, inclusive. • nullindchar debe especificar símbolos incluidos en el conjunto ASCII estándar entre los elementos de código x20 t x7F, inclusive. Esto hace referencia a símbolos y elementos de código ASCII. <p>Nota:</p> <ol style="list-style-type: none"> 1. El modificador codepage no se puede utilizar con el modificador lobsinfile. 2. Si se produce expansión de datos cuando la página de códigos se convierte de la página de códigos de la aplicación a la página de códigos de la base de datos, los datos podrían truncarse y se puede producir una pérdida de datos. |
| dateformat=" <i>x</i> " | <p><i>x</i> es el formato de la fecha en el archivo fuente.² Los elementos de fecha válidos son:</p> <p>AAAA - Año (cuatro dígitos entre 0000 y 9999) M - Mes (uno o dos dígitos entre 1 y 12) MM - Mes (dos dígitos entre 1 y 12; se excluye mutuamente con M) D - Día (uno o dos dígitos entre 1 y 31) DD - Día (dos dígitos entre 1 y 31; se excluye mutuamente con D) DDD - Día del año (tres dígitos entre 001 y 366; se excluye mutuamente con otros elementos de día o mes)</p> <p>Se asigna un valor por omisión de 1 para cada elemento que no se haya especificado. He aquí unos ejemplos de formatos de fecha:</p> <p>"D-M-AAAA" "MM.DD.AAAA" "AAAADDD"</p> |
| implieddecimal | <p>La ubicación de una coma decimal implícita la determina la definición de columna; ya no se supone que está al final del valor. Por ejemplo, el valor 12345 se carga en una columna DECIMAL(8,2) como 123,45, <i>no</i> como 12345,00.</p> |

Tabla 44. Modificadores de tipo de archivo válidos para el programa de utilidad de importación: Formatos de archivo ASCII (ASC/DEL) (continuación)

| Modificador | Descripción |
|----------------|---|
| timeformat="x" | <p>x es el formato de la hora en el archivo fuente.² Los elementos de hora válidos son:</p> <ul style="list-style-type: none"> H - Hora (uno o dos dígitos entre 0 y 12 para un sistema de 12 horas y entre 0 y 24 para un sistema de 24 horas) HH - Hora (dos dígitos entre 0 y 12 para un sistema de 12 horas y entre 0 y 24 para un sistema de 24 horas; se excluye mutuamente con H) M - Minuto (uno o dos dígitos entre 0 y 59) MM - Minuto (dos dígitos entre 0 y 59; se excluye mutuamente con M) S - Segundo (uno o dos dígitos entre 0 y 59) SS - Segundo (dos dígitos entre 0 y 59; se excluye mutuamente con S) SSSSS - Segundo del día después de medianoche (5 dígitos entre 00000 y 86399; se excluye mutuamente con otros elementos de hora) TT - Indicador de meridiano (AM o PM) <p>Se asigna un valor por omisión de 0 para cada elemento que no se haya especificado. He aquí unos ejemplos de formatos de hora:</p> <p>"HH:MM:SS" "HH.MM TT" "SSSSS"</p> |

Tabla 44. Modificadores de tipo de archivo válidos para el programa de utilidad de importación: Formatos de archivo ASCII (ASC/DEL) (continuación)

| Modificador | Descripción |
|---------------------|---|
| timestampformat="x" | <p>x es el formato de la indicación de fecha y hora en el archivo fuente.² Los elementos válidos de la indicación de fecha y hora son:</p> <ul style="list-style-type: none"> AAAA - Año (cuatro dígitos entre 0000 y 9999) M - Mes (uno o dos dígitos entre 1 y 12) MM - Mes (dos dígitos entre 01 y 12; se excluye mutuamente con M y MMM) MMM - Mes (abreviación de tres letras no sensible a mayúsculas y minúsculas para el nombre del mes; se excluye mutuamente con M y MM) D - Día (uno o dos dígitos entre 1 y 31) DD - Día (dos dígitos entre 1 y 31; se excluye mutuamente con D) DDD - Día del año (tres dígitos entre 001 y 366; se excluye mutuamente con otros elementos de día o mes) H - Hora (uno o dos dígitos entre 0 y 12 para un sistema de 12 horas y entre 0 y 24 para un sistema de 24 horas) HH - Hora (dos dígitos entre 0 y 12 para un sistema de 12 horas y entre 0 y 24 para un sistema de 24 horas; se excluye mutuamente con H) M - Minuto (uno o dos dígitos entre 0 y 59) MM - Minuto (dos dígitos entre 0 y 59; se excluye mutuamente con M, minuto) S - Segundo (uno o dos dígitos entre 0 y 59) SS - Segundo (dos dígitos entre 0 y 59; se excluye mutuamente con S) SSSS - Segundo del día después de medianoche (5 dígitos entre 00000 y 86399; se excluye mutuamente con otros elementos de hora) U (de 1 a 12 veces) <ul style="list-style-type: none"> - Segundos fraccionados (el número de casos de U representa el número de dígitos con cada dígito comprendido entre 0 y 9) TT - Indicador de meridiano (AM o PM) <p>Se asigna un valor por omisión de 1 para los elementos AAAA, M, MM, D, DD o DDD no especificados. Se asigna un valor por omisión de 'Jan' a un elemento MMM no especificado. Se asigna un valor por omisión de 0 para todos los demás elementos no especificados. A continuación se muestra un ejemplo de un formato de indicación de la hora:</p> <p style="text-align: center;">"AAAA/MM/DD HH:MM:SS.UUUUUU"</p> <p>Los valores válidos para el elemento MMM incluyen: 'ene', 'feb', 'mar', 'abr', 'may', 'jun', 'jul', 'ago', 'sep', 'oct', 'nov' y 'dic'. Estos valores son sensibles a las mayúsculas y minúsculas.</p> <p>El ejemplo siguiente ilustra cómo importar datos que contienen formatos de fecha y hora definidos por el usuario a una tabla llamada schedule:</p> <pre>db2 import from delfile2 of del modified by timestampformat="aaaa.mm.dd hh:mm tt" insert into schedule</pre> |

Tabla 44. Modificadores de tipo de archivo válidos para el programa de utilidad de importación: Formatos de archivo ASCII (ASC/DEL) (continuación)

| Modificador | Descripción |
|--------------------|--|
| usegraphiccodepage | <p>Si se proporciona usegraphiccodepage, se supone que los datos importados a campos de datos de objeto gráfico o de gran objeto de caracteres de doble byte (DBCLOB) se encuentran en la página de códigos gráfica. Se supone que el resto de los datos estarán en la página de códigos de caracteres. La página de códigos gráfica está asociada con la página de códigos de caracteres. IMPORT determina la página de códigos de caracteres mediante el modificador codepage, si se especifica, o mediante la página de códigos de la aplicación si no se especifica el modificador codepage.</p> <p>Este modificador debe utilizarse junto con el archivo de datos delimitado generado por la recuperación de descarte de tabla sólo si la tabla que se recupera tiene datos gráficos.</p> <p>Restricciones</p> <p>El modificador usegraphiccodepage NO DEBE especificarse con archivos DEL creados por el programa de utilidad EXPORT, ya que estos archivos contienen datos codificados en una sola página de códigos. También ignorarán el modificador usegraphiccodepage los grandes objetos de caracteres de doble byte (DBCLOB) en los archivos.</p> |
| xmlchar | <p>Especifica que los documentos XML se codifican con la página de códigos de caracteres.</p> <p>Esta opción es útil para procesar documentos XML que estén codificados con la página de códigos de caracteres especificada pero que no contengan una declaración de codificación.</p> <p>Para cada documento, si existe un identificador de declaración y contiene un atributo de codificación, la codificación debe coincidir con la página de códigos de caracteres, de lo contrario se rechazará la fila que contiene el documento. Tenga en cuenta que la página de códigos es el valor especificado por el modificador de tipo de archivo codepage o la página de códigos de la aplicación, si no se ha especificado. Por omisión, los documentos están codificados en Unicode o bien contienen un identificador de declaración con un atributo de codificación.</p> |
| xmlgraphic | <p>Especifica que los documentos XML se codifican con la página de códigos de gráficos especificada.</p> <p>Esta opción es útil para procesar documentos XML que estén codificados con una página de códigos de gráficos específica, pero que no contengan una declaración de codificación.</p> <p>Para cada documento, si existe un identificador de declaración y contiene un atributo de codificación, la codificación debe coincidir con la página de códigos de gráficos, de lo contrario se rechazará la fila que contiene el documento. Tenga en cuenta que la página de códigos de gráficos es el componente gráfico del valor especificado por el modificador de tipo de archivo codepage o el componente gráfico de la página de códigos de la aplicación, si no se ha especificado. Por omisión, los documentos están codificados en Unicode o bien contienen un identificador de declaración con un atributo de codificación.</p> <p>Nota: Si se especifica el modificador xmlgraphic con el mandato IMPORT, el documento XML que se va a importar debe codificarse en la página de códigos UTF-16. De lo contrario, el documento XML se puede rechazar con un error de análisis, o se puede importar a la tabla dañando los datos.</p> |

Tabla 45. Modificadores de tipo de archivo válidos para el programa de utilidad de importación: Formato de archivo ASC (ASCII no delimitado)

| Modificador | Descripción |
|----------------|---|
| nochecklengths | Si se especifica nochecklengths, se intenta importar cada fila, incluso si los datos fuente tienen una definición de columna que excede el tamaño de la columna de tabla de destino. Dichas filas pueden importarse satisfactoriamente si la conversión de página de códigos hace que los datos fuente se acorten; por ejemplo, datos EUC de 4 bytes en la fuente pueden acortarse a datos DBCS de 2 bytes en el destino y necesitar la mitad del espacio. Esta opción es especialmente útil si se sabe que los datos fuente cabrán en todos los casos, a pesar de las discrepancias de las definiciones de columna. |
| nullindchar=x | x es un carácter individual. Cambia el carácter que indica un valor NULL a x . El valor por omisión de x es \backslash . ³ Este modificador es sensible a las mayúsculas y minúsculas para archivos de datos EBCDIC, excepto cuando el carácter es una letra inglesa. Por ejemplo, si se especifica que el carácter indicador de nulo sea la letra N, se reconoce también n como un indicador de nulo. |
| reclen=x | x es un entero con un valor máximo de 32 767. Se leen x caracteres para cada fila y no se utiliza un carácter de nueva línea para indicar el final de la fila. |
| striptblanks | Trunca los espacios en blanco de cola al cargar datos en un campo de longitud variable. Si no se especifica esta opción, se conservan los espacios en blanco. En el ejemplo siguiente, striptblanks hace que el programa de utilidad de importación trunque los espacios en blanco de cola: <pre>db2 import from miarchivo.asc of asc modified by striptblanks method l (1 10, 12 15) messages msgs.txt insert into staff</pre> Esta opción no se puede especificar junto con striptnulls. Son opciones que se excluyen mutuamente. Esta opción sustituye a la opción t obsoleta, que se soporta únicamente por compatibilidad con niveles anteriores. |
| striptnulls | Trunca los NULL de cola (caracteres 0x00) al cargar datos en un campo de longitud variable. Si no se especifica esta opción, se conservan los NULL. Esta opción no se puede especificar junto con striptblanks. Son opciones que se excluyen mutuamente. Esta opción sustituye a la opción padwithzero obsoleta, que sólo se soporta por compatibilidad con niveles anteriores. |

Tabla 46. Modificadores de tipo de archivo válidos para el programa de utilidad de importación: Formato de archivo DEL (ASCII delimitado)

| Modificador | Descripción |
|-------------|--|
| chardelx | x es un delimitador de serie de un solo carácter. El valor por omisión son las comillas dobles ("). El carácter especificado se utiliza en lugar de las comillas para encerrar una serie de caracteres. ³⁴ Si desea especificar explícitamente las comillas como delimitador de serie de caracteres, debe especificarse de la manera siguiente: <pre> modified by chardel""</pre> También se pueden especificar las comillas simples (') como delimitador de serie de caracteres. En el ejemplo siguiente, chardel'' hace que el programa de utilidad de importación interprete cualquier comilla simple (') que encuentre como un delimitador de serie de caracteres: <pre>db2 "import from myfile.del of del modified by chardel'' method p (1, 4) insert into staff (id, years)"</pre> |

Tabla 46. Modificadores de tipo de archivo válidos para el programa de utilidad de importación: Formato de archivo DEL (ASCII delimitado) (continuación)

| Modificador | Descripción |
|-----------------|---|
| coldelx | <p>x es un delimitador de columna de un solo carácter. El valor por omisión es una coma (.). En lugar de una coma, se utiliza el carácter especificado para indicar el final de una columna.³⁴</p> <p>En el ejemplo siguiente, coldel; hace que el programa de utilidad de importación interprete cualquier signo de punto y coma (;) que encuentre como un delimitador de columna:</p> <pre>db2 import from miarchivo.del of del modified by coldel; messages msgs.txt insert into staff</pre> |
| decplusblank | <p>Carácter de signo más. Hace que a los valores decimales positivos se les ponga un espacio en blanco como prefijo en lugar de un signo más (+). La acción por omisión es poner a los valores decimales positivos un signo más como prefijo.</p> |
| decptx | <p>x es un sustituto de un solo carácter del punto como carácter de coma decimal. El valor por omisión es un punto (.). En lugar de un punto, se utilizará el carácter especificado como carácter de coma decimal.³⁴</p> <p>En el ejemplo siguiente, decpt; hace que el programa de utilidad de importación interprete cualquier punto y coma (;) que encuentre como una coma decimal:</p> <pre>db2 "import from myfile.del of del modified by chardel" decpt; messages msgs.txt insert into staff"</pre> |
| delprioritychar | <p>La prioridad por omisión actual para los delimitadores es: delimitador de registro, delimitador de carácter, delimitador de columna. Este modificador protege a las aplicaciones existentes que dependen de la prioridad más antigua invirtiendo las prioridades de los delimitadores del modo siguiente: delimitador de carácter, delimitador de registro, delimitador de columna. Sintaxis:</p> <pre>db2 import ... modified by delprioritychar ...</pre> <p>Por ejemplo, dado el archivo de datos DEL siguiente:</p> <pre>"Smith, Joshua",4000,34.98<delimitador de fila> "Vincent,<delimitador de fila>, is a manager", 4005,44.37<delimitador de fila></pre> <p>Si se especifica el modificador delprioritychar, sólo habrá dos filas en este archivo de datos. El segundo <delimitador de fila> se interpretará como parte de la primera columna de datos de la segunda fila, mientras que el primero y tercero <delimitadores de fila> se interpretarán como delimitadores de registro reales. Si <i>no</i> se especifica este modificador, habrá tres filas en este archivo de datos, cada una de las cuales estará delimitada por un <delimitador de fila>.</p> |
| keepblanks | <p>Conserva los blancos iniciales y de cola en cada campo de tipo CHAR, VARCHAR, LONG VARCHAR o CLOB. Sin esta opción, se eliminan todos los espacios en blanco iniciales y de cola que no están incluidos entre los delimitadores de caracteres y se inserta un NULL en la tabla para todos los campos en blanco.</p> |

Tabla 46. Modificadores de tipo de archivo válidos para el programa de utilidad de importación: Formato de archivo DEL (ASCII delimitado) (continuación)

| Modificador | Descripción |
|-------------|---|
| nochardel | <p>El programa de utilidad de importación asumirá que todos los bytes encontrados entre los delimitadores de columna forman parte de los datos de la columna. Los delimitadores de carácter se analizarán como parte de los datos de la columna. Esta opción no debe especificarse si los datos se han exportado utilizando DB2 (a menos que se haya especificado nochardel en el momento de la exportación). Se proporciona para dar soporte a archivos de datos de proveedores que no tienen delimitadores de caracteres. El uso incorrecto podría causar la pérdida o la corrupción de los datos.</p> <p>Esta opción no se puede especificar con charde1x, delprioritychar o nodoubledel. Son opciones que se excluyen mutuamente.</p> |
| nodoubledel | Suprime el reconocimiento de los delimitadores de caracteres dobles. |

Tabla 47. Modificadores de tipo de archivo válidos para el programa de utilidad de importación: Formato de archivo IXF

| Modificador | Descripción |
|-----------------------------|--|
| forcein | <p>Indica al programa de utilidad que acepte datos a pesar de las discrepancias de página de códigos y que suprima la conversión entre páginas de códigos.</p> <p>Se comprueban los campos de destino de longitud fija para verificar que son suficientemente grandes para los datos. Si se especifica nochecklengths, no se realiza ninguna comprobación y se efectúa un intento de importación de cada fila.</p> |
| indexixf | Indica al programa de utilidad que descarte todos los índices definidos actualmente en la tabla existente y que cree otros nuevos a partir de las definiciones de índice del archivo PC/IXF. Esta opción sólo se puede utilizar cuando se está sustituyendo el contenido de una tabla. No se puede utilizar con una vista o cuando se especifica una <i>columna-inserción</i> . |
| indexschema= <i>esquema</i> | Utiliza el <i>esquema</i> especificado para el nombre de índice durante la creación de índices. Si no se especifica <i>esquema</i> (pero la palabra clave indexschema sí se especifica), se utiliza el ID de usuario de conexión. Si no se especifica la palabra clave, se utiliza el esquema del archivo IXF. |
| nochecklengths | Si se especifica nochecklengths, se intenta importar cada fila, incluso si los datos fuente tienen una definición de columna que excede el tamaño de la columna de tabla de destino. Dichas filas pueden importarse satisfactoriamente si la conversión de página de códigos hace que los datos fuente se acorten; por ejemplo, datos EUC de 4 bytes en la fuente pueden acortarse a datos DBCS de 2 bytes en el destino y necesitar la mitad del espacio. Esta opción es especialmente útil si se sabe que los datos fuente cabrán en todos los casos, a pesar de las discrepancias de las definiciones de columna. |
| forcecreate | Especifica que la tabla debe crearse con una posible información limitada o con ausencia de información después de devolver SQL3311N durante una operación de importación. |

Tabla 48. Comportamiento de IMPORT al utilizar codepage y usegraphiccodepage

| codepage=N | usegraphiccodepage | Comportamiento de IMPORT |
|------------|--------------------|---|
| Ausente | Ausente | Se supone que todos los datos del archivo están en la página de códigos de la aplicación. |

Tabla 48. Comportamiento de IMPORT al utilizar codepage y usegraphiccodepage (continuación)

| codepage=N | usegraphiccodepage | Comportamiento de IMPORT |
|------------|--------------------|--|
| Presente | Ausente | Se supone que todos los datos del archivo están en la página de códigos N. Aviso: Los datos gráficos se corromperán cuando se importen a la base de datos si N es una página de códigos de un solo byte. |
| Ausente | Presente | Se supone que los datos de caracteres del archivo están en la página de códigos de la aplicación. Se supone que los datos gráficos están en la página de códigos de los datos gráficos de aplicación. Si la página de códigos de la aplicación es de un solo byte, se supone que todos los datos están en la página de códigos de la aplicación. Aviso: Si la página de códigos de la aplicación es de un solo byte, los datos gráficos se corromperán cuando se importen a la base de datos, aunque ésta contenga columnas gráficas. |
| Presente | Presente | Se presupone que los datos del carácter se encuentran en la página de códigos N. Se presupone que los datos gráficos se encuentran en la página de códigos gráficos N. Si N es una página de códigos de un solo byte o de doble byte, se supone que todos los datos están en la página de códigos N. Aviso: Los datos gráficos se corromperán cuando se importen a la base de datos si N es una página de códigos de un solo byte. |

Nota:

1. El programa de utilidad de importación no emite ningún aviso si se intentan utilizar tipos de archivo no soportados con la opción **MODIFIED BY**. Si se realiza dicho intento, la operación de importación fallará y se devolverá un código de error.
2. Es obligatorio escribir la serie de formato de fecha entre comillas dobles. Los separadores de campo no pueden contener ninguno de los caracteres siguientes: a-z, A-Z y 0-9. El separador de campo no debe ser igual que el delimitador de carácter o de campo en el formato de archivo DEL. Un separador de campo es opcional si las posiciones inicial y final de un elemento no son ambiguas. La ambigüedad puede existir si (en función del modificador) se usan elementos tales como D, H, M o S, debido a la longitud variable de las entradas.

Para formatos de indicación de la hora, hay que tener cuidado de evitar la ambigüedad entre los descriptores de mes y de minuto, dado que ambos utilizan la letra M. Un campo de mes debe estar junto a otros campos de fecha. Un campo de minuto debe ser adyacente a otros campos de hora. He aquí unos formatos ambiguos de indicación de la hora:

- "M" (puede ser mes o minuto)
- "M:M" (¿Cuál es cada uno?)
- "M:AAAA:M" (Ambos se interpretan como mes.)
- "S:M:AAAA" (adyacente a un valor de hora y un valor de fecha)

En los casos ambiguos, el programa de utilidad informará con un mensaje de error y la operación fallará.

He aquí unos formatos no ambiguos de indicación de fecha y hora:

```
"M:AAAA" (Mes)
"S:M" (Minuto)
"M:AAAA:S:M" (Mes...Minuto)
"M:H:AAAA:M:D" (Minuto...Mes)
```

Algunos caracteres, como las comillas dobles y las barras inclinadas invertidas, deben ir precedidos de un carácter de escape (por ejemplo, \).

3. Los valores de tipo carácter proporcionados para los modificadores `chardel`, `coldel` o `decpt` deben especificarse en la página de códigos de los datos fuente.

Se puede especificar el elemento de código de carácter (en lugar del símbolo de carácter), utilizando la sintaxis `xJJ` o `0xJJ`, donde `JJ` es la representación hexadecimal del elemento de código. Por ejemplo, para especificar el carácter `#` como delimitador de columna, utilice uno de los siguientes:

```
... modified by coldel# ...
... modified by coldel0x23 ...
... modified by coldelX23 ...
```

4. En la sección *Consideraciones sobre el delimitador para mover datos* se muestran las restricciones que se aplican a los caracteres que se pueden utilizar como alteración temporal de los delimitadores.
5. Los siguientes modificadores de tipo de archivo no están permitidos cuando se importa a un apodo:
 - `indexif`
 - `indexschema`
 - `dldelfiletype`
 - `nodefaults`
 - `usedefaults`
 - `no_type_idfiletype`
 - `generatedignore`
 - `generatedmissing`
 - `identityignore`
 - `identitymissing`
 - `lobsinfile`
6. El formato de archivo **WSF** no está soportado para las columnas XML. El soporte para este formato de archivo también ha quedado en desuso y se puede eliminar en futuros releases. Se recomienda empezar a usar un formato de archivo compatible en lugar de archivos WSF antes de que se elimine el soporte.
7. La modalidad **CREATE** no está soportada para las columnas XML.
8. Todos los datos XML deben residir en archivos XML aparte del archivo de datos principal. Debe existir un XDS (Especificador de datos XML) (o un valor NULL) para cada columna XML en el archivo de datos principal.
9. Se supone que los documentos de XML están en formato Unicode o que contienen un identificador de declaración que incluye un atributo de codificación, a menos que se especifique el modificador de tipo de archivo `XMLCHAR` o `XMLGRAPHIC`.
10. Las filas que contienen documentos que no tengan formato correcto se rechazarán.
11. Si se especifica la opción **XMLVALIDATE**, los documentos que se validan satisfactoriamente se anotarán con respecto a su esquema coincidente con la

información de esquema a medida que se inserten. Las filas que contengan documentos que no consigan validarse ante el esquema coincidente serán rechazadas. Para realizar la validación satisfactoriamente, los privilegios del usuario que invoca la importación deben incluir al menos uno de los siguientes:

- Autorización DBADM
 - Privilegio USAGE sobre el esquema XML a utilizar en la validación
12. Al importar a una tabla que contiene una columna de indicación de fecha y hora de cambio de fila oculta implícitamente, no se cumplirá la propiedad oculta implícitamente de la columna. Por lo tanto el modificador de tipo de archivo `rowchangetimestampmissing` *debe* especificarse en el mandato de importación si los datos de la columna no existen en los datos que se van a importar y no existe ninguna lista de columna explícita.

Mandato INITIALIZE TAPE utilizando el procedimiento ADMIN_CMD

Inicializa cintas para las operaciones de copia de seguridad y restauración en los dispositivos de cinta de modalidad continua. Este mandato sólo está soportado en los sistemas operativos Windows.

Autorización

Una de las siguientes:

- *sysadm*
- *sysctrl*
- *sysmaint*

Conexión necesaria

Base de datos

Sintaxis del mandato

►► INITIALIZE TAPE ON *dispositivo* USING *tamaño-bloque* ►►

Parámetros del mandato

ON *dispositivo*

Especifica un nombre de dispositivo de cinta válido. El valor por omisión es `\\.\TAPE0`. El dispositivo especificado debe ser relativo al servidor.

USING *tamaño-bloque*

Especifica el tamaño de bloque para el dispositivo, en bytes. El dispositivo se inicializa para utilizar el tamaño de bloque especificado, si el valor está dentro del rango soportado de tamaños de bloque para el dispositivo.

El tamaño de almacenamiento intermedio especificado para el mandato `BACKUP DATABASE` y para `RESTORE DATABASE` debe ser divisible por el tamaño de bloque especificado aquí.

Si no se especifica un valor para este parámetro, el dispositivo se inicializa para utilizar el tamaño de bloque por omisión. Si se especifica un valor de cero, el dispositivo se inicializa para utilizar un tamaño de bloque de

longitud variable; si el dispositivo no soporta la modalidad de bloque de longitud variable, se devuelve un error.

Al hacer copia de seguridad en cinta, el uso del tamaño de bloque variable no está soportado actualmente. Si debe usar esta opción, asegúrese de que tiene implantados procedimientos bien probados que le permitan hacer una recuperación satisfactoria, utilizando imágenes de copia de seguridad que se crearon con un tamaño de bloque variable.

Al utilizar un tamaño de bloque variable, debe especificar un tamaño de almacenamiento intermedio de copia de seguridad que sea menor o igual que el límite máximo de los dispositivos de cinta que esté utilizando. Para optimizar el rendimiento, el tamaño del almacenamiento intermedio debe ser igual al límite de tamaño de bloque máximo del dispositivo que se utilice.

Ejemplo

Inicialice el dispositivo de cinta para que utilice un tamaño de bloque igual a 2048 bytes, si el valor está dentro del rango de tamaños de bloque soportados del dispositivo.

```
CALL SYSPROC.ADMIN_CMD( 'initialize tape using 2048' )
```

Notas de uso

El estado de la ejecución del mandato se devuelve en el área de comunicaciones SQL (SQLCA) obtenida a partir de la sentencia CALL.

Mandato LOAD utilizando el procedimiento ADMIN_CMD

Carga datos en una tabla DB2. Los datos que residen en el servidor pueden estar en forma de archivo, cinta o conexión con nombre. Los datos también se pueden cargar desde un cursor definido desde una consulta que se ejecuta para la base de datos conectada actualmente o una base de datos diferente con la misma instancia o mediante un script o aplicación escritos por un usuario. Si el atributo COMPRESS de la tabla está establecido en YES, los datos cargados estarán sujetos a compresión en todos los datos y en cada partición de la base de datos para los que ya exista un diccionario en la tabla, incluidos los datos del objeto de almacenamiento XML de la tabla.

Enlace rápido con “Modificadores de tipo de archivo para el programa de utilidad de carga” en la página 143.

Restricciones

El programa de utilidad de carga no soporta la carga de datos a nivel de jerarquía. El programa de utilidad de carga no es compatible con las tablas agrupadas en clústeres de rangos.

Ámbito

Este mandato puede emitirse para varias particiones de base de datos en una sola petición.

Autorización

Una de las siguientes:

- *dataaccess*
- Autorización LOAD en la base de datos y
 - Privilegio INSERT en la tabla cuando se invoca el programa de utilidad de carga en modalidad INSERT, en modalidad TERMINATE (para terminar una operación de inserción de carga anterior) o en modalidad RESTART (para reiniciar una operación de inserción de carga anterior)
 - Privilegio INSERT y DELETE en la tabla cuando se invoca el programa de utilidad de carga en modalidad REPLACE, en modalidad TERMINATE (para terminar una operación de sustitución de carga anterior) o RESTART (para reiniciar una operación de sustitución de carga anterior)
 - Privilegio INSERT en la tabla de excepción, si dicha tabla se utiliza como parte de la operación de carga.
- Para cargar datos en una tabla que tenga columnas protegidas, el ID de autorización de sesión debe tener credenciales LBAC que permitan el acceso de grabación a todas las columnas protegidas de la tabla. De lo contrario, la carga no se realizará y se devolverá un error (SQLSTATE 5U014).
- Para cargar datos en una tabla que tenga filas protegidas, el id de autorización de sesión debe tener una etiqueta de seguridad que cumpla los siguientes criterios:
 - Formar parte de la política de seguridad que protege la tabla
 - Haber sido otorgada al ID de autorización de sesión para el acceso de escritura o para todos los accesos

Si el id de autorización de sesión no contiene una etiqueta de seguridad como ésta, la carga no se realizará y se devolverá un error (SQLSTATE 5U014). Esta etiqueta de seguridad se utiliza para proteger una fila cargada si las credenciales LBAC del ID de autorización de sesión no permiten que se grave en la etiqueta de seguridad que protege dicha fila en los datos. Sin embargo, esto no sucede cuando la política de seguridad que protege la tabla se ha creado con la opción RESTRICT NOT AUTHORIZED WRITE SECURITY LABEL de la sentencia CREATE SECURITY POLICY. En este caso, la carga no se realiza y se devuelve un error (SQLSTATE 42519).

- Si se especifica la opción REPLACE, el ID de autorización de sesión debe tener la autorización para descartar la tabla.
- Si se especifica la opción LOCK WITH FORCE, se necesita la autorización SYSADM.

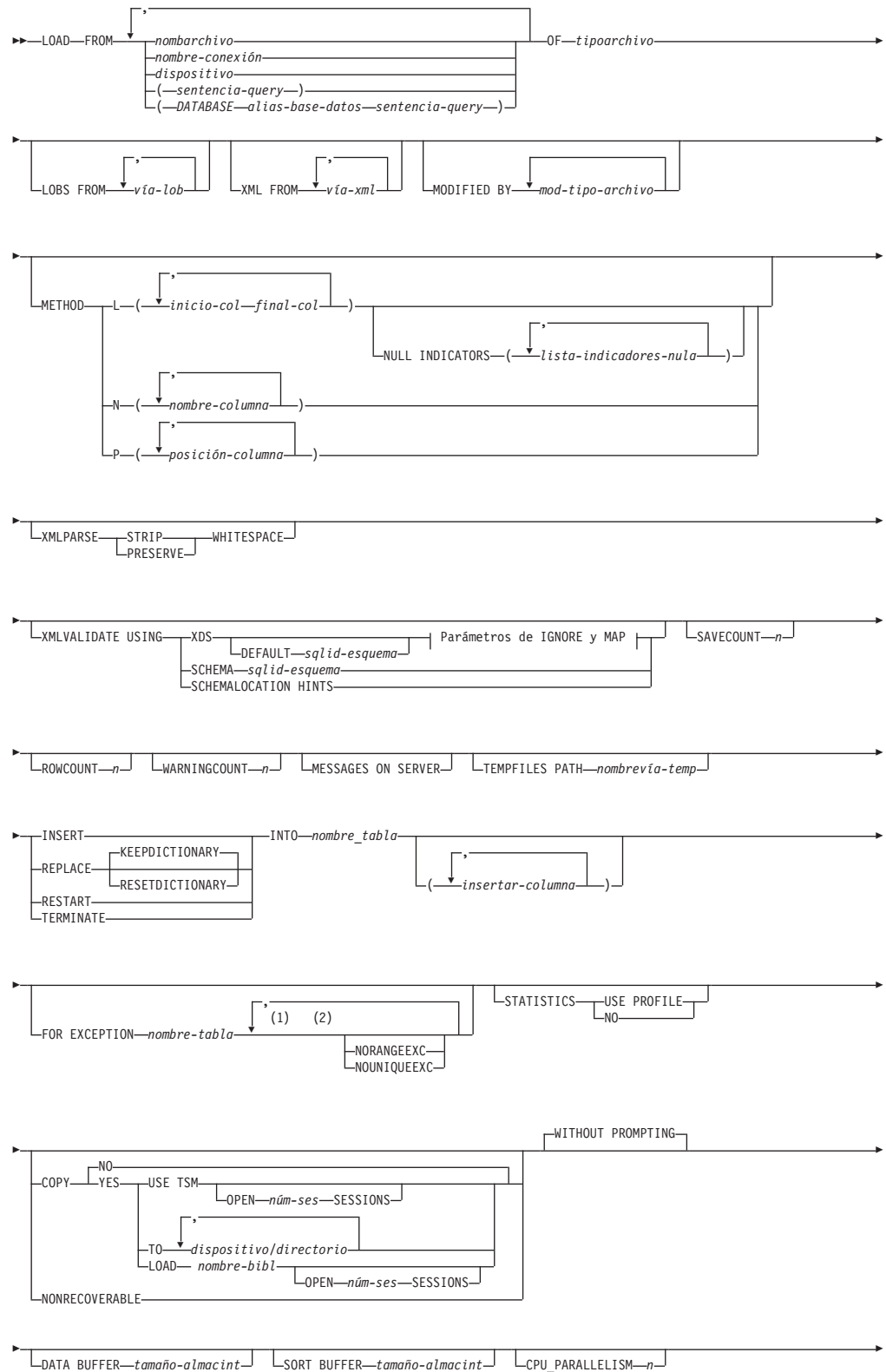
Dado que todos los procesos de carga (y todos los procesos del servidor DB2 en general) son propiedad del propietario de la instancia y todos estos procesos utilizan la identificación del propietario de la instancia para acceder a los archivos necesarios, el propietario de la instancia debe tener acceso de lectura a los archivos de entrada de datos. El propietario de instancia debe poder leer estos archivos de datos de entrada, independientemente de quién invoque el mandato.

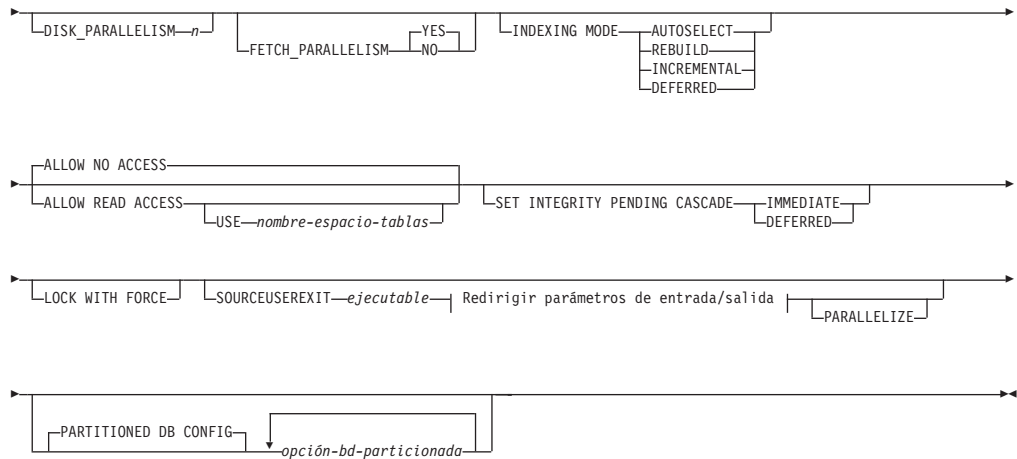
Conexión necesaria

Base de datos.

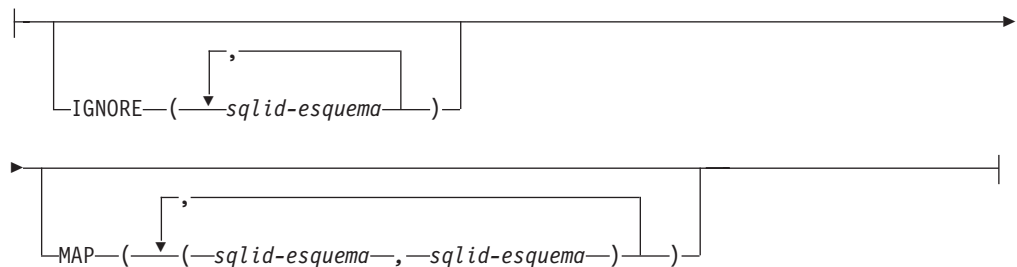
Instancia. No se necesita una conexión explícita. Si se ha establecido una conexión a la base de datos, se intenta una conexión implícita a la instancia local.

Sintaxis del mandato





Parámetros de IGNORE y de MAP:



Redirigir parámetros de entrada/salida:



Notas:

- 1 Estas palabras clave pueden aparecer en cualquier orden.
- 2 Cada una de estas palabras clave sólo puede aparecer una vez.

Parámetros del mandato

FROM *nombarchivo* | *nombre-conexión* | *dispositivo(sentencia-consulta)* | (*DATABASE alias-base-datos* *sentencia-query*)

Especifica el archivo, conexión o dispositivo que hace referencia a una sentencia de SQL que contiene los datos que se están cargando o la propia sentencia de SQL y la base de datos fuente opcional que se cargará desde el cursor.

La opción *sentencia-sql* se utiliza para CARGAR desde un cursor. Sólo contiene una sentencia de consulta que está entre paréntesis y que puede comenzar por VALUES, SELECT o WITH. Por ejemplo,

LOAD FROM (SELECT * FROM T1) OF CURSOR INSERT INTO T2

Cuando la cláusula *DATABASE alias-base-datos* se incluye antes de la sentencia de consulta en los paréntesis, el mandato LOAD intentará cargar los datos mediante la *sentencia-consulta* desde la base de datos

suministrada, tal y como se indica en el nombre *alias-base-datos*, que se define en el servidor. Debe apuntar a una base de datos que exista en el servidor y sea una base de datos diferente de la aplicación a la que está conectada actualmente. Observe que la carga, LOAD se ejecutará con el ID de usuario y la contraseña que se han suministrado explícitamente para la base de datos conectada actualmente (una conexión implícita harán que la carga, LOAD no se realice correctamente).

Si la fuente de entrada es un archivo, una conexión o un dispositivo, debe poderse acceder desde la partición del coordinador en el servidor.

Si se especifican varios nombres, éstos se procesarán en secuencia. Si el último elemento especificado es un dispositivo de cinta y se solicita al usuario una cinta, LOAD no se ejecutará correctamente y el procedimiento ADMIN_CMD devolverá un error.

Nota:

1. Debe utilizarse un nombre de archivo de vía de acceso totalmente calificada y que ya exista en el servidor.
2. Si se exportan datos a un archivo mediante el mandato *EXPORT* mediante el procedimiento *ADMIN_CMD*, el archivo de datos será propiedad del ID de usuario delimitado. Normalmente, este archivo ya no es accesible para el propietario de la instancia. Para ejecutar LOAD (cargar) desde CLP o el procedimiento *ADMIN_CMD*, el archivo de datos debe ser accesible para el ID de propietario de la instancia, por lo que debe otorgarse acceso de lectura al archivo de datos para el propietario de la instancia.
3. Se soporta la carga de datos de múltiples archivos IXF si los archivos están físicamente separados pero lógicamente forman un solo archivo. No se soporta si los archivos están física y lógicamente separados. (Múltiples archivos se considerarían uno solo lógicamente, si se hubiesen creados todos con una invocación del mandato *EXPORT*.)
4. Si carga datos XML de archivos a tablas en un entorno de base de datos particionada, los archivos de datos XML deben poderse leer desde todas las particiones de bases de datos donde se están cargando.

OF tipoarchivo

Especifica el formato de los datos:

- ASC (formato ASCII no delimitado).
- DEL (formato ASCII delimitado).
- IXF (Integration Exchange Format, versión PC) es un formato binario que utiliza exclusivamente DB2.
- CURSOR (cursor declarado para una sentencia SELECT o VALUES).

Nota: Cuando se utiliza un tipo de archivo CURSOR para cargar datos XML en una tabla en un entorno de base de datos distribuida, no se da soporte a las modalidades PARTITION_ONLY y LOAD_ONLY.

LOBS FROM vía-lob

Vía de acceso a los archivos de datos que contienen los valores LOB que se deben cargar. La vía de acceso debe finalizar con una barra inclinada. La vía de acceso debe estar totalmente calificada y ser accesible desde la partición del coordinador en el servidor. Los nombres de los archivos de datos LOB se almacenan en el archivo de datos principal (ASC, DEL o IXF), en la columna que se cargará en la columna LOB. El número máximo

de vías de acceso que se puede especificar es 999. Esto activará implícitamente el comportamiento de LOBSINFILE.

Esta opción se ignora cuando se especifica junto con el tipo de archivo CURSOR.

MODIFIED BY *mod-tipo-archivo*

Especifica opciones de modificador de tipo de archivo. Consulte "Modificadores de tipo de archivo para el programa de utilidad de carga" en la página 143.

METHOD

L Especifica los números de columna inicial y final de los que se deben cargar datos. Un número de columna es un desplazamiento de bytes respecto al principio de una fila de datos. Se numera empezando por 1. Este método sólo se puede utilizar con archivos ASC y es el único método válido para dicho tipo de archivo.

NULL INDICATORS *lista-indicadores-nulos*

Esta opción sólo se puede utilizar cuando se especifica el parámetro METHOD L, es decir, el archivo de entrada es un archivo ASC). Una lista de indicadores nulos es una lista de enteros positivos separados por comas que especifican el número de columna de cada campo de indicador nulo. El número de columna es el desplazamiento de bytes del campo de indicador nulo respecto al principio de una fila de datos. Debe haber una entrada en la lista de indicadores nulos para cada campo de datos definido en el parámetro METHOD L. Un número de columna de cero indica que el campo de datos correspondiente siempre contiene datos.

Un valor de Y en la columna de indicador NULL especifica que los datos de la columna son NULL. Cualquier carácter *distinto de* Y en la columna del indicador NULL especifica que los datos de columna no son NULL y se cargarán los datos de columna especificados por METHOD L.

El carácter de indicador NULL se puede cambiar utilizando la opción MODIFIED BY.

N Especifica los nombres de las columnas del archivo de datos que se deben cargar. Las mayúsculas y minúsculas de estos nombres de columna deben coincidir con las mayúsculas y minúsculas de los nombres correspondientes en los catálogos del sistema. Cada columna de tabla que no sea anulable debe tener una entrada correspondiente en la lista METHOD N. Por ejemplo, dados los campos de datos F1, F2, F3, F4, F5 y F6 y las columnas de tabla C1 INT, C2 INT NOT NULL, C3 INT NOT NULL y C4 INT, el método N (F2, F1, F4, F3) es una petición válida, mientras que el método N (F2, F1) no es válido. Este método sólo se puede utilizar con archivos de tipo IXF o CURSOR.

P Especifica los números de campo (numerados a partir de 1) de los campos de datos de entrada que se van a cargar. Cada columna de tabla que no sea anulable debe tener una entrada correspondiente en la lista METHOD P. Por ejemplo, dados los campos de datos F1, F2, F3, F4, F5 y F6 y las columnas de tabla C1 INT, C2 INT NOT NULL, C3 INT NOT NULL y C4 INT, el método P (2, 1, 4, 3) es

una petición válida, mientras que el método P (2, 1) no es válido. Este método sólo se puede utilizar con los tipos de archivo IXF, DEL o CURSOR, y es el único método válido para el tipo de archivo DEL.

XML FROM *vía-xml*

Especifica una o más vías de acceso que contienen los archivos XML. Los XDS están incluidos en el archivo de datos principal (ASC, DEL, o IXF), en la columna que se cargará en la columna XML.

XMLPARSE

Especifica cómo se analizan los documentos XML. Si no se especifica esta opción, el comportamiento de análisis de los documentos XML vendrá determinado por el valor del registro especial CURRENT XMLPARSE OPTION.

STRIP WHITESPACE

Especifica que hay que eliminar el espacio en blanco al analizar el documento XML.

PRESERVE WHITESPACE

Especifica que no hay que eliminar el espacio en blanco al analizar el documento XML.

XMLVALIDATE

Especifica que los documentos XML se validan con respecto a un esquema, cuando sea pertinente.

USING XDS

Los documentos XML se validan con respecto al esquema XML identificado por el especificador de datos XML (XDS) en el documento de datos principal. Por omisión, si se invoca la opción XMLVALIDATE con la cláusula USING XDS, el esquema que se utiliza para realizar la validación estará determinado por el atributo SCH del XDS. Si no existe un atributo SCH en el XDS, no se producirá ningún esquema de validación a menos que la cláusula DEFAULT especifique un esquema por omisión.

Las cláusulas DEFAULT, IGNORE y MAP se pueden utilizar para modificar el comportamiento de determinación del esquema. Estas tres cláusulas opcionales se aplican directamente a las especificaciones del XDS, no entre ellas. Por ejemplo, si se selecciona un esquema porque está especificado por la cláusula DEFAULT, no se omitirá si también está especificado por la cláusula IGNORE. Asimismo, si se selecciona un esquema porque se ha especificado como la primera parte de un par en la cláusula MAP, no se volverá a correlacionar si también se ha especificado en la segunda parte de otro par de cláusulas MAP.

USING SCHEMA *sqlid-esquema*

Los documentos XML se validan con respecto al esquema XML que tiene el identificador SQL especificado. En este caso, el atributo SCH del especificador de datos XML (XDS) se ignorará para todas las columnas XML.

USING SCHEMALOCATION HINTS

Los documentos XML se validan con respecto a los esquemas identificados por las sugerencias de ubicación de esquemas XML en los documentos XML fuente. Si no se encuentra un atributo schemaLocation en el documento XML, no se producirá ninguna

validación. Cuando se especifica la cláusula USING SCHEMALOCATION HINTS, el atributo SCH del Especificador de datos XML (XDS) se omitirá para todas las columnas XML.

Vea los ejemplos de la opción XMLVALIDATE que figuran más abajo.

IGNORE *sqlid-esquema*

Esta opción sólo se puede utilizar cuando se especifica el parámetro USING XDS. La cláusula IGNORE especifica una lista de uno o más esquemas que hay que ignorar si están identificados mediante un atributo SCH. Si existe un atributo SCH en el especificador de datos XML (XDS) de un documento XML cargado, y el esquema identificado mediante el atributo SCH está incluido en la lista de esquemas que se deben ignorar (IGNORE), no se producirá la validación de esquemas para el documento XML cargado.

Nota:

Si se especifica un esquema en la cláusula IGNORE, tampoco puede estar presente en el lado izquierdo de un par de esquemas en la cláusula MAP.

La cláusula IGNORE sólo se aplica al XDS. Un esquema que está correlacionado por la cláusula MAP no se omitirá posteriormente si se ha especificado en la cláusula IGNORE.

DEFAULT *sqlid-esquema*

Esta opción sólo se puede utilizar cuando se especifica el parámetro USING XDS. El esquema especificado mediante la cláusula DEFAULT identifica un esquema que se debe utilizar para la validación cuando el Especificador de datos XML (XDS) de un documento XML cargado no contiene un atributo SCH que identifique un esquema XML.

La cláusula DEFAULT tiene prioridad sobre las cláusulas IGNORE y MAP. Si un XDS satisface la cláusula DEFAULT, se omitirán las especificaciones IGNORE y MAP.

MAP *sqlid-esquema*

Esta opción sólo se puede utilizar cuando se especifica el parámetro USING XDS. Utilice la cláusula MAP para especificar esquemas alternativos que deben utilizarse en lugar de los especificados por el atributo SCH de un Especificador de datos XML (XDS) para cada documento XML cargado. La cláusula MAP especifica una lista de uno o más pares de esquemas, donde cada par representa una correlación de un esquema con otro. El primer esquema del par representa un esquema al que hace referencia un atributo SCH de un XDS. El segundo esquema del par representa el esquema que debe utilizarse para llevar a cabo la validación de esquemas.

Si un esquema está presente en el lado izquierdo de un par de esquemas de la cláusula MAP, no puede especificarse también en la cláusula IGNORE.

Una vez aplicada una correlación de par de esquemas, el resultado es final. La operación de correlación no es transitiva y, por lo tanto, el esquema elegido no se aplicará ulteriormente a otra correlación de par de esquemas.

Un esquema no se puede correlacionar más de una vez, es decir, no puede aparecer en la parte izquierda de más de un par.

SAVECOUNT *n*

Especifica que el programa de utilidad de carga debe establecer puntos de coherencia después de cada *n* filas. Este valor se convierte en una cuenta de páginas y se redondea por exceso a los intervalos del tamaño de extensión. Dado que se emite un mensaje en cada punto de coherencia, se deberá seleccionar esta opción si la operación de carga se supervisará utilizando LOAD QUERY. Si el valor de *n* no es suficientemente alto, la sincronización de las actividades realizadas en cada punto de coherencia influirá en el rendimiento.

El valor por omisión es cero, que significa que no se establecerán puntos de coherencia, a menos que sean necesarios.

Esta opción no se tiene en cuenta cuando se especifica junto con el tipo de archivo CURSOR o cuando se carga una tabla que contiene una columna XML.

ROWCOUNT *n*

Especifica el número de *n* registros físicos del archivo que se deben cargar. Permite a un usuario cargar solamente las primeras *n* filas de un archivo.

WARNINGCOUNT *n*

Detiene la operación de carga después de *n* avisos. Establezca este parámetro si no se esperan avisos, pero se desea que se verifique si se están utilizando el archivo y la tabla correctos. Si el archivo de carga o la tabla de destino se especifican de modo incorrecto, el programa de utilidad de carga generará un aviso por cada fila que intente cargar, lo que hará que la carga falle. Si *n* es cero o no se especifica esta opción, la operación de carga continuará independientemente del número de avisos emitidos. Si la operación de carga se detiene porque se ha alcanzado el umbral de avisos, se puede iniciar otra operación de carga en modalidad RESTART. La operación de carga continuará automáticamente desde el último punto de coherencia. Alternativamente, se puede iniciar otra operación de carga en modalidad REPLACE, empezando al principio del archivo de entrada.

MESSAGES ON SERVER

Especifica que hay que guardar el archivo de mensajes creado en el servidor por el mandato LOAD. El conjunto de resultados devuelto incluirá las dos columnas siguientes: MSG_RETRIEVAL, que es la sentencia de SQL necesaria para recuperar todos los mensajes de aviso y de error producidos durante esta operación, y MSG_REMOVAL, que es la sentencia de SQL necesaria para hacer limpieza de los mensajes.

Si no se especifica esta cláusula, el archivo de mensajes se suprimirá cuando el procedimiento ADMIN_CMD retorne al llamador. La columna MSG_RETRIEVAL y la columna MSG_REMOVAL del conjunto de resultados contendrán valores nulos.

Observe que, con cláusula o sin ella, el ID de usuario delimitado debe tener autorización para crear archivos bajo el directorio indicado por la variable de registro DB2_UTIL_MSGPATH.

TEMPFILES PATH *nombrevía-temp*

Especifica el nombre de la vía de acceso a utilizar al crear archivos temporales durante una operación de carga y debe estar totalmente calificada de acuerdo a la partición de base de datos del servidor.

Los archivos temporales ocupan espacio del sistema de archivos. A veces, este requisito de espacio es bastante importante. A continuación se

proporciona una estimación de la cantidad de espacio del sistema de archivos que se deberá asignar para todos los archivos temporales:

- 136 bytes para cada mensaje que genere el programa de utilidad de carga
- 15 KB de actividad general si el archivo de datos contiene datos de campo largo o LOB. Esta cantidad puede aumentar significativamente si se especifica la opción INSERT y hay una gran cantidad de datos LOB o de campo largo en la tabla.

INSERT

Una de cuatro modalidades bajo las que se puede ejecutar el programa de utilidad de carga. Añade los datos cargados a la tabla sin cambiar los datos de tabla existentes.

REPLACE

Una de cuatro modalidades bajo las que se puede ejecutar el programa de utilidad de carga. Suprime todos los datos existentes de la tabla e inserta los datos cargados. La definición de tabla y las definiciones de índice no cambian. Si se utiliza esta opción al mover datos entre jerarquías, sólo se pueden sustituir los datos de una jerarquía entera, no de subtablas individuales.

KEEPDICTIONARY

Se conserva un diccionario de compresión existente a través de la operación LOAD REPLACE. Siempre que el atributo COMPRESS de la tabla sea YES, los datos sustituidos recientemente estarán sujetos a la compresión utilizando el diccionario que existía antes de invocar la carga. Si anteriormente en la tabla no existía ningún diccionario, se crea uno nuevo utilizando los datos que se sustituyen en la tabla siempre y cuando el atributo COMPRESS de la tabla sea YES. La cantidad de datos que son necesarios para crear el diccionario de compresión en este caso está sujeta a las políticas de ADC. Estos datos se llenan en la tabla como descomprimidos. Cuando el diccionario se haya insertado en la tabla, los datos restantes que se van a cargar estarán sujetos a la compresión con este diccionario. Este es el parámetro por omisión. Para obtener un resumen, vea la Tabla 1 que figura a continuación.

El ejemplo siguiente conserva el diccionario antiguo si está actualmente en la tabla:

```
CALL SYSPROC.ADMIN_CMD('load from staff.del of del replace
keepdictionary into SAMPLE.STAFF statistics use profile
data buffer 8')
```

Tabla 49. LOAD REPLACE KEEPDICTIONARY

| Comprimir | Existe el diccionario de datos de fila de tabla | Existe el diccionario de objetos de almacenamiento XML ¹ | Diccionario de compresión | Compresión de datos |
|-----------|---|---|---|---|
| YES | YES | YES | Conserva los diccionarios XML de datos de fila de la tabla. | Los datos que se cargarán están sujetos a una compresión. |

Tabla 49. LOAD REPLACE KEEPDICTIONARY (continuación)

| Comprimir | Existe el diccionario de datos de fila de tabla | Existe el diccionario de objetos de almacenamiento XML ¹ | Diccionario de compresión | Compresión de datos |
|-----------|---|---|--|--|
| YES | YES | NO | Conserva el diccionario de datos de fila de tabla y genera un nuevo diccionario XML. | Los datos de fila de la tabla que se cargarán están sujetos a compresión. Después de generar el diccionario XML, los datos XML restantes que se cargarán están sujetos a compresión. |
| YES | NO | YES | Genera un diccionario de datos de fila de tabla y conserva el diccionario XML. | Después de la generación de los datos de fila de la tabla, los datos de fila de la tabla restantes que deben cargarse están sujetos a compresión. Los datos XML que deben cargarse están sujetos a compresión. |
| YES | NO | NO | Genera los diccionarios XML y de datos de fila de la tabla nuevos. | Después de generar los diccionarios, los datos restantes que deben cargarse están sujetos a compresión. |
| NO | YES | YES | Conserva los diccionarios XML de datos de fila de la tabla. | Los datos que deben cargarse no se comprimen. |
| NO | YES | NO | Conserva el diccionario de datos de fila de tabla. | Los datos que deben cargarse no se comprimen. |
| NO | NO | YES | Sin efecto en el diccionario de fila de la tabla. Conserva el diccionario XML. | Los datos que deben cargarse no se comprimen. |
| NO | NO | NO | Sin efecto. | Los datos que deben cargarse no se comprimen. |

Nota:

1. Se puede crear un diccionario de compresión para el objeto de almacenamiento XML de una tabla sólo si las columnas XML se añaden a la tabla en DB2 Versión 9.7 o una versión posterior o si la tabla se migra mediante un movimiento de tabla en línea.

RESETDICTIONARY

Esta directiva da instrucciones al proceso LOAD REPLACE para crear un diccionario nuevo para el objeto de datos de la tabla siempre que el atributo COMPRESS de la tabla sea YES. Si el atributo de COMPRESS es NO y ya existía un diccionario en la tabla, se eliminará y no se insertará ningún diccionario nuevo en la tabla. Se puede crear un diccionario de compresión con tan sólo un registro de usuario. Si el tamaño del conjunto de datos cargados es cero y ya existe un diccionario, el diccionario no se conservará. La cantidad de datos necesarios para crear un diccionario con esta directiva no está sujeta a las políticas de ADC. Para obtener un resumen, vea la Tabla 2 que figura a continuación.

El siguiente ejemplo restablecerá el diccionario actual y creará uno nuevo:

```
CALL SYSPROC.ADMIN_CMD('load from staff.del of del replace
resetdictionary into SAMPLE.STAFF statistics use profile
data buffer 8')
```

Tabla 50. LOAD REPLACE RESETDICTIONARY

| Comprimir | Existe el diccionario de datos de fila de tabla | Existe el diccionario de objetos de almacenamiento XML ¹ | Diccionario de compresión | Compresión de datos |
|-----------|---|---|---|---|
| YES | YES | YES | Genera nuevos diccionarios ² . Si la opción DATA CAPTURE CHANGES está habilitada en las sentencias CREATE TABLE o ALTER TABLE, se conserva el diccionario de datos de fila de tabla actual (y se hace referencia a él como <i>diccionario de compresión histórica</i>). | Después de generar los diccionarios, los datos restantes que deben cargarse están sujetos a compresión. |
| YES | YES | NO | Genera nuevos diccionarios ² . Si la opción DATA CAPTURE CHANGES está habilitada en las sentencias CREATE TABLE o ALTER TABLE, se conserva el diccionario de datos de fila de tabla actual (y se hace referencia a él como <i>diccionario de compresión histórica</i>). | Después de generar los diccionarios, los datos restantes que deben cargarse están sujetos a compresión. |
| YES | NO | YES | Genera nuevos diccionarios. | Después de generar los diccionarios, los datos restantes que deben cargarse están sujetos a compresión. |
| YES | NO | NO | Genera nuevos diccionarios. | Después de generar los diccionarios, los datos restantes que deben cargarse están sujetos a compresión. |
| NO | YES | YES | Elimina diccionarios. | Los datos que deben cargarse no se comprimen. |
| NO | YES | NO | Elimina el diccionario de datos de fila de tabla. | Los datos que deben cargarse no se comprimen. |
| NO | NO | YES | Elimina el diccionario de objetos de almacenamiento XML. | Los datos que deben cargarse no se comprimen. |
| NO | NO | NO | Sin efecto. | Ningún dato de la tabla se comprime. |

Notas:

1. Se puede crear un diccionario de compresión para el objeto de almacenamiento XML de una tabla sólo si las columnas XML se añaden a la tabla en DB2 Versión 9.7 o una versión posterior o si la tabla se migra mediante un movimiento de tabla en línea.
2. Si hay un diccionario y se habilita el atributo de compresión, pero no hay registros que cargar en la partición de tabla, no se puede generar un nuevo diccionario y la operación RESETDICTIONARY no conservará el diccionario existente.

TERMINATE

Una de cuatro modalidades bajo las que se puede ejecutar el programa de utilidad de carga. Termina una operación de carga interrumpida anteriormente y retrotrae la operación hasta el momento en que se empezó, incluso si se habían pasado puntos de coherencia. Los estados de los espacios de tablas implicados en la operación vuelven a ser normales y todos los objetos de tabla pasan a estar coherentes (puede que los objetos de índice se marquen como no válidos, en cuyo caso tendrá lugar automáticamente una reconstrucción de índice en el siguiente acceso). Si la operación de carga que se está terminando es una operación LOAD REPLACE, la tabla se truncará a una tabla vacía después de la operación LOAD TERMINATE. Si la operación de carga que se está terminando es una operación LOAD INSERT, la tabla retendrá todos los registros originales después de la operación LOAD TERMINATE. Para obtener un resumen de la gestión de diccionarios, vea la Tabla 3 que figura a continuación.

La opción LOAD TERMINATE no eliminará un estado de copia de seguridad pendiente de los espacios de tablas.

RESTART

Una de cuatro modalidades bajo las que se puede ejecutar el programa de utilidad de carga. Reinicia una operación de carga interrumpida anteriormente. La operación de carga continuará automáticamente desde el último punto de coherencia de la fase de carga, creación o supresión. Para obtener un resumen de la gestión de diccionarios, vea la Tabla 4 que se muestra a continuación.

INTO *nombre-tabla*

Especifica la tabla de base de datos en la que deben cargarse los datos. Esta tabla no puede ser una tabla de sistema, una tabla temporal declarada o una tabla temporal creada. Se puede especificar un alias o el nombre de tabla totalmente calificado o no calificado. Un nombre de tabla calificado tiene el formato esquema.nombretabla. Si se especifica un nombre de tabla no calificado, la tabla se calificará con CURRENT SCHEMA.

columna-inserción

Especifica la columna de tabla en la que deben insertarse los datos.

El programa de utilidad de carga no puede analizar columnas cuyos nombres contienen uno o más espacios. Por ejemplo,

```
CALL SYSPROC.ADMIN_CMD('load from delfile1 of del noheader
method P (1, 2, 3, 4, 5, 6, 7, 8, 9)
insert into table1 (BLOB1, S2, I3, Int 4, I5, I6, DT7, I8, TM9)')
```

fallará debido a la columna Int 4. La solución consiste en escribir entre comillas dobles dichos nombres de columna:

```
CALL SYSPROC.ADMIN_CMD('load from delfile1 of del noheader
method P (1, 2, 3, 4, 5, 6, 7, 8, 9)
insert into table1 (BLOB1, S2, I3, "Int 4", I5, I6, DT7, I8, TM9)')
```

FOR EXCEPTION *nombre-tabla*

Especifica la tabla de excepción en la que se copiarán las filas erróneas. Se copiará cualquier fila que esté violando un índice exclusivo o un índice de claves primarias. Si se especifica un nombre de tabla no calificado, la tabla se calificará con CURRENT SCHEMA.

La información que se graba en la tabla de excepción *no* se graba en el archivo de vuelco. En un entorno de bases de datos particionadas, se debe definir una tabla de excepción para las particiones de bases de datos en las que se ha definido la tabla de carga. De lo contrario, el archivo de vuelcos contiene filas que no se pueden cargar porque no son válidas o contienen errores de sintaxis.

Cuando se cargan datos XML, no se da soporte a la utilización de la cláusula FOR EXCEPTION para especificar la carga de una tabla de excepciones en los casos siguientes:

- Cuando se utilice el control de acceso basado en etiquetas (LBAC).
- Cuando se carguen datos en una tabla particionada de datos.

NORANGEEXC

Indica que si se rechaza una fila debido a una violación del rango, no se insertará en la tabla de excepción.

NOUNIQUEEXC

Indica que si se rechaza una fila porque viola una restricción exclusiva, no se insertará en la tabla de excepción.

STATISTICS USE PROFILE

Indica a la carga que recopile estadísticas durante la carga de acuerdo con el perfil definido para esta tabla. Este perfil se debe crear antes de ejecutar la carga. El perfil se crea con el mandato RUNSTATS. Si el perfil no existe y se indica a la carga que debe recopilar estadísticas de acuerdo con el perfil, se devolverá un aviso y no se recopilarán estadísticas.

STATISTICS NO

Especifica que no se deben reunir estadísticas y que las estadísticas de los catálogos no se deben modificar. Es el valor por omisión.

COPY NO

Especifica que el espacio de tablas en el que reside la tabla se colocará en estado de pendiente de copia de seguridad si se habilita la recuperación en avance (es decir, está activado *logretain* o *userexit*). La opción COPY NO también pondrá el estado del espacio de tablas en el estado de espacio de tablas Carga en curso. Éste es un estado transitorio que desaparecerá cuando la carga se complete o termine anormalmente. Los datos de cualquier tabla del espacio de tablas no se pueden actualizar o suprimir hasta que se realice una copia de seguridad del espacio de tablas o una copia de seguridad de la base de datos completa. Sin embargo, es posible acceder a los datos de cualquier tabla utilizando la sentencia SELECT.

LOAD con COPY NO en una base de datos recuperable deja los espacios de tablas en estado pendiente de copia de seguridad. Por ejemplo, ejecutar una opción LOAD con COPY NO y INDEXING MODE DEFERRED dejará los índices con necesidad de renovación. Ciertas consultas en la tabla podrían requerir una exploración de índice y no serán satisfactorias hasta que se renueven los índices. El índice no puede renovarse si reside en un

espacio de tablas que se encuentre en estado de pendiente de copia de seguridad. En ese caso, no se permitirá acceso a la tabla hasta que se realice una copia de seguridad. La base de datos efectúa la renovación de los índices automáticamente cuando una consulta accede al índice. Si no se especifica una de las opciones COPY NO, COPY YES o NONRECOVERABLE, y la base de datos es recuperable (**logretain** o **logarchmeth1** está habilitado), la opción COPY NO es el valor por omisión.

COPY YES

Especifica que se guardará una copia de los datos cargados. Esta opción no es válida si la recuperación en avance está inhabilitada.

USE TSM

Especifica que la copia se almacenará utilizando Tivoli Storage Manager (TSM).

OPEN *núm-sesiones* **SESSIONS**

Número de sesiones de E/S que deben utilizarse con TSM o el producto del proveedor. El valor por omisión es 1.

TO *dispositivo/directorio*

Especifica el dispositivo o el directorio en el que se creará la imagen de copia.

LOAD *nombre-bibl*

El nombre de la biblioteca compartida (DLL en sistemas operativos Windows) que contiene las funciones de E/S de copia de seguridad y restauración del proveedor que se debe utilizar. Puede contener la vía de acceso completa. Si no se proporciona la vía de acceso completa, se tomará por omisión la vía de acceso donde residen los programas de salida de usuario.

NONRECOVERABLE

Especifica que la transacción de carga debe marcarse como no recuperable y que no será posible recuperarla mediante una acción subsiguiente de recuperación en avance. El programa de utilidad de recuperación en avance saltará la transacción y marcará la tabla en la que se estaban cargando datos como "no válida". El programa de utilidad también ignorará las transacciones subsiguientes para la tabla. Después de que se haya completado la operación de recuperación en avance, una tabla de este tipo sólo se puede descartar o restaurar desde una copia de seguridad (completa o de espacio de tablas) realizada después de un punto de confirmación a continuación de la realización de la operación de carga no recuperable.

Con esta opción, los espacios de tablas no se ponen en estado de pendiente de copia de seguridad a continuación de la operación de carga y durante la operación de carga no se tiene que realizar una copia de los datos cargados. Si no se especifica una de las opciones COPY NO, COPY YES, o NONRECOVERABLE y la base de datos no es recuperable (**logretain** o **logarchmeth1** no está habilitado), NONRECOVERABLE es el valor por omisión.

WITHOUT PROMPTING

Especifica que la lista de archivos de datos contiene todos los archivos que se deben cargar y que los dispositivos o directorios listados son suficientes para la operación de carga entera. Si no se encuentra un archivo de entrada de continuación o los destinos de copia se llenan antes de que finalice la operación de carga, la operación de carga fallará y la tabla permanecerá en estado de pendiente de carga.

Es el valor por omisión. Las acciones que normalmente requieren la intervención del usuario devolverán un mensaje de error.

DATA BUFFER *tamaño-almacint*

Especifica el número de páginas de 4 KB (independientemente del grado de paralelismo) que se deben utilizar como espacio de almacenamiento intermedio para transferir datos dentro del programa de utilidad. Si el valor especificado es menor que el mínimo algorítmico, se utilizará el recurso mínimo necesario y no se devolverá ningún aviso.

Esta memoria se asigna directamente desde la pila del programa de utilidad, cuyo tamaño puede modificarse mediante el parámetro de configuración de base de datos *util_heap_sz*.

Si no se especifica un valor, el programa de utilidad calcula un valor por omisión inteligente en la ejecución. El valor por omisión se basa en un porcentaje del espacio libre disponible en la pila del programa de utilidad en el tiempo de creación de instancias del cargador, así como en algunas características de la tabla.

SORT BUFFER *tamaño-almacint*

Esta opción especifica un valor que prevalece sobre el parámetro de configuración de base de datos SORTHEAP durante una operación de carga. Sólo es relevante al cargar tablas con índices y sólo cuando el parámetro INDEXING MODE no se especifica como DEFERRED. El valor especificado no puede exceder el valor de SORTHEAP. Este parámetro es útil para acelerar la memoria de clasificación que se utiliza al cargar tablas con muchos índices sin cambiar el valor de SORTHEAP, que afectaría al proceso general de consulta.

CPU_PARALLELISM *n*

Especifica el número de procesos o hebras que el programa de utilidad creará para analizar, convertir y formatear registros al crear objetos de tabla. Este parámetro está diseñado para aprovechar el número de procesos que se ejecutan por partición de base de datos. Es especialmente útil al cargar datos clasificados previamente, porque se conserva el orden de registro de los datos fuente. Si el valor de este parámetro es cero o no se ha especificado, el programa de utilidad de carga utiliza un valor inteligente por omisión (generalmente basado en el número de CPU disponibles) en la ejecución.

Nota:

1. Si se utiliza este parámetro con tablas que contienen campos LOB o LONG VARCHAR, su valor se convierte en uno, independientemente del número de CPU de sistema o del valor especificado por el usuario.
2. Si se especifica un valor pequeño para el parámetro SAVECOUNT, el cargador realiza muchas más operaciones de E/S para desechar datos y metadatos de tabla. Cuando CPU_PARALLELISM es mayor que uno, las operaciones de desecho son asíncronas, permitiendo al cargador aprovechar la CPU. Cuando CPU_PARALLELISM se establece en uno, el cargador espera en la E/S durante los puntos de coherencia. Una operación de carga con CPU_PARALLELISM establecido en dos y SAVECOUNT establecido en 10 000 se completa mucho más rápidamente que la misma operación con CPU_PARALLELISM establecido en uno, aunque sólo haya una CPU.

DISK_PARALLELISM *n*

Especifica el número de procesos o hebras que el programa de utilidad de

carga creará para grabar datos en los contenedores del espacio de tablas. Si no se especifica un valor, el programa de utilidad selecciona un valor por omisión inteligente basándose en el número de contenedores de espacios de tablas y en las características de la tabla.

FETCH_PARALLELISM YES | NO

Cuando se realiza una carga desde un cursor en el que se ha declarado el cursor mediante la palabra clave DATABASE o cuando se utiliza la entrada de soporte de almacenamiento `sqlu_remotefetch_entry` de la API y esta opción tiene el valor YES, el programa de utilidad de carga intentará paralelizar la captación desde la fuente de datos remota si es posible. Si se establece en NO, no se realizará ninguna captación en paralelo. El valor por omisión es YES. Para obtener más información, consulte *Mover datos utilizando el tipo de archivo CURSOR*.

INDEXING MODE

Especifica si el programa de utilidad de carga debe volver a crear índices o si debe ampliarlos de forma incremental. Los valores válidos son:

AUTOSELECT

El programa de utilidad de carga decidirá automáticamente entre la modalidad REBUILD o INCREMENTAL. La decisión se basa en la cantidad de datos que se cargan y en la profundidad del árbol de índice. La información relacionada con la profundidad del árbol de índice se almacena en el objeto de índice. RUNSTATS no es necesario para rellenar esta información. AUTOSELECT es la modalidad de indexación por omisión.

REBUILD

Se volverán a crear todos los índices. El programa de utilidad debe tener suficientes recursos para clasificar todas las partes de claves de índice para los datos de tabla antiguos y añadidos.

INCREMENTAL

Se ampliarán los índices con datos nuevos. Esta propuesta consume espacio libre de índice. Sólo necesita suficiente espacio de clasificación para añadir claves de índice para los registros insertados. Este método sólo se soporta en los casos en los que el objeto de índice es válido y accesible al principio de una operación de carga (por ejemplo, no es válido inmediatamente después de una operación de carga en la que se ha especificado la modalidad DEFERRED). Si se especifica esta modalidad, pero no se soporta debido al estado del índice, se devuelve un aviso y la operación de carga continúa en modalidad REBUILD. Asimismo, si se empieza una operación de reinicio de carga en la fase de creación de carga, no se soporta la modalidad INCREMENTAL.

La creación de índices incremental no se soporta cuando se cumplen todas las condiciones siguientes:

- Se especifica la opción LOAD COPY (*logarchmeth1* con la opción USEREXIT o LOGRETAIN).
- La tabla reside en un espacio de tablas DMS.
- El objeto de índice reside en un espacio de tablas que está compartido por otros objetos de tabla que pertenecen a la tabla que se está cargando.

Para ignorar esta restricción, se recomienda colocar los índices en un espacio de tablas independiente.

DEFERRED

El programa de utilidad de carga no intentará la creación de índices si se especifica esta modalidad. Los índices se marcarán como que necesitan una renovación. El primer acceso a tales índices que no esté relacionado con una operación de carga podría forzar una reconstrucción, o puede que se vuelvan a construir los índices cuando se reinicie la base de datos. Este planteamiento necesita suficiente espacio de clasificación para todas las partes clave del índice más grande. El tiempo total empleado posteriormente para la construcción de índices es mayor que el necesario en la modalidad REBUILD. Por consiguiente, al efectuar múltiples operaciones de carga con creación de índices diferida, es aconsejable (desde un punto de vista de rendimiento) dejar que la última operación de carga de la secuencia realice una reconstrucción de índices, en lugar de dejar que los índices se vuelvan a crear en el primero acceso que no sea de carga.

La creación de índices diferida sólo se soporta para tablas con índices no exclusivos, de modo que las claves duplicadas insertadas durante la fase de carga no son permanentes después de la operación de carga.

ALLOW NO ACCESS

La carga bloqueará la tabla de destino para su acceso exclusivo durante la carga. El estado de tabla se establecerá en Carga en curso durante la carga. ALLOW NO ACCESS es el comportamiento por omisión. Es la única opción válida para LOAD REPLACE.

Cuando la tabla tiene restricciones, el estado de ésta se establecerá en Pendiente de establecimiento de integridad, así como en Carga en proceso. La sentencia SET INTEGRITY se debe utilizar para sacar a la tabla del estado Pendiente de establecimiento de integridad.

ALLOW READ ACCESS

La carga bloqueará la tabla de destino en modalidad de compartimiento. El estado de la tabla se establecerá Carga en curso y Acceso de lectura. Los lectores pueden acceder a la parte no delta de los datos mientras se carga la tabla. En otras palabras, los lectores de la tabla podrán acceder a los datos que existían antes del inicio de la carga y los datos que se están cargando no estarán disponibles hasta que se haya completado la carga. LOAD TERMINATE o LOAD RESTART de una carga ALLOW READ ACCESS pueden utilizar esta opción; LOAD TERMINATE o LOAD RESTART de una carga ALLOW NO ACCESS no pueden utilizar esta opción. Además, esta opción no es válida si los índices de la tabla de destino están marcados como que necesitan reconstrucción.

Cuando haya restricciones en la tabla, el estado de la tabla se establecerá en Pendiente de establecimiento de integridad, así como Carga en proceso y Acceso de lectura. Al final de la carga, el estado de tabla Carga en proceso se eliminará, pero los estados de tabla Pendiente de establecimiento de integridad y Acceso de lectura permanecerán. La sentencia SET INTEGRITY se debe utilizar para sacar a la tabla del estado Pendiente de establecimiento de integridad. Mientras la tabla está en los estados Pendiente de establecer integridad y Acceso de lectura, la porción no delta de los datos sigue siendo accesible para los lectores y la nueva porción (delta) de los datos seguirá siendo inaccesible hasta que se haya completado la sentencia SET INTEGRITY. Un usuario puede realizar varias cargas de la misma tabla sin emitir una sentencia SET INTEGRITY. Sin

embargo, sólo los datos originales (comprobados) seguirán visibles hasta que se emita la sentencia SET INTEGRITY.

ALLOW READ ACCESS también da soporte a los modificadores siguientes:

USE *nombre-espacio-tablas*

Si se vuelven a crear los índices, se crea una copia duplicada del índice en el espacio de tablas *nombre-espacio-tablas* y se copia en el espacio de tablas original al final de la carga durante una operación INDEX COPY PHASE. Con esta opción sólo se pueden utilizar espacios de tablas temporales del sistema. Si entonces no se especifica, el índice duplicado se creará en el mismo espacio de tablas que el objeto de índice. Si la copia duplicada se crea en el espacio de tablas como objeto de índice, la copia del objeto de índice duplicado sobre el antiguo objeto de índice es instantánea. Si la copia duplicada está en un espacio de tablas diferente del objeto de índice, se realiza una copia física. Esto podría implicar una E/S y tiempo considerables. La copia se realiza mientras la tabla está fuera de línea al final de una carga durante la operación INDEX COPY PHASE.

Sin esta opción, el índice duplicado se construye en el mismo espacio de tablas que el original. Dado que tanto el índice original como el duplicado por omisión residen simultáneamente en el mismo espacio de tablas, es posible que no haya espacio suficiente para mantener ambos índices en un mismo espacio de tablas. El uso de esta opción asegura que se retiene suficiente espacio de tablas para los índices.

Esta opción se omite si el usuario no especifica INDEXING MODE REBUILD o INDEXING MODE AUTOSELECT. Esta opción también se omitirá si se elige INDEXING MODE AUTOSELECT y la carga decide actualizar de forma incremental el índice.

SET INTEGRITY PENDING CASCADE

Si LOAD pone la tabla en el estado Pendiente de establecimiento de integridad, la opción SET INTEGRITY PENDING CASCADE permite al usuario especificar si el estado Pendiente de establecimiento de integridad de la tabla cargada se pondrá inmediatamente en cascada en todos los descendientes (incluidas las tablas de claves foráneas descendientes, las tablas de consulta materializada inmediatamente descendientes y las tablas de etapas inmediatas descendientes).

IMMEDIATE

Indica que el estado Pendiente de establecimiento de integridad se extiende inmediatamente a todas las tablas de claves foráneas descendientes, las tablas de consulta materializada inmediatamente y las tablas de etapas descendientes. En el caso de una operación LOAD INSERT, el estado Pendiente de establecimiento de integridad no se extiende a las tablas de claves foráneas descendientes, incluso aunque se especifique la opción IMMEDIATE.

Cuando la tabla cargada se comprueba posteriormente para detectar posibles violaciones de restricciones (mediante la opción IMMEDIATE CHECKED de la sentencia SET INTEGRITY), las tablas de claves foráneas descendientes que se pusieron en el

estado Establecer integridad pendiente con acceso de lectura se pondrán en el estado Sin acceso Pendiente de establecimiento de integridad.

DEFERRED

Indica que sólo la tabla cargada se pondrá en el estado Pendiente de establecimiento de integridad. Los estados de las tablas de clave foránea descendientes, las tablas de consulta materializadas inmediatas descendientes y las tablas por fases inmediatas descendientes seguirán sin cambios.

Posteriormente, las tablas de claves foráneas descendientes pueden colocarse implícitamente en el estado Pendiente de establecimiento de integridad cuando sus tablas padre se comprueban para detectar posibles violaciones de restricciones (mediante la opción IMMEDIATE CHECKED de la sentencia SET INTEGRITY). Las tablas de consulta materializada inmediatas descendientes y las tablas de etapas inmediatas descendientes se pondrán implícitamente en estado Pendiente de establecimiento de integridad cuando se compruebe la existencia de posibles violaciones de la integridad en una de sus tablas subyacentes. Se emitirá un aviso (SQLSTATE 01586) para indicar que las tablas dependientes se han colocado en estado Pendiente de establecimiento de integridad. Consulte las notas de la sentencia SET INTEGRITY en la publicación Consulta de SQL para averiguar cuándo se pondrán las tablas descendientes en el estado Pendiente de establecimiento de integridad.

Si no se especifica la opción SET INTEGRITY PENDING CASCADE:

- Sólo la tabla cargada se pondrá en el estado Pendiente de establecimiento de integridad. El estado de las tablas de claves foráneas descendientes, las tablas de consulta materializada inmediatas descendientes y las tablas de etapas inmediatas descendientes no cambiará y, posteriormente, se podrá poner implícitamente en el estado Pendiente de establecimiento de integridad cuando se compruebe si la tabla cargada contiene posibles violaciones de restricciones.

Si LOAD no pone la tabla de destino en el estado Pendiente de establecimiento de integridad, no se tendrá en cuenta la opción SET INTEGRITY PENDING CASCADE.

LOCK WITH FORCE

El programa de utilidad adquiere varios bloqueos, incluidos los bloqueos de tabla en proceso de carga. En lugar de esperar, y posiblemente exceder el tiempo de espera, al adquirir un bloqueo, esta opción permite que la carga fuerce otras aplicaciones que retienen bloqueos que están en conflicto en la tabla de destino. El programa de utilidad de carga no forzará las aplicaciones que retienen bloqueos que están en conflicto en las tablas de catálogos del sistema. Las aplicaciones forzadas se retrotraerán y se liberarán los bloqueos que necesita el programa de utilidad de carga. Entonces, el programa de utilidad de carga podrá continuar. Esta opción requiere la misma autorización que el mandato FORCE APPLICATIONS (SYSADM o SYSCTRL).

Las cargas ALLOW NO ACCESS pueden forzar las aplicaciones que contienen bloqueos en conflicto al principio de la operación de carga. Al inicio de la carga, el programa de utilidad puede forzar aplicaciones que intentan consultar o modificar la tabla.

Las cargas ALLOW READ ACCESS pueden forzar las aplicaciones que contienen bloqueos en conflicto al comienzo o al final de la operación de carga. Al inicio de la carga, el programa de utilidad de carga puede forzar aplicaciones que intentan modificar la tabla. Al final de la operación de carga, el programa de utilidad de carga puede forzar aplicaciones que intentan consultar o modificar la tabla.

SOURCEUSEREXIT*ejecutable*

Especifica un nombre de archivo ejecutable que se invocará para alimentar datos en el programa de utilidad.

REDIRECT

INPUT FROM

BUFFER *almacenamiento-intermedio-entrada*

La secuencia de bytes especificada en *almacenamiento-intermedio-entrada* se transfiere al descriptor de archivo STDIN del proceso que ejecuta el ejecutable suministrado.

FILE *archivo-entrada*

El contenido de este archivo en el extremo cliente se transfiere al descriptor de archivo STDIN del proceso que ejecuta el ejecutable suministrado.

OUTPUT TO

FILE *archivo-salida*

Los descriptores de archivo STDOUT y STDERR se capturan en el archivo del lado del servidor totalmente calificado que se ha especificado.

PARALLELIZE

Aumenta el rendimiento de los datos que acceden al programa de utilidad de carga invocando simultáneamente varios procesos de salida de usuario. Esta opción sólo es aplicable en entornos de base de datos de varias particiones y se omite en entornos de base de datos de una sola partición.

Para obtener más información, consulte *Mover datos utilizando una aplicación personalizada (salida de usuario)*.

PARTITIONED DB CONFIG *opción-bd-particionada*

Permite ejecutar una carga en una tabla distribuida en varias particiones de base de datos. El parámetro PARTITIONED DB CONFIG permite especificar opciones de configuración específicas de la base de datos particionada. Los valores de *opción-bd-particionada* pueden ser uno cualquiera de los siguientes:

PART_FILE_LOCATION x
OUTPUT_DBPARTNUMS x
PARTITIONING_DBPARTNUMS x
MODE x
MAX_NUM_PART_AGENTS x
ISOLATE_PART_ERRS x
STATUS_INTERVAL x
PORT_RANGE x
CHECK_TRUNCATION
MAP_FILE_INPUT x
MAP_FILE_OUTPUT x
TRACE x

```

NEWLINE
DISTFILE x
OMIT_HEADER
RUN_STAT_DBPARTNUM x

```

Encontrará descripciones detalladas de estas opciones en *Load configuration options for partitioned database environments*.

RESTARTCOUNT
Reservado.

USING directorio
Reservado.

Ejemplo

Emita una carga con la opción de sustitución de los datos de la tabla de empleados de un archivo.

```

CALL SYSPROC.ADMIN_CMD('LOAD FROM /home/theresax/tmp/emp_exp.dat
  OF DEL METHOD P (1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14)
  MESSAGES /home/theresax/tmp/emp_load.msg
  REPLACE INTO THERESAX.EMPLOYEE (EMPNO, FIRSTNME, MIDINIT, LASTNAME,
  WORKDEPT, PHONENO, HIREDATE, JOB, EDLEVEL, SEX, BIRTHDATE, SALARY,
  BONUS, COMM) COPY NO INDEXING MODE AUTOSELECT ISOLATE_PART_ERRS
  LOAD_ERRS_ONLY MODE PARTITION_AND_LOAD' )

```

A continuación se muestra un ejemplo de salida de una base de datos de partición individual.

Conjunto de resultados 1

```

-----
ROWS_READ      ROWS_SKIPPED    ROWS_LOADED    ROWS_REJECTED  ...
-----
          32              0              32              0 ...

```

1 registro(s) seleccionado(s).

Estado de devolución = 0

Salida de una base de datos de partición individual (continuación).

```

... ROWS_DELETED    ROWS_COMMITTED    MSG_RETRIEVAL
... -----
...              0              32 SELECT SQLCODE, MSG_TEXT FROM
...              TABLE(SYSPROC.ADMIN_GET_MSGS(
...              '2203498_thx')) AS MSG

```

Salida de una base de datos de partición individual (continuación).

```

... MSG_REMOVAL
... -----
... CALL SYSPROC.ADMIN_REMOVE_MSGS('2203498_thx')
...

```

Nota: En este conjunto de resultados también se devuelven las siguientes columnas, pero se establecen en un valor NULL porque sólo se llenan cuando se llenan en una base de datos de múltiples particiones: ROWS_PARTITIONED y NUM_AGENTINFO_ENTRIES.

A continuación se muestra un ejemplo de salida de una base de datos de partición múltiple.

Conjunto de resultados 1

```
-----  
ROWS_READ      ROWS_REJECTED    ROWS_PARTITIONED  NUM_AGENTINFO_ENTRIES ...  
-----  
                32                0                32                5 ...  
                ...                ...                ...                ...  
                ...                ...                ...                ...
```

1 registro(s) seleccionado(s).

Salida de una base de datos de partición múltiple (continuación).

```
... MSG_RETRIEVAL                      MSG_REMOVAL  
... -----  
... SELECT DBPARTITIONNUM, AGENT_TYPE,  CALL SYSPROC.ADMIN_REMOVE_MSGS  
...   SQLCODE, MSG_TEXT FROM TABLE      ('2203498_thx')  
...   (SYSPROC.ADMIN_GET_MSGS  
...   ('2203498_thx')) AS MSG
```

Nota: En este conjunto de resultados también se devuelven las siguientes columnas, pero se establecen en un valor NULL porque sólo se llenan cuando se cargan en una base de datos de partición individual: ROWS_SKIPPED, ROWS_LOADED, ROWS_DELETED y ROWS_COMMITTED.

Salida de una base de datos de partición múltiple (continuación).

Conjunto de resultados 2

```
-----  
DBPARTITIONNUM  SQLCODE  TABSTATE  AGENTTYPE  
-----  
                10          0 NORMAL    LOAD  
                20          0 NORMAL    LOAD  
                30          0 NORMAL    LOAD  
                20          0 NORMAL    PARTITION  
                10          0 NORMAL    PRE_PARTITION
```

1 registro(s) seleccionado(s).

Estado de devolución = 0

Ejemplos de carga de datos de documentos XML

Carga de datos XML

Ejemplo 1

El usuario ha construido un archivo de datos con campos XDS para describir los documentos que se van a insertar en la tabla. Puede parecerse al siguiente:

```
1, "<XDS FIL=""file1.xml"" />"  
2, "<XDS FIL='file2.xml' OFF='23' LEN='45' />"
```

Para la primera fila, el documento XML se identifica mediante el archivo denominado file1.xml. Observe que puesto que el delimitador de carácter es el carácter de comillas dobles, y las comillas dobles están dentro del XDS, las marcas de comillas dobles contenidas en el XDS se duplican. Para la segunda fila, el documento XML se identifica mediante el archivo denominado file2.xml y comienza en el desplazamiento de byte 23, y tiene 45 bytes de longitud.

Ejemplo 2

El usuario emite un mandato de carga sin analizar ni validar opciones para la columna XML y los datos se cargan satisfactoriamente:

```
LOAD FROM data.del of DEL INSERT INTO mytable
```

Carga de datos XML de CURSOR

La carga de datos de cursor es igual que con un tipo de columna relacional habitual. El usuario tiene dos tablas, T1 y T2, cada una de las cuales consiste en una única columna XML denominada C1. Para cargar (LOAD) de T1 a T2, el usuario primero declarará un cursor:

```
DECLARE X1 CURSOR FOR SELECT C1 FROM T1;
```

A continuación, el usuario puede emitir un mandato LOAD utilizando el tipo de cursor:

```
LOAD FROM X1 of CURSOR INSERT INTO T2
```

Aplicar opciones de LOAD específicas de XML al tipo de cursor es igual que cargar desde un archivo.

Notas de uso

- Los datos se cargan en la secuencia que aparecen en el archivo de entrada. Si se desea una secuencia determinada, se deberán clasificar los datos antes de intentar una carga. Si no es necesario conservar el orden de los datos fuente, considere la posibilidad de utilizar el modificador de tipo de archivo *ANYORDER*, que se describe a continuación en la sección *Modificadores de tipo de archivo para el programa de utilidad de carga*.
- El programa de utilidad de carga crea índices basándose en las definiciones existentes. Las tablas de excepción se utilizan para manejar duplicados en claves exclusivas. El programa de utilidad no fuerza la integridad referencial, no realiza ninguna comprobación de restricciones ni actualiza las tablas de consulta materializadas que dependen de las tablas que se están cargando. Las tablas que incluyen restricciones de comprobación o de referencia se ponen en el estado Pendiente de establecimiento de integridad. Las tablas de resumen que se definen con *REFRESH IMMEDIATE* y que dependen de las tablas que se cargan, también se ponen en el estado Pendiente de establecimiento de integridad. Ejecute la sentencia *SET INTEGRITY* para sacar a las tablas del estado Pendiente de establecimiento de integridad. Las operaciones de carga no se pueden llevar a cabo en tablas de consulta materializadas.
- Si se necesita un índice de clúster en la tabla, los datos deben clasificarse en el índice de clúster antes de la carga. Sin embargo, no es necesario clasificar los datos antes de cargarlos en una tabla de clúster multidimensional (MDC).
- Si se especifica una tabla de excepción cuando se carga en una tabla protegida, las filas que están protegidas por etiquetas de seguridad no válidas se enviarán a dicha tabla. Esto puede permitir que los usuarios que tienen acceso a la tabla de excepción accedan a datos a los que normalmente tendrían autorización para acceder. Para una mejor seguridad, tenga cuidado sobre a quién otorga acceso a la tabla de excepción, suprima cada fila en cuanto esté arreglada y copiadas en la tabla que se está cargando y descarte la tabla de excepción en cuanto haya terminado.
- En las etiquetas de seguridad, cuando están en formato interno, puede haber caracteres de nueva línea. Si carga el archivo utilizando el formato de archivo *DEL*, los caracteres de nueva línea se podrían malinterpretar como

delimitadores. Si surge este problema, utilice la prioridad por omisión más antigua para los delimitadores especificando el modificador de tipo de archivo `delprioritychar` en el mandato `LOAD`.

- El programa de utilidad `LOAD` emite una sentencia `COMMIT` al principio de la operación, que en el caso de conexiones de Tipo 2, hace que el procedimiento vuelva a `SQL30090N` con el código de razón 1.
- Las vías de acceso del mandato `LOAD` deben ser una vía de acceso totalmente calificada en la partición del coordinador del servidor.
- Para realizar una carga utilizando el tipo de archivo `CURSOR` donde se ha especificado la palabra clave `DATABASE` durante el mandato `DECLARE CURSOR`, el ID de usuario y la contraseña que se utilizan para autenticarse en la base de datos a la que se está conectado actualmente (para la carga) se utilizarán para autenticarse en la base de datos fuente (especificada por la opción `DATABASE` del mandato `DECLARE CURSOR`). Si no se ha especificado ningún ID de usuario ni contraseña para la conexión con la base de datos de carga, debe especificarse un ID de usuario y una contraseña para la base de datos fuente durante el mandato `DECLARE CURSOR`.
- La carga de un archivo `PC/IXF` con varias partes cuyas partes individuales se copian de un sistema `Windows` a un sistema `AIX` está soportada. Los nombres de todos los archivos deben especificarse en el mandato `LOAD`. Por ejemplo, `LOAD FROM DATA.IXF, DATA.002 OF IXF INSERT INTO TABLE1`. La carga en el sistema operativo `Windows` de archivos `PC/IXF` divididos lógicamente no está soportada.
- Cuando se reinicia un mandato `LOAD` anómalo, el comportamiento irá a continuación del comportamiento existente en el sentido de que la fase `BUILD` se verá forzada a utilizar la modalidad `REBUILD` para los índices.
- No se da soporte a la carga de documentos `XML` entre bases de datos y se devuelve el mensaje de error `SQL1407N`.

Resumen de la gestión de diccionarios `LOAD TERMINATE` y `LOAD RESTART`

En la tabla siguiente se resume el comportamiento de gestión de diccionarios de compresión para el proceso de `LOAD` bajo la directiva `TERMINATE`.

Tabla 51. Gestión de diccionario `LOAD TERMINATE`

| Atributo <code>COMPRESS</code> (comprimir) tabla | ¿Existe el diccionario de datos de fila de tabla antes que <code>LOAD</code> ? | El diccionario de objetos de almacenamiento <code>XML</code> existe antes que <code>LOAD</code> ¹ | <code>TERMINATE: LOAD REPLACE KEEPDICTIONARY</code> o <code>LOAD INSERT</code> | <code>TERMINATE: LOAD REPLACE RESETDICTIONARY</code> |
|--|--|--|--|--|
| YES | YES | YES | Conservar los diccionarios existentes. | No se conserva ningún diccionario. ² |
| YES | YES | NO | Conservar diccionario existente. | No se conserva nada. ² |
| YES | NO | YES | Conservar diccionario existente. | No se conserva nada. |
| YES | NO | NO | No se conserva nada. | No se conserva nada. |
| NO | YES | YES | Conservar los diccionarios existentes. | No se conserva nada. |
| NO | YES | NO | Conservar diccionario existente. | No se conserva nada. |

Tabla 51. Gestión de diccionario LOAD TERMINATE (continuación)

| Atributo COMPRESS (comprimir) tabla | ¿Existe el diccionario de datos de fila de tabla antes que LOAD? | El diccionario de objetos de almacenamiento XML existe antes que LOAD ¹ | TERMINATE: LOAD REPLACE KEEPDICTIONARY o LOAD INSERT | TERMINATE: LOAD REPLACE RESETDICTIONARY |
|-------------------------------------|--|--|--|---|
| NO | NO | YES | Conservar diccionario existente. | No se conserva nada. |
| NO | NO | NO | No hacer nada. | No hacer nada. |

Nota:

1. Se puede crear un diccionario de compresión para el objeto de almacenamiento XML de una tabla sólo si las columnas XML se añaden a la tabla en DB2 Versión 9.7 o una versión posterior o si la tabla se migra mediante un movimiento de tabla en línea.
2. En el caso especial en el que la tabla tiene habilitada la captura de datos, se conserva el diccionario de datos de fila de tabla.

LOAD RESTART trunca una tabla hasta el último punto de coherencia alcanzado. Como parte del proceso LOAD RESTART, habrá un diccionario de compresión en la tabla si existía en la tabla en el momento en que se accedió al último punto de coherencia LOAD. En dicho caso, LOAD RESTART no creará un diccionario nuevo. Para obtener un resumen de las posibles condiciones, vea la Tabla 4 que figura a continuación.

Tabla 52. Gestión de diccionario LOAD RESTART

| Atributo COMPRESS (comprimir) tabla | ¿Existe el diccionario de datos de fila de tabla antes que el punto de coherencia LOAD? ¹ | ¿Existía el diccionario de objetos de almacenamiento XML antes que el último LOAD? ² | RESTART: LOAD REPLACE KEEPDICTIONARY o LOAD INSERT | RESTART: LOAD REPLACE RESETDICTIONARY |
|-------------------------------------|--|---|--|---|
| YES | YES | YES | Conservar los diccionarios existentes. | Conservar los diccionarios existentes. |
| YES | YES | NO | Mantener el diccionario de datos de fila de tabla existente y generar el diccionario XML sujeto a ADC. | Mantener el diccionario de datos de fila de tabla existente y generar un diccionario XML. |
| YES | NO | YES | Generar un diccionario de datos de fila de tabla sujeto a ADC. Conservar el diccionario XML existente. | Generar el diccionario de datos de fila de tabla. Conservar el diccionario XML existente. |
| YES | NO | NO | Generar los diccionarios de datos de fila de la tabla y los XML sujetos a ADC. | Generar los diccionarios XML y de datos de fila de la tabla. |
| NO | YES | YES | Conservar los diccionarios existentes. | Eliminar los diccionarios existentes. |
| NO | YES | NO | Conservar el diccionario de datos de fila de tabla existente. | Eliminar el diccionario de datos de fila de tabla existente. |

Tabla 52. Gestión de diccionario LOAD RESTART (continuación)

| Atributo COMPRESS (comprimir) tabla | ¿Existe el diccionario de datos de fila de tabla antes que el punto de coherencia LOAD? ¹ | ¿Existía el diccionario de objetos de almacenamiento XML antes que el último LOAD? ² | RESTART: LOAD REPLACE KEEPDICTIONARY o LOAD INSERT | RESTART: LOAD REPLACE RESETDICTIONARY |
|-------------------------------------|--|---|--|--|
| NO | NO | YES | Conservar el diccionario XML existente. | Eliminar el diccionario XML existente. |
| NO | NO | NO | No hacer nada. | No hacer nada. |

Notas:

1. No se tiene en cuenta la opción SAVECOUNT cuando se cargan los datos XML, las operaciones de carga que presentan anomalías durante la fase de carga se reinician desde el principio de la operación.
2. Se puede crear un diccionario de compresión para el objeto de almacenamiento XML de una tabla sólo si las columnas XML se añaden a la tabla en DB2 Versión 9.7 o una versión posterior o si la tabla se migra mediante un movimiento de tabla en línea.

Información de conjunto de resultados

El estado de la ejecución del mandato se devuelve en el área de comunicaciones SQL (SQLCA) obtenida a partir de la sentencia CALL. Si la ejecución es satisfactoria, el mandato devuelve información adicional. Una base de datos de partición individual devolverá un conjunto de resultados; una base de datos de múltiples particiones devolverá dos conjuntos de resultados.

- Tabla 53: Conjunto de resultados para la operación de carga.
- Tabla 54 en la página 142: El conjunto de resultados 2 contiene información para cada partición de base de datos en una operación de carga de múltiples particiones.

Tabla 53. Conjunto de resultados devuelto por el mandato LOAD

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| ROWS_READ | BIGINT | Número de filas leídas durante la operación de carga. |
| ROWS_SKIPPED | BIGINT | Número de filas que se han saltado antes de que comenzara la operación de carga. Esta información sólo se devuelve para una base de datos de partición individual. |
| ROWS_LOADED | BIGINT | Número de filas cargadas en la tabla destino. Esta información sólo se devuelve para una base de datos de partición individual. |
| ROWS_REJECTED | BIGINT | Número de filas que no se ha podido cargar en la tabla de destino. |
| ROWS_DELETED | BIGINT | Número de filas duplicadas que no se han cargado en la tabla de destino. Esta información sólo se devuelve para una base de datos de partición individual. |

Tabla 53. Conjunto de resultados devuelto por el mandato LOAD (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-----------------------|---------------|---|
| ROWS_COMMITTED | BIGINT | Número total de filas procesadas; el número de filas que se ha cargado satisfactoriamente en la tabla de destino, más el número de filas omitidas y rechazadas. Esta información sólo se devuelve para una base de datos de partición individual. |
| ROWS_PARTITIONED | BIGINT | Número de filas distribuidas por todos los agentes de distribución de base de datos. Esta información sólo se devuelve para una base de datos de múltiples particiones. |
| NUM_AGENTINFO_ENTRIES | BIGINT | Número de entradas devueltas en el segundo conjunto de resultados para una base de datos de múltiples particiones. Es el número de entradas de información de agente generadas por la operación de carga. Esta información sólo se devuelve para una base de datos de múltiples particiones. |
| MSG_RETRIEVAL | VARCHAR(512) | Sentencia de SQL que sirve para recuperar los mensajes creados por este programa de utilidad. Por ejemplo, <pre>SELECT SQLCODE, MSG FROM TABLE (SYSPROC.ADMIN_GET_MSGS ('2203498_thx')) AS MSG</pre> <p>Esta información sólo se devuelve si se especifica la cláusula MESSAGES ON SERVER.</p> |
| MSG_REMOVAL | VARCHAR(512) | Sentencia de SQL que sirve para hacer limpieza de los mensajes creados por este programa de utilidad. Por ejemplo: <pre>CALL SYSPROC.ADMIN_REMOVE_MSGS ('2203498_thx')</pre> <p>Esta información sólo se devuelve si se especifica la cláusula MESSAGES ON SERVER.</p> |

Tabla 54. El conjunto de resultados 2 devuelto por el mandato LOAD para cada partición de base de datos en una base de datos de múltiples particiones.

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| DBPARTITIONNUM | SMALLINT | El número de partición de base de datos en el que el agente ha ejecutado la operación de carga. |
| SQLCODE | INTEGER | El SQLCODE final resultante del proceso de carga. |

Tabla 54. El conjunto de resultados 2 devuelto por el mandato LOAD para cada partición de base de datos en una base de datos de múltiples particiones. (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| TABSTATE | VARCHAR(20) | <p>Estado de tabla después de que se completara la operación de carga. Puede ser:</p> <ul style="list-style-type: none"> • LOADPENDING: Indica que la carga no se ha completado, pero la tabla en la partición se ha dejado en un estado LOAD PENDING (carga pendiente). Debe realizarse una operación de reinicio o terminación de carga en la partición de base de datos. • NORMAL: Indica que la carga se ha completado satisfactoriamente en la partición de base de datos y la tabla ha salido del estado LOAD IN PROGRESS (o LOAD PENDING). Observe que la tabla puede seguir estando en estado Pendiente de establecimiento de integridad si es necesario realizar más proceso de restricciones, pero esta interfaz no comunica este estado. • UNCHANGED: Indica que la carga no se ha completado debido a un error, pero el estado de la tabla aún no ha cambiado. Es necesario realizar una operación de reinicio o terminación de carga en la partición de base de datos. <p>Nota: Esta interfaz no devuelve todos los estados de tabla posibles.</p> |
| AGENTTYPE | VARCHAR(20) | <p>Tipo de agente y es uno de los siguientes:</p> <ul style="list-style-type: none"> • FILE_TRANSFER • LOAD • LOAD_TO_FILE • PARTITIONING • PRE_PARTITIONING |

Modificadores de tipo de archivo para el programa de utilidad de carga

Tabla 55. Modificadores de tipo de archivo válidos para el programa de utilidad de carga: Todos los formatos de archivo

| Modificador | Descripción |
|-----------------|---|
| anyorder | Este modificador se utiliza junto con el parámetro <i>cpu_parallelism</i> . Especifica que la conservación del orden de los datos fuente no es necesaria, produciendo una ventaja significativa de rendimiento adicional en los sistemas SMP. Si el valor de <i>cpu_parallelism</i> es 1, se ignora esta opción. Esta opción no se soporta si <i>SAVECOUNT</i> > 0, dado que la recuperación de anomalías después de un punto de coherencia requiere que los datos se carguen en secuencia. |
| generatedignore | Este modificador informa al programa de utilidad de carga que en el archivo de datos existen datos para todas las columnas generadas pero que éstos se deberán ignorar. Esto hace que el programa de utilidad genere todos los valores para las columnas generadas. Este modificador no se puede utilizar con el modificador <i>generatedmissing</i> o el modificador <i>generatedoverride</i> . |

Tabla 55. Modificadores de tipo de archivo válidos para el programa de utilidad de carga: Todos los formatos de archivo (continuación)

| Modificador | Descripción |
|-------------------|---|
| generatedmissing | <p>Si se especifica este modificador, el programa de utilidad supone que el archivo de datos de entrada no contiene datos para la columna generada (ni siquiera NULL). Esto hace que el programa de utilidad genere todos los valores para las columnas generadas. Este modificador no se puede utilizar con el modificador generatedignore o el modificador generatedoverride.</p> |
| generatedoverride | <p>Este modificador indica al programa de utilidad de carga que acepte datos proporcionados por el usuario para todas las columnas generadas de la tabla (lo contrario a las normas normales para estos tipos de columnas). Esto es útil al migrar datos desde otro sistema de base de datos o al cargar una tabla de datos que se ha recuperado utilizando la opción RECOVER DROPPED TABLE en el mandato ROLLFORWARD DATABASE. Cuando se utilice este modificador, se rechazarán las filas sin datos o con datos NULL para una columna no anulable generada (SQL3116W). Cuando se utiliza este modificador, la tabla se colocará en estado Pendiente de establecer integridad. Para sacar la tabla del estado Pendiente de establecimiento de integridad sin verificar los valores suministrados por el usuario, emita el siguiente mandato después de la operación de carga:</p> <pre>SET INTEGRITY FOR < nombre-tabla > GENERATED COLUMN IMMEDIATE UNCHECKED</pre> <p>Para sacar la tabla del estado Pendiente de establecimiento de integridad y obligar a ejecutar una verificación de los valores suministrados por el usuario, ejecute el mandato siguiente después de la operación de carga:</p> <pre>SET INTEGRITY FOR < nombre-tabla > IMMEDIATE CHECKED.</pre> <p>Cuando se especifica este modificador y hay una columna generada en cualquiera de las claves de particionamiento, las claves de dimensión o de distribución, el mandato LOAD convertirá automáticamente el modificador a generatedignore y proseguirá con la carga. Esto tendrá un efecto de regeneración de todos los valores de columna generados.</p> <p>Este modificador no se puede utilizar con el modificador generatedmissing o generatedignore.</p> |
| identityignore | <p>Este modificador informa al programa de utilidad que existen datos para la columna de identidad en el archivo de datos pero que éstos se deberán ignorar. Esto hace que el programa de utilidad genere todos los valores de identidad. El comportamiento será el mismo para las columnas de identidad GENERATED ALWAYS y GENERATED BY DEFAULT. Esto significa que, para las columnas GENERATED ALWAYS, no se rechazará ninguna fila. Este modificador no se puede utilizar con el modificador identitymissing o identityoverride.</p> |
| identitymissing | <p>Si se especifica este modificador, el programa de utilidad supone que el archivo de datos de entrada no contiene datos para la columna de identidad (ni siquiera ningún NULL) y, por consiguiente, generará un valor para cada fila. El comportamiento será el mismo para las columnas de identidad GENERATED ALWAYS y GENERATED BY DEFAULT. Este modificador no se puede utilizar con el modificador identityignore o identityoverride.</p> |

Tabla 55. Modificadores de tipo de archivo válidos para el programa de utilidad de carga: Todos los formatos de archivo (continuación)

| Modificador | Descripción |
|--------------------------|--|
| identityoverride | <p>Este modificador sólo se deberá utilizar cuando exista una columna de identidad definida como GENERATED ALWAYS en la tabla que se debe cargar. Indica al programa de utilidad que acepte datos explícitos no NULL para dicha columna (lo contrario a las normas habituales para estos tipos de columnas de identidad). Esto es útil al migrar datos desde otro sistema de base de datos cuando la tabla debe definirse como GENERATED ALWAYS o cuando se carga una tabla desde datos que se han recuperado utilizando la opción DROPPED TABLE RECOVERY en el mandato ROLLFORWARD DATABASE. Cuando se utilice este modificador, se rechazarán las filas sin datos o con datos NULL para la columna de identidad (SQL3116W). Este modificador no se puede utilizar con el modificador identitymissing o identityignore. El programa de utilidad no intentará mantener o verificar la exclusividad de los valores de la columna de identidad de la tabla cuando se utilice esta opción.</p> |
| indexfreespace= <i>x</i> | <p><i>x</i> es un entero entre 0 y 99 inclusive. El valor se interpreta como el porcentaje de cada página de índice que se debe dejar como espacio libre cuando la carga reconstruye el índice. La carga con INDEXING MODE INCREMENTAL pasa por alto esta opción. La primera entrada de una página se añade sin restricciones; las entradas subsiguientes se añaden para mantener el porcentaje de umbral de espacio libre. El valor por omisión es el utilizado cuando se ejecuta CREATE INDEX.</p> <p>Este valor tiene prioridad sobre el valor PCTFREE especificado en la sentencia CREATE INDEX. La opción indexfreespace afecta sólo a las páginas hojas de índice.</p> |
| lobsinfile | <p><i>vía-lob</i> especifica la vía de acceso a los archivos que contienen datos de LOB. Los archivos de entrada de carga ASC, DEL o IXF contienen los nombres de los archivos que tienen datos LOB en la columna LOB.</p> <p>Esta opción no está soportada conjuntamente con el tipo de archivo CURSOR.</p> <p>La cláusula LOBS FROM especifica dónde se ubican los archivos LOB cuando se utiliza el modificador "lobsinfile". La cláusula LOBS FROM activará implícitamente la conducta LOBSINFIL. La cláusula LOBS FROM lleva al programa de utilidad LOAD la lista de vías de acceso para buscar los archivos LOB durante la carga de los datos.</p> <p>Cada vía de acceso contiene por lo menos un archivo que contiene por lo menos un LOB al que apunta un Especificador de ubicación de LOB (LLS) en el archivo de datos. El LLS es una representación de serie de la ubicación de un LOB en un archivo almacenado en la vía de acceso del archivo LOB. El formato de un LLS es <i>nombarch.ext.nnn.mmm/</i>, donde <i>nombarch.ext</i> es el nombre del archivo que contiene el LOB, <i>nnn</i> es el desplazamiento en bytes del LOB dentro del archivo y <i>mmm</i> es la longitud del LOB en bytes. Por ejemplo, si se almacena la serie db2exp.001.123.456/ en el archivo de datos, el LOB está ubicado en el desplazamiento 123 dentro del archivo db2exp.001 y tiene una longitud de 456 bytes.</p> <p>Para indicar un LOB nulo, entre el tamaño como -1. Si el tamaño se especifica como 0, se trata como un LOB de longitud 0. Para los LOB nulos de longitud -1, se pasan por alto el desplazamiento y el nombre de archivo. Por ejemplo, el LLS de un LOB nulo puede ser db2exp.001.7.-1/.</p> |

Tabla 55. Modificadores de tipo de archivo válidos para el programa de utilidad de carga: Todos los formatos de archivo (continuación)

| Modificador | Descripción |
|----------------------------|---|
| noheader | <p>Salta el código de verificación de cabecera (sólo aplicable a operaciones de carga en tablas que residen en un grupo de nodos de una partición de base de datos de una sola partición).</p> <p>Si se utiliza la carga MPP por omisión (modalidad PARTITION_AND_LOAD) para una tabla que reside en un grupo de particiones de bases de datos de una sola partición, no se espera que el archivo tenga una cabecera. Por tanto, no es necesario el modificador noheader. Si se utiliza la modalidad LOAD_ONLY, se espera que el archivo tenga una cabecera. La única circunstancia en la que se deberá necesitar la utilización del modificador noheader si desea realizar la operación LOAD_ONLY utilizando un archivo que no tenga una cabecera.</p> |
| norowwarnings | Suprime todos los avisos sobre filas rechazadas. |
| pagefreespace=x | <p>x es un entero entre 0 y 100 inclusive. El valor se interpreta como el porcentaje de cada página de datos que se debe dejar como espacio libre. Si el valor especificado no es válido debido al tamaño mínimo de fila (por ejemplo, una fila que tenga como mínimo 3000 bytes de longitud y un valor x de 50), la fila se colocará en una página nueva. Si se especifica un valor de 100, cada fila residirá en una página nueva. El valor PCTFREE de una tabla determina la cantidad de espacio libre designado por página. Si no se ha establecido un valor pagefreespace en la operación de carga o un valor PCTFREE en una tabla, el programa de utilidad llenará tanto espacio como sea posible en cada página. El valor establecido por pagefreespace alterará temporalmente el valor PCTFREE especificado para la tabla.</p> |
| rowchangetimestampignore | <p>Este modificador informa al programa de utilidad de carga que los datos de la columna de indicación de fecha y hora de cambio de fila existen en el archivo de datos pero deben omitirse. Esto hace que el programa de utilidad genere todas las columnas ROW CHANGE TIMESTAMP. El comportamiento será el mismo tanto para las columnas GENERATED ALWAYS como para GENERATED BY DEFAULT. Esto significa que, para las columnas GENERATED ALWAYS, no se rechazará ninguna fila. Este modificador no se puede utilizar con el modificador rowchangetimestampmissing o rowchangetimestampoverride.</p> |
| rowchangetimestampmissing | <p>Si se especifica este modificador, el programa de utilidad presupone que el archivo de datos de entrada no contiene ningún dato para la columna de indicación de fecha y hora de cambio de fila (ni siquiera ningún NULL) y por lo tanto, generará un valor para cada fila. El comportamiento será el mismo tanto para las columnas GENERATED ALWAYS como para GENERATED BY DEFAULT. Este modificador no se puede utilizar con el modificador rowchangetimestampignore o rowchangetimestampoverride.</p> |
| rowchangetimestampoverride | <p>Este modificador sólo debe utilizarse cuando exista una columna de indicación de fecha y hora de cambio de fila definida como GENERATED ALWAYS en la tabla que se debe cargar. Indica al programa de utilidad que acepte datos explícitos no NULL para dicha columna (lo contrario de las normas habituales para estos tipos de columnas de indicación de fecha y hora de cambio de fila). Esto es útil al migrar datos desde otro sistema de base de datos cuando la tabla debe definirse como GENERATED ALWAYS o cuando se carga una tabla desde datos que se han recuperado utilizando la opción DROPPED TABLE RECOVERY en el mandato ROLLFORWARD DATABASE. Cuando se utiliza este modificador, se rechazarán las filas sin ningún dato o datos NULL para la columna ROW CHANGE TIMESTAMP (SQL3116W). Este modificador no se puede utilizar con el modificador rowchangetimestampmissing o rowchangetimestampignore. El programa de utilidad de carga no intentará mantener o verificar la exclusividad de los valores de la columna de indicación de fecha y hora de cambio de fila cuando se utilice esta opción.</p> |

Tabla 55. Modificadores de tipo de archivo válidos para el programa de utilidad de carga: Todos los formatos de archivo (continuación)

| Modificador | Descripción |
|-------------------------|---|
| seclabelchar | <p>Indica que las etiquetas de seguridad en el archivo fuente de entrada están en formato de serie para los valores de etiqueta de seguridad en lugar de en el formato numérico codificado por omisión. LOAD convierte cada etiqueta de seguridad al formato interno al cargarse. Si una serie no tiene el formato adecuado, la fila no se carga y se devuelve un aviso (SQLSTATE 01H53, SQLCODE SQL3242W). Si la serie no representa una etiqueta de seguridad válida que forma parte de la política de seguridad que protege la tabla, la fila no se carga y se devuelve un aviso (SQLSTATE 01H53, SQLCODE SQL3243W).</p> <p>Este modificador no puede especificarse si se especifica el modificador seclabelname, de lo contrario la carga falla y se devuelve un error (SQLCODE SQL3525N).</p> <p>Si tiene una tabla que consta de una sola columna DB2SECURITYLABEL, el archivo de datos puede tener el aspecto siguiente:</p> <pre> "CONFIDENTIAL:ALPHA:G2" "CONFIDENTIAL;SIGMA:G2" "TOP SECRET:ALPHA:G2" </pre> <p>Para cargar o importar estos datos, es necesario que se utilice el modificador de tipo de archivo SECLABELCHAR:</p> <pre>LOAD FROM input.del OF DEL MODIFIED BY SECLABELCHAR INSERT INTO t1</pre> |
| seclabelname | <p>Indica que las etiquetas de seguridad en el archivo fuente de entrada están indicadas por su nombre en lugar de en el formato numérico codificado por omisión. LOAD convertirá el nombre a la etiqueta de seguridad adecuada, si existe. Si no existe ninguna etiqueta de seguridad con el nombre indicado para la política de seguridad que protege la tabla, la fila no se carga y se devuelve un aviso (SQLSTATE 01H53, SQLCODE SQL3244W).</p> <p>Este modificador no puede especificarse si se especifica el modificador seclabelchar, de lo contrario la carga falla y se devuelve un error (SQLCODE SQL3525N).</p> <p>Si tiene una tabla que consta de una sola columna DB2SECURITYLABEL, el archivo de datos podría contener nombres de etiquetas de seguridad similares a:</p> <pre> "LABEL1" "LABEL1" "LABEL2" </pre> <p>Para cargar o importar estos datos, es necesario que se utilice el modificador de tipo de archivo SECLABELNAME:</p> <pre>LOAD FROM input.del OF DEL MODIFIED BY SECLABELNAME INSERT INTO t1</pre> <p>Nota: Si el tipo de archivo es ASC, los espacios a continuación del nombre de la etiqueta de seguridad se interpretarán como parte del nombre. Para evitarlo utilice el modificador de tipo de archivo striptblanks para asegurarse de que se eliminan los espacios.</p> |
| totalreespace= <i>x</i> | <p><i>x</i> es un entero mayor o igual que 0. El valor se interpreta como el porcentaje de las páginas totales de la tabla que debe añadirse al final de tabla como espacio libre. Por ejemplo, si <i>x</i> es 20 y la tabla tiene 100 páginas de datos después de que se hayan cargado los datos, se añadirán 20 páginas vacías adicionales. El número total de páginas de datos para la tabla será de 120. El total de páginas de datos no es un factor en el número de páginas de índice de la tabla. Esta opción no afecta al objeto de índice. Si se realizan dos cargas especificando esta opción, la segunda carga no volverá a utilizar el espacio extra añadido al final por la primera carga.</p> |

Tabla 55. Modificadores de tipo de archivo válidos para el programa de utilidad de carga: Todos los formatos de archivo (continuación)

| Modificador | Descripción |
|-------------|--|
| usedefaults | <p>Si se ha especificado una columna fuente para una columna de tabla de destino, pero dicha columna fuente no contiene datos para una o más instancias de fila, se cargan los valores por omisión. He aquí unos ejemplos de datos que faltan:</p> <ul style="list-style-type: none"> • Para archivos DEL: se especifican dos delimitadores de columna adyacentes (",,") o dos delimitadores de columnas adyacentes separados por un número arbitrario de espacios (" , ") para un valor de columna. • Para archivos DEL/ASC/WSF: una fila que no tiene suficientes columnas o que no es suficientemente larga para la especificación original. Para archivos ASC, los valores de columna NULL no se considera que falten de modo explícito y no se sustituirán los valores de columna NULL por valores por omisión. Los valores de columna NULL se representan mediante todos los caracteres de espacio para columnas numéricas, de fecha, hora e indicación de fecha y hora o utilizando NULL INDICATOR para una columna de cualquier tipo para indicar que la columna es NULL. <p>Sin esta opción, si la columna fuente no contiene datos para una instancia de fila, se produce una de las acciones siguientes:</p> <ul style="list-style-type: none"> • Para archivos DEL/ASC/WSF: Si la columna es anulable, se carga un NULL. Si la columna no es anulable, el programa de utilidad rechaza la fila. |

Tabla 56. Modificadores de tipo de archivo válidos para el programa de carga: Formatos de archivo ASCII (ASC/DEL)

| Modificador | Descripción |
|-------------|---|
| codepage=x | <p>x es una serie de caracteres ASCII. El valor se interpreta como la página de códigos de los datos del archivo de entrada. Convierte datos de tipo carácter (y datos numéricos especificados en caracteres) de esta página de códigos a la página de códigos de la base de datos durante la operación de carga.</p> <p>Se aplican las siguientes normas:</p> <ul style="list-style-type: none"> • Para DBCS (gráfico) puro, DBCS mixto y EUC, los delimitadores están restringidos al rango de x00 a x3F, inclusive. • Para datos DEL especificados en una página de códigos EBCDIC, es posible que los delimitadores no coincidan con los caracteres DBCS de desplazamiento a teclado estándar y de desplazamiento a teclado ideográfico. • nullindchar debe especificar símbolos incluidos en el conjunto ASCII estándar entre los elementos de código x20 t x7F, inclusive. Esto hace referencia a símbolos y elementos de código ASCII. Los datos EBCDIC pueden utilizar los símbolos correspondientes, aunque los elementos de código serán diferentes. <p>Esta opción no está soportada conjuntamente con el tipo de archivo CURSOR.</p> |

Tabla 56. Modificadores de tipo de archivo válidos para el programa de carga: Formatos de archivo ASCII (ASC/DEL) (continuación)

| Modificador | Descripción |
|-------------------|--|
| dateformat="x" | <p>x es el formato de la fecha en el archivo fuente.¹ Los elementos de fecha válidos son:</p> <ul style="list-style-type: none"> AAAA - Año (cuatro dígitos entre 0000 y 9999) M - Mes (uno o dos dígitos entre 1 y 12) MM - Mes (dos dígitos entre 1 y 12; se excluye mutuamente con M) D - Día (uno o dos dígitos entre 1 y 31) DD - Día (dos dígitos entre 1 y 31; se excluye mutuamente con D) DDD - Día del año (tres dígitos entre 001 y 366; se excluye mutuamente con otros elementos de día o mes) <p>Se asigna un valor por omisión de 1 para cada elemento que no se haya especificado. He aquí unos ejemplos de formatos de fecha:</p> <pre>"D-M-AAAA" "MM.DD.AAAA" "AAAADD"</pre> |
| dumpfile = x | <p>x es el nombre totalmente calificado (de acuerdo con la partición de base de datos de servidor) de un archivo de excepción en el que se graban las filas rechazadas. Se graba un máximo de 32 KB de datos por registro. He aquí un ejemplo que muestra cómo especificar un archivo de vuelcos:</p> <pre>db2 load from data of del modified by dumpfile = /u/user/nombre-archivo insert into table_name</pre> <p>El archivo se creará y será propiedad del propietario de la instancia. Para alterar temporalmente los permisos de archivo por omisión, utilice el modificador de tipo de archivo dumpfileaccessall.</p> <p>Nota:</p> <ol style="list-style-type: none"> 1. En un entorno de bases de datos particionadas, la vía de acceso debe ser local en la partición de base de datos de carga, de forma que las operaciones de carga que se ejecuten simultáneamente no intenten escribir en el mismo archivo. 2. El contenido del archivo se graba en disco en una modalidad asíncrona de almacenamiento intermedio. En el caso de una operación de carga anómala o interrumpida, el número de registros confirmados en disco no se puede saber con certeza y no se puede garantizar la coherencia después de una operación LOAD RESTART. Sólo se puede suponer que el archivo está completo para una operación de carga que se inicia y se completa en un solo paso. 3. Si el archivo especificado ya existe, no se volverá a crear, sino que se añadirá. |
| dumpfileaccessall | <p>Otorga acceso de lectura a 'OTHERS' cuando se crea un archivo de vuelco.</p> <p>Este modificador de tipo de archivo sólo es válido cuando:</p> <ol style="list-style-type: none"> 1. se utiliza junto con el modificador de tipo de archivo dumpfile 2. el usuario dispone de privilegio SELECT sobre la tabla de destino de carga 3. se emite en una partición de base de datos del servidor DB2 que reside en un sistema operativo UNIX. <p>Si el archivo especificado ya existe, los permisos no se modificarán.</p> |

Tabla 56. Modificadores de tipo de archivo válidos para el programa de carga: Formatos de archivo ASCII (ASC/DEL) (continuación)

| Modificador | Descripción |
|----------------|---|
| fastparse | <p>Debe utilizarse con precaución. Reduce la comprobación de la sintaxis en los valores de columna suministrados por el usuario y mejora el rendimiento. Se garantiza que las tablas sean arquitectónicamente correctas (el programa de utilidad realiza suficientes comprobaciones para evitar una interrupción o violación de segmentación). No obstante, no se valida la coherencia de los datos. Esta opción debe utilizarse solamente si se está seguro de que los datos son coherentes y correctos. Por ejemplo, si los datos suministrados por el usuario contienen un valor no válido de la columna de indicación de fecha y hora de :1>0-00-20-07.11.12.000000, este valor se insertará en la tabla si se ha especificado FASTPARSE, y se rechazará si no se ha especificado.</p> |
| implieddecimal | <p>La ubicación de una coma decimal implícita la determina la definición de columna; ya no se supone que está al final del valor. Por ejemplo, el valor 12345 se carga en una columna DECIMAL(8,2) como 123,45, <i>no</i> como 12345,00.</p> <p>Este modificador no se puede utilizar con el modificador packeddecimal.</p> |
| timeformat="x" | <p>x es el formato de la hora en el archivo fuente.¹ Los elementos de hora válidos son:</p> <ul style="list-style-type: none"> H - Hora (uno o dos dígitos entre 0 y 12 para un sistema de 12 horas y entre 0 y 24 para un sistema de 24 horas) HH - Hora (dos dígitos entre 0 y 12 para un sistema de 12 horas y entre 0 y 24 para un sistema de 24 horas; se excluye mutuamente con H) M - Minuto (uno o dos dígitos entre 0 y 59) MM - Minuto (dos dígitos entre 0 y 59; se excluye mutuamente con M) S - Segundo (uno o dos dígitos entre 0 y 59) SS - Segundo (dos dígitos entre 0 y 59; se excluye mutuamente con S) SSSSS - Segundo del día después de medianoche (5 dígitos entre 00000 y 86399; se excluye mutuamente con otros elementos de hora) TT - Indicador de meridiano (AM o PM) <p>Se asigna un valor por omisión de 0 para cada elemento que no se ha especificado. He aquí unos ejemplos de formatos de hora:</p> <p>"HH:MM:SS" "HH.MM TT" "SSSSS"</p> |

Tabla 56. Modificadores de tipo de archivo válidos para el programa de carga: Formatos de archivo ASCII (ASC/DEL) (continuación)

| Modificador | Descripción |
|---------------------|---|
| timestampformat="x" | <p>x es el formato de la indicación de fecha y hora en el archivo fuente.¹ Los elementos válidos de la indicación de fecha y hora son:</p> <ul style="list-style-type: none"> AAAA - Año (cuatro dígitos entre 0000 y 9999) M - Mes (uno o dos dígitos entre 1 y 12) MM - Mes (dos dígitos entre 01 y 12; se excluye mutuamente con M y MMM) MMM - Mes (abreviación de tres letras no sensible a mayúsculas y minúsculas para el nombre del mes; se excluye mutuamente con M y MM) D - Día (uno o dos dígitos entre 1 y 31) DD - Día (dos dígitos entre 1 y 31; se excluye mutuamente con D) DDD - Día del año (tres dígitos entre 001 y 366; se excluye mutuamente con otros elementos de día o mes) H - Hora (uno o dos dígitos entre 0 y 12 para un sistema de 12 horas y entre 0 y 24 para un sistema de 24 horas) HH - Hora (dos dígitos entre 0 y 12 para un sistema de 12 horas y entre 0 y 24 para un sistema de 24 horas; se excluye mutuamente con H) M - Minuto (uno o dos dígitos entre 0 y 59) MM - Minuto (dos dígitos entre 0 y 59; se excluye mutuamente con M, minuto) S - Segundo (uno o dos dígitos entre 0 y 59) SS - Segundo (dos dígitos entre 0 y 59; se excluye mutuamente con S) SSSSS - Segundo del día después de medianoche (5 dígitos entre 00000 y 86399; se excluye mutuamente con otros elementos de hora) U (de 1 a 12 veces) - Segundos fraccionados (el número de casos de U representa el número de dígitos con cada dígito comprendido entre 0 y 9) TT - Indicador de meridiano (AM o PM) |

Tabla 56. Modificadores de tipo de archivo válidos para el programa de carga: Formatos de archivo ASCII (ASC/DEL) (continuación)

| Modificador | Descripción |
|---|---|
| <p>timestampformat="x" (Continuación)</p> | <p>Se asigna un valor por omisión de 1 para los elementos AAAA, M, MM, D, DD o DDD no especificados. Se asigna un valor por omisión de 'Ene' a un elemento MMM no especificado. Se asigna un valor por omisión de 0 para todos los demás elementos no especificados. A continuación se muestra un ejemplo de un formato de indicación de la hora:</p> <p style="text-align: center;">"AAAA/MM/DD HH:MM:SS.UUUUUU"</p> <p>Los valores válidos para el elemento MMM incluyen: 'ene', 'feb', 'mar', 'abr', 'may', 'jun', 'jul', 'ago', 'sep', 'oct', 'nov' y 'dic'. Estos valores son sensibles a las mayúsculas y minúsculas.</p> <p>Si no se especifica el modificador <code>TIMESTAMPFORMAT</code>, el programa de utilidad de carga formatea el campo de indicación de fecha y hora utilizando uno de los dos formatos posibles:</p> <p>AAAA-MM-DD-HH.MM.SS AAAA-MM-DD HH:MM:SS</p> <p>El programa de utilidad de carga elige el formato examinando el separador entre DD y HH. Si se trata de un guión, '-', el programa de utilidad de carga utilizará los guiones habituales y el formato de puntos (AAAA-MM-DD-HH.MM.SS). Si se trata de un espacio en blanco, el programa de utilidad de carga prevé un signo de dos puntos '.' para separar HH, MM y SS.</p> <p>En cualquiera de los formatos, si incluye el campo de microsegundos (UUUUUU), el programa de utilidad de carga espera el punto '.' como separador. Cualquiera de los dos formatos AAAA-MM-DD-HH.MM.SS.UUUUUU o AAAA-MM-DD HH:MM:SS.UUUUUU son aceptables.</p> <p>En el siguiente ejemplo se muestra cómo cargar datos que contienen formatos de fecha y hora definidos por el usuario en una tabla llamada <code>schedule</code>:</p> <pre>db2 load from delfile2 of del modified by timestampformat="aaa.mm.dd hh:mm tt" insert into schedule</pre> |
| <p>usegraphiccodepage</p> | <p>Si se proporciona <code>usegraphiccodepage</code>, se supone que los datos cargados en campos de datos de objeto gráfico o de gran objeto de caracteres de doble byte (DBCLOB) se encuentran en la página de códigos gráfica. Se supone que el resto de los datos estarán en la página de códigos de caracteres. La página de códigos gráfica está asociada con la página de códigos de caracteres. <code>LOAD</code> determina la página de códigos de caracteres a través del modificador <code>codepage</code>, si se especifica o a través de la página de códigos de la base de datos si no se especifica el modificador <code>codepage</code>.</p> <p>Este modificador debe utilizarse junto con el archivo de datos delimitado generado por la recuperación de descarte de tabla sólo si la tabla que se recupera tiene datos gráficos.</p> <p>Restricciones</p> <p>El modificador <code>usegraphiccodepage</code> NO DEBE especificarse con archivos <code>DEL</code> creados por el programa de utilidad <code>EXPORT</code>, ya que estos archivos contienen datos codificados en una sola página de códigos. También ignorarán el modificador <code>usegraphiccodepage</code> los grandes objetos de caracteres de doble byte (DBCLOB) en los archivos.</p> |

Tabla 56. Modificadores de tipo de archivo válidos para el programa de carga: Formatos de archivo ASCII (ASC/DEL) (continuación)

| Modificador | Descripción |
|-------------|---|
| xmlchar | <p>Especifica que los documentos XML se codifican con la página de códigos de caracteres.</p> <p>Esta opción es útil para procesar documentos XML que estén codificados con la página de códigos de caracteres especificada pero que no contengan una declaración de codificación.</p> <p>Para cada documento, si existe un identificador de declaración y contiene un atributo de codificación, la codificación debe coincidir con la página de códigos de caracteres, de lo contrario se rechazará la fila que contiene el documento. Tenga en cuenta que la página de códigos es el valor especificado por el modificador de tipo de archivo codepage o la página de códigos de la aplicación, si no se ha especificado. Por omisión, los documentos están codificados en Unicode o bien contienen un identificador de declaración con un atributo de codificación.</p> |
| xmlgraphic | <p>Especifica que los documentos XML se codifican con la página de códigos de gráficos especificada.</p> <p>Esta opción es útil para procesar documentos XML que estén codificados con una página de códigos de gráficos específica, pero que no contengan una declaración de codificación.</p> <p>Para cada documento, si existe un identificador de declaración y contiene un atributo de codificación, la codificación debe coincidir con la página de códigos de gráficos, de lo contrario se rechazará la fila que contiene el documento. Tenga en cuenta que la página de códigos de gráficos es el componente gráfico del valor especificado por el modificador de tipo de archivo codepage o el componente gráfico de la página de códigos de la aplicación, si no se ha especificado. Por omisión, los documentos están codificados en Unicode o bien contienen un identificador de declaración con un atributo de codificación.</p> |

Tabla 57. Modificadores de tipo de archivo válidos para el programa de carga: Formatos de archivo ASC (ASCII no delimitado)

| Modificador | Descripción |
|----------------|---|
| binarynumerics | <p>Los datos numéricos (pero no DECIMAL) deben estar en formato binario, no en la representación de caracteres. Esto evita conversiones costosas.</p> <p>Esta opción sólo se soporta con ASC posicional, utilizando los registros de longitud fija especificados por la opción reclen.</p> <p>Se aplican las siguientes normas:</p> <ul style="list-style-type: none"> • No se realiza ninguna conversión entre tipos de datos, a excepción de BIGINT, INTEGER y SMALLINT. • Las longitudes de los datos deben coincidir con las definiciones de las columnas de destino. • Los FLOAT deben estar en formato de Coma flotante IEEE. • Se supone que los datos binarios del archivo fuente de carga están en formato big-endian (byte más significativo primero), independientemente de la plataforma en la que se esté ejecutando la operación de carga. <p>No pueden existir NULL en los datos para las columnas afectadas por este modificador. Cuando se utiliza este modificador, los espacios en blanco (normalmente interpretados como NULL), se interpretan como un valor binario.</p> |

Tabla 57. Modificadores de tipo de archivo válidos para el programa de carga: Formatos de archivo ASC (ASCII no delimitado) (continuación)

| Modificador | Descripción |
|----------------|---|
| nochecklengths | <p>Si se especifica nochecklengths, se intenta cargar cada fila, incluso si los datos fuente tienen una definición de columna que excede el tamaño de la columna de tabla de destino. Dichas filas se pueden cargar satisfactoriamente si la conversión de página de códigos hace que los datos fuente se acorten; por ejemplo, datos EUC de 4 bytes en la fuente pueden acortarse a datos DBCS de 2 bytes en el destino y necesitar la mitad del espacio. Esta opción es especialmente útil si se sabe que los datos fuente cabrán en todos los casos, a pesar de las discrepancias de las definiciones de columna.</p> |
| nullindchar=x | <p>x es un carácter individual. Cambia el carácter que indica un valor NULL a x. El valor por omisión de x es \backslash.²</p> <p>Este modificador es sensible a las mayúsculas y minúsculas para archivos de datos EBCDIC, excepto cuando el carácter es una letra inglesa. Por ejemplo, si se especifica que el carácter indicador de NULL sea la letra N, se reconoce también n como un indicador de NULL.</p> |
| packeddecimal | <p>Carga directamente datos decimales empaquetados, dado que el modificador binarynumerics no incluye el tipo de campo DECIMAL.</p> <p>Esta opción sólo se soporta con ASC posicional, utilizando los registros de longitud fija especificados por la opción reclen.</p> <p>Los valores soportados para la porción de signo son:</p> <ul style="list-style-type: none"> + = 0xC 0xA 0xE 0xF - = 0xD 0xB <p>No pueden existir NULL en los datos para las columnas afectadas por este modificador. Cuando se utiliza este modificador, los espacios en blanco (normalmente interpretados como NULL), se interpretan como un valor binario.</p> <p>Independientemente de la plataforma del servidor, se supone que el orden de bytes de los datos binarios en el archivo fuente de carga es big-endian (byte más significativo primero); es decir, cuando se utiliza este modificador en los sistemas operativos Windows, no se debe invertir el orden de los bytes.</p> <p>Este modificador no se puede utilizar con el modificador implieddecimal.</p> |
| reclen=x | <p>x es un entero con un valor máximo de 32767. Se leen x caracteres para cada fila y no se utiliza un carácter de nueva línea para indicar el final de la fila.</p> |
| striptblanks | <p>Trunca los espacios en blanco de cola al cargar datos en un campo de longitud variable. Si no se especifica esta opción, se conservan los espacios en blanco.</p> <p>Esta opción no se puede especificar junto con striptnulls. Son opciones que se excluyen mutuamente. Esta opción sustituye a la opción t obsoleta, que se soporta únicamente por compatibilidad con niveles anteriores.</p> |
| striptnulls | <p>Trunca los NULL de cola (caracteres 0x00) al cargar datos en un campo de longitud variable. Si no se especifica esta opción, se conservan los NULL.</p> <p>Esta opción no se puede especificar junto con striptblanks. Son opciones que se excluyen mutuamente. Esta opción sustituye a la opción padwithzero obsoleta, que sólo se soporta por compatibilidad con niveles anteriores.</p> |

Tabla 57. Modificadores de tipo de archivo válidos para el programa de carga: Formatos de archivo ASC (ASCII no delimitado) (continuación)

| Modificador | Descripción |
|--------------|--|
| zoneddecimal | <p>Carga datos decimales con zona, puesto que el modificador BINARYNUMERICs no incluye el tipo de campo DECIMAL. Esta opción sólo se soporta con ASC posicional, utilizando los registros de longitud fija especificados por la opción RECLLEN.</p> <p>Los valores de signo de medio byte pueden ser uno de los siguientes:</p> <ul style="list-style-type: none"> + = 0xC 0xA 0xE 0xF - = 0xD 0xB <p>Los valores soportados para dígitos son de 0x0 a 0x9.</p> <p>Los valores soportados para zonas son de 0x3 a 0xF.</p> |

Tabla 58. Modificadores de tipo de archivo válidos para el programa de carga: Formatos de archivo DEL (ASCII delimitado)

| Modificador | Descripción |
|-----------------|--|
| chardelx | <p>x es un delimitador de serie de un solo carácter. El valor por omisión son las comillas dobles ("). El carácter especificado se utiliza en lugar de las comillas dobles para encerrar una serie de caracteres.²³ Si desea especificar explícitamente las comillas dobles (") como delimitador de serie de caracteres, debe especificarse de la manera siguiente:</p> <pre>modified by chardel""</pre> <p>También se pueden especificar las comillas simples (') como delimitador de serie de caracteres del modo siguiente:</p> <pre>modified by chardel''</pre> |
| coldelx | <p>x es un delimitador de columna de un solo carácter. El valor por omisión es una coma (.). En lugar de una coma, se utiliza el carácter especificado para indicar el final de una columna.²³</p> |
| decplusblank | <p>Carácter de signo más. Hace que a los valores decimales positivos se les ponga un espacio en blanco como prefijo en lugar de un signo más (+). La acción por omisión es poner a los valores decimales positivos un signo más como prefijo.</p> |
| decptx | <p>x es un sustituto de un solo carácter del punto como carácter de coma decimal. El valor por omisión es un punto (.). En lugar de un punto, se utilizará el carácter especificado como carácter de coma decimal.²³</p> |
| delprioritychar | <p>La prioridad por omisión actual para los delimitadores es: delimitador de registro, delimitador de carácter, delimitador de columna. Este modificador protege a las aplicaciones existentes que dependen de la prioridad más antigua invirtiendo las prioridades de los delimitadores del modo siguiente: delimitador de carácter, delimitador de registro, delimitador de columna. Sintaxis:</p> <pre>db2 load ... modified by delprioritychar ...</pre> <p>Por ejemplo, dado el archivo de datos DEL siguiente:</p> <pre>"Smith, Joshua",4000,34.98<delimitador de fila> "Vincent,<delimitador de fila>, is a manager", 4005,44.37<delimitador de fila></pre> <p>Si se especifica el modificador delprioritychar, sólo habrá dos filas en este archivo de datos. El segundo <delimitador de fila> se interpretará como parte de la primera columna de datos de la segunda fila, mientras que el primero y tercero <delimitadores de fila> se interpretarán como delimitadores de registro reales. Si no se especifica este modificador, habrá tres filas en este archivo de datos, cada una de las cuales estará delimitada por un <delimitador de fila>.</p> |

Tabla 58. Modificadores de tipo de archivo válidos para el programa de carga: Formatos de archivo DEL (ASCII delimitado) (continuación)

| Modificador | Descripción |
|-------------|---|
| keepblanks | <p>Conserva los blancos iniciales y de cola en cada campo de tipo CHAR, VARCHAR, LONG VARCHAR o CLOB. Sin esta opción, se eliminan todos los espacios en blanco iniciales y de cola que no están incluidos entre los delimitadores de caracteres y se inserta un NULL en la tabla para todos los campos en blanco.</p> <p>El ejemplo siguiente ilustra cómo cargar datos en una tabla denominada TABLE1, al mismo tiempo que se conservan todos los espacios iniciales y de cola del archivo de datos:</p> <pre>db2 load from delfile3 of del modified by keepblanks insert into table1</pre> |
| nochardel | <p>El programa de utilidad de carga asumirá que todos los bytes encontrados entre los delimitadores de columna forman parte de los datos de la columna. Los delimitadores de carácter se analizarán como parte de los datos de la columna. Esta opción no debe especificarse si los datos se han exportado utilizando DB2 (a menos que se haya especificado nochardel en el momento de la exportación). Se proporciona para dar soporte a archivos de datos de proveedores que no tienen delimitadores de caracteres. El uso incorrecto podría causar la pérdida o la corrupción de los datos.</p> <p>Esta opción no se puede especificar con chardelx, delprioritychar o nodoubledel. Son opciones que se excluyen mutuamente.</p> |
| nodoubledel | Suprime el reconocimiento de los delimitadores de caracteres dobles. |

Tabla 59. Modificadores de tipo de archivo válidos para el programa de utilidad de carga: Formato de archivo IXF

| Modificador | Descripción |
|----------------|---|
| forcein | <p>Indica al programa de utilidad que acepte datos a pesar de las discrepancias de página de códigos y que suprima la conversión entre páginas de códigos.</p> <p>Se comprueban los campos de destino de longitud fija para verificar que son suficientemente grandes para los datos. Si se especifica nochecklengths, no se realiza ninguna comprobación y se realiza un intento de cargar cada fila.</p> |
| nochecklengths | Si se especifica nochecklengths, se intenta cargar cada fila, incluso si los datos fuente tienen una definición de columna que excede el tamaño de la columna de tabla de destino. Dichas filas se pueden cargar satisfactoriamente si la conversión de página de códigos hace que los datos fuente se acorten; por ejemplo, datos EUC de 4 bytes en la fuente pueden acortarse a datos DBCS de 2 bytes en el destino y necesitar la mitad del espacio. Esta opción es especialmente útil si se sabe que los datos fuente cabrán en todos los casos, a pesar de las discrepancias de las definiciones de columna. |

Nota:

1. Es obligatorio escribir la serie de formato de fecha entre comillas dobles. Los separadores de campo no pueden contener ninguno de los caracteres siguientes: a-z, A-Z y 0-9. El separador de campo no debe ser igual que el delimitador de carácter o de campo en el formato de archivo DEL. Un separador de campo es opcional si las posiciones inicial y final de un elemento no son ambiguas. La ambigüedad puede existir si (en función del modificador) se usan elementos tales como D, H, M o S, debido a la longitud variable de las entradas. Para formatos de indicación de la hora, hay que tener cuidado de evitar la ambigüedad entre los descriptores de mes y de minuto, dado que ambos

utilizan la letra M. Un campo de mes debe estar junto a otros campos de fecha. Un campo de minuto debe ser adyacente a otros campos de hora. He aquí unos formatos ambiguos de indicación de fecha y hora:

"M" (puede ser mes o minuto)
 "M:M" (¿Cuál es cada uno?)
 "M:AAA:M" (Ambos se interpretan como mes.)
 "S:M:AAA" (adyacente a un valor de hora y un valor de fecha)

En los casos ambiguos, el programa de utilidad informará con un mensaje de error y la operación fallará.

He aquí unos formatos no ambiguos de indicación de fecha y hora:

"M:AAA" (Mes)
 "S:M" (Minuto)
 "M:AAA:S:M" (Mes...Minuto)
 "M:H:AAA:M:D" (Minuto...Mes)

Algunos caracteres, como las comillas dobles y las barras inclinadas invertidas, deben ir precedidos de un carácter de escape (por ejemplo, \).

- Los valores de tipo carácter proporcionados para los modificadores char del, coldel o decpt deben especificarse en la página de códigos de los datos fuente. Se puede especificar el elemento de código de carácter (en lugar del símbolo de carácter), utilizando la sintaxis xJJ o 0xJJ, donde JJ es la representación hexadecimal del elemento de código. Por ejemplo, para especificar el carácter # como delimitador de columna, utilice uno de los siguientes:

... modified by coldel# ...
 ... modified by coldel0x23 ...
 ... modified by coldelX23 ...

- En la sección *Consideraciones sobre el delimitador para mover datos* se muestran las restricciones que se aplican a los caracteres que se pueden utilizar como alteración temporal de los delimitadores.
- El programa de utilidad de carga no emite un aviso si se intenta utilizar tipos de archivo no soportados con la opción MODIFIED BY. Si se intenta esto, la operación de carga falla y se devuelve un código de error.
- Al importar a una tabla que contiene una columna de indicación de fecha y hora de cambio de fila oculta implícitamente, no se cumplirá la propiedad oculta implícitamente de la columna. Por lo tanto el modificador de tipo de archivo rowchangetimestampmissing *debe* especificarse en el mandato de importación si los datos de la columna no existen en los datos que se van a importar y no existe ninguna lista de columna explícita.

Tabla 60. Comportamiento de LOAD al utilizar codepage y usegraphiccodepage

| codepage=N | usegraphiccodepage | Comportamiento de LOAD |
|------------|--------------------|--|
| Ausente | Ausente | Se supone que todos los datos del archivo están en la página de códigos de la base de datos, no en la página de códigos de la aplicación, aunque se especifique la opción CLIENT. |
| Presente | Ausente | Se supone que todos los datos del archivo están en la página de códigos N. Aviso: Los datos gráficos se corromperán cuando se carguen en la base de datos si N es una página de códigos de un solo byte. |

Tabla 60. Comportamiento de LOAD al utilizar codepage y usegraphiccodepage (continuación)

| codepage=N | usegraphiccodepage | Comportamiento de LOAD |
|------------|--------------------|--|
| Ausente | Presente | <p>Se supone que los datos de caracteres del archivo están en la página de códigos de la base de datos, aunque se especifique la opción CLIENT. Se supone que los datos gráficos están en la página de códigos de los datos gráficos de la base de datos, aunque se especifique la opción CLIENT.</p> <p>Si la página de códigos de la base de datos es de un solo byte, se supone que todos los datos están en la página de códigos de la base de datos.</p> <p>Aviso: Los datos gráficos se corromperán cuando se carguen en una base de datos de un solo byte.</p> |
| Presente | Presente | <p>Se presupone que los datos del carácter se encuentran en la página de códigos N. Se presupone que los datos gráficos se encuentran en la página de códigos gráficos N.</p> <p>Si N es una página de códigos de un solo byte o de doble byte, se supone que todos los datos están en la página de códigos N.</p> <p>Aviso: Los datos gráficos se corromperán cuando se carguen en la base de datos si N es una página de códigos de un solo byte.</p> |

Mandato PRUNE HISTORY/LOGFILE utilizando el procedimiento ADMIN_CMD

Se utiliza para suprimir entradas del archivo histórico de recuperación o para suprimir archivos de anotaciones cronológicas de la vía de acceso del archivo de anotaciones cronológicas activo de la partición de base de datos conectada en este momento. La supresión de entradas del archivo histórico de recuperación podría ser necesaria si el archivo crece excesivamente y el período de retención es largo.

Autorización

Una de las siguientes:

- *sysadm*
- *sysctrl*
- *sysmaint*
- *dbadm*

Conexión necesaria

Base de datos

Sintaxis del mandato



Parámetros del mandato

HISTORY *indicfechahora*

Identifica un rango de entradas del archivo histórico de recuperación que se suprimirán. Se puede especificar una indicación de fecha y hora completa (en el formato *aaaammddhhmmss*) o un prefijo inicial (mínimo *aaaa*). Todas las entradas con indicaciones de fecha y hora iguales o menores que la indicación de fecha y hora proporcionada se suprimirán del archivo histórico de recuperación.

WITH FORCE OPTION

Especifica que se podarán las entradas de acuerdo con la indicación de fecha y hora especificada, incluso si se suprimen del archivo algunas entradas del conjunto de restauración más reciente. Un conjunto de restauración es la copia de seguridad de base de datos completa más reciente que incluye las restauraciones de dicha imagen de copia de seguridad. Si no se especifica este parámetro, se mantendrán en el histórico todas las entradas realizadas desde la imagen de copia de seguridad en adelante.

AND DELETE

Especifica que los archivos de anotaciones asociados se suprimirán físicamente (de acuerdo con la información de la ubicación) cuando se elimine la entrada del archivo histórico. Esta opción resulta especialmente útil para garantizar la recuperación del espacio de almacenamiento de archivador cuando los archivos de anotaciones ya no se necesitan. Si está archivando anotaciones mediante un programa de salida de usuario, las anotaciones no se podrán suprimir utilizando esta opción.

Si establece el parámetro de configuración de base de datos **auto_del_rec_obj** en ON, la invocación de PRUNE HISTORY con el parámetro AND DELETE también suprimirá físicamente las imágenes de copia de seguridad de carga y las imágenes de copia de carga si se recorta la entrada del archivo histórico.

LOGFILE PRIOR TO *nombre-archivo-anotaciones*

Especifica una serie para un nombre de archivo de anotaciones; por ejemplo S0000100.LOG. Se suprimirán todos los archivos de anotaciones anteriores al archivo de anotaciones especificado (sin incluirlo). El parámetro de configuración de base de datos **logretain** debe establecerse en RECOVERY o CAPTURE.

Ejemplo

Ejemplo 1: eliminar todas las entradas del archivo histórico de recuperación escritas antes del 31 de diciembre de 2003 o ese mismo día:

```
CALL SYSPROC.ADMIN_CMD ('prune history 20031231')
```

Ejemplo 2: Suprimir todos los archivos de anotaciones cronológicas de la vía de acceso del archivo de anotaciones cronológicas activo anteriores a (pero sin incluir) S0000100.LOG:

```
CALL SYSPROC.ADMIN_CMD('prune logfile prior to S0000100.LOG')
```

Notas de uso

Si se utiliza WITH FORCE OPTION, podrían suprimirse entradas que son necesarias para la restauración automática de las bases de datos. Las restauraciones manuales seguirán funcionando correctamente. La utilización de este mandato

también puede evitar que el programa de utilidad db2ckrst pueda analizar correctamente la cadena completa de imágenes de copia de seguridad necesarias. El uso del mandato PRUNE HISTORY sin la opción WITH FORCE OPTION impide que se supriman las entradas necesarias.

Dichas entradas con el estado DB2HISTORY_STATUS_DO_NOT_DELETE no se recortarán. Si se utiliza la opción WITH FORCE OPTION, los objetos marcados como DB2HISTORY_STATUS_DO_NOT_DELETE se seguirán recortando o suprimiendo. Puede establecer el estado de las entradas del archivo histórico de recuperación en DB2HISTORY_STATUS_DO_NOT_DELETE utilizando el mandato UPDATE HISTORY, el mandato ADMIN_CMD con UPDATE_HISTORY o la API db2HistoryUpdate. Puede utilizar el estado DB2HISTORY_STATUS_DO_NOT_DELETE para impedir que se recorten entradas del archivo histórico de recuperación de claves y evitar que se supriman los objetos de recuperación asociados a ellas.

Puede recortar las entradas del archivo histórico de base de datos de copia de seguridad instantánea utilizando el mandato PRUNE HISTORY, pero no puede suprimir los objetos de recuperación físicos relacionados utilizando el parámetro AND DELETE. La manera de suprimir los objetos de copia de seguridad instantánea es utilizar el mandato db2acsutil.

El mandato solo afecta a la partición de base de datos a la que está conectada actualmente la aplicación.

Mandato QUIESCE DATABASE utilizando el procedimiento ADMIN_CMD

Fuerza que todos los usuarios salgan de la base de datos especificada y los pone en la modalidad inmovilizada. Mientras la base de datos esté en la modalidad inmovilizada, se pueden realizar tareas administrativas en ella. Después de completar las tareas administrativas, utilice el mandato UNQUIESCE para activar la base de datos y permitir que otros usuarios se conecten a ella sin tener que concluir e iniciar de nuevo la base de datos.

En esta modalidad, solo los usuarios que posean autorización en esta modalidad restringida tienen permiso para conectarse a la base de datos. Los usuarios que poseen la autorización *sysadm* y *dbadm* siempre tienen acceso a una base de datos mientras está inmovilizada.

Ámbito

QUIESCE DATABASE da como resultado que todos los objetos de la base de datos están en modalidad inmovilizada. Sólo el usuario/grupo permitido y *sysadm*, *sysmaint*, *dbadm* o *sysctrl* podrán acceder a la base de datos o sus objetos.

Autorización

Una de las siguientes:

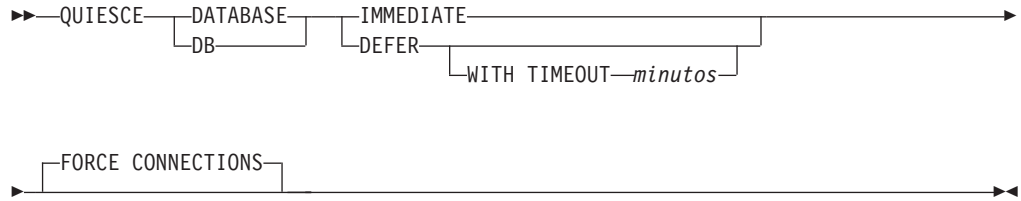
Para inmovilización a nivel de base de datos:

- *sysadm*
- *dbadm*

Conexión necesaria

Base de datos

Sintaxis del mandato



Parámetros del mandato

DEFER

Espera a las aplicaciones hasta que confirman la unidad de trabajo actual.

WITH TIMEOUT *minutos*

Especifica un período de tiempo, en minutos, que se deberá esperar para que las aplicaciones confirmen la unidad de trabajo actual. Si no se especifica ningún valor, en un entorno de bases de datos de una sola partición el valor por omisión será de 10 minutos. En un entorno de bases de datos particionadas, se utilizará el valor especificado por el parámetro de configuración de gestor de bases de datos **start_stop_time**.

IMMEDIATE

No espera a que se confirmen las transacciones, retrotrae las transacciones inmediatamente.

FORCE CONNECTIONS

Fuerza la desconexión de las conexiones.

DATABASE

Inmoviliza la base de datos. Todos los objetos de la base de datos se pondrán en la modalidad inmovilizada. Solo los usuarios especificados de grupos especificados y los usuarios con autorización *sysadm*, *sysmaint* y *sysctrl* podrán acceder a la base de datos o a sus objetos.

Ejemplo

Forzar la desconexión de todos los usuarios con conexiones a la base de datos.

```
CALL SYSPROC.ADMIN_CMD( 'quiesce db immediate' )
```

- Este mandato forzará a todos los usuarios a desconectarse de la base de datos si se proporciona la opción **FORCE CONNECTIONS**. **FORCE CONNECTIONS** es el comportamiento por omisión; el parámetro está permitido en el mandato por motivos de compatibilidad.
- El mandato se sincronizará con **FORCE CONNECTIONS** y solo se completará una vez que **FORCE CONNECTIONS** se haya completado.

Notas de uso

- Después de **QUIESCE DATABASE**, los usuarios con autorización *sysadm*, *sysmaint*, *sysctrl* o *dbadm* y privilegios **GRANT/REVOKE** podrán designar quién se podrá conectar. Esta información se almacenará de forma permanente en las tablas de catálogos de la base de datos.

Por ejemplo,

```
grant quiesce_connect on database to <username/groupname>
revoke quiesce_connect on database from <username/groupname>
```

- El estado de la ejecución del mandato se devuelve en el área de comunicaciones SQL (SQLCA) obtenida a partir de la sentencia CALL.

Mandato QUIESCE TABLESPACES FOR TABLE utilizando el procedimiento ADMIN_CMD

Inmoviliza los espacios de tablas para una tabla. Existen tres modalidades de inmovilización válidas: compartimiento, intención de actualización y exclusiva. La función de inmovilización provoca tres estados posibles:

- Inmovilizado: SHARE
- Inmovilizado: UPDATE
- Inmovilizado: EXCLUSIVE

Ámbito

En un entorno de una sola partición, este mandato inmoviliza todos los espacios de tablas implicados en una operación de carga en modalidad exclusiva durante el tiempo que dura la operación de carga. En un entorno de bases de datos particionadas, este mandato actúa localmente en una partición de base de datos. Sólo inmoviliza la parte de espacios de tablas que pertenecen a la partición de base de datos en la que se realiza la operación de carga. En el caso de las tablas particionadas, se inmovilizan todos los espacios de tablas que figuran en SYSDATAPARTITIONS.TBSPACEID y SYSDATAPARTITIONS.LONG_TBSPACEID asociados a una tabla y cuyo estado sea normal, conectados o desconectados (por ejemplo, SYSDATAPARTITIONS.STATUS de 'N', 'A' o 'D', respectivamente).

Autorización

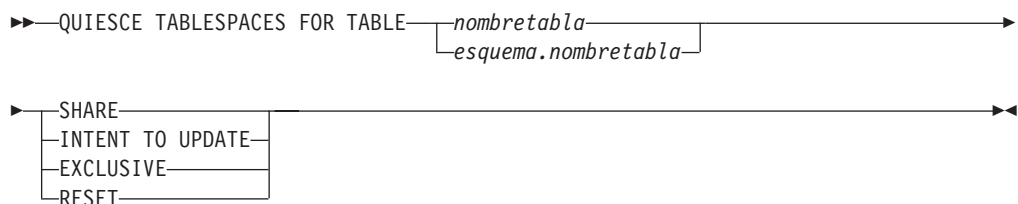
Una de las siguientes:

- *sysadm*
- *sysctrl*
- *sysmaint*
- *dbadm*
- *load*

Conexión necesaria

Base de datos

Sintaxis del mandato



Parámetros del mandato

TABLE

nombretabla

Especifica el nombre de tabla no calificado. La tabla no puede ser una tabla de catálogos del sistema.

esquema.nombre-tabla

Especifica el nombre de tabla calificado. Si no se proporciona *esquema*, se utilizará CURRENT SCHEMA. La tabla no puede ser una tabla de catálogos del sistema.

SHARE

Especifica que la inmovilización debe ser en modalidad de compartimiento.

Cuando se realiza una petición de "compartimiento de inmovilización", la transacción solicita intentar bloqueos de compartimiento para los espacios de tablas y un bloqueo de compartimiento para la tabla. Cuando la transacción obtiene los bloqueos, el estado de los espacios de tablas cambia a QUIESCED SHARE. El estado sólo se otorga al inmovilizador si no hay ningún estado en conflicto mantenido por otros usuarios. El estado de los espacios de tablas, junto con el ID de autorización y el ID de agente de base de datos del inmovilizador, se registran en la tabla de espacios de tablas, para que el estado sea permanente. No se puede cambiar la tabla mientras los espacios de tablas para la tabla están en estado QUIESCED SHARE. Se permiten otras peticiones de modalidad de compartimiento en la tabla y los espacios de tablas. Cuando la transacción se confirma o se retrotrae, se liberan los bloqueos, pero los espacios de tablas para la tabla permanecen en estado QUIESCED SHARE hasta que se restablece explícitamente el estado.

INTENT TO UPDATE

Especifica que la inmovilización debe ser en modalidad de intento de actualización.

Cuando se realiza una petición "intento de actualización de inmovilización", los espacios de tablas se bloquean en modalidad de intento exclusivo (IX) y la tabla se bloquea en modalidad de actualización (U). El estado de los espacios de tablas se registra en la tabla de espacios de tablas.

EXCLUSIVE

Especifica que la inmovilización debe estar en modalidad exclusiva.

Cuando se realiza una petición de "inmovilización exclusiva", la transacción solicita bloqueos superexclusivos en los espacios de tablas y un bloqueo superexclusivo en la tabla. Cuando la transacción obtiene los bloqueos, el estado de los espacios de tablas cambia a QUIESCED EXCLUSIVE. El estado de los espacios de tablas, junto con el ID de autorización y el ID de agente de base de datos del inmovilizador, se registran en la tabla de espacios de tablas. Dado que los espacios de tablas se mantienen en modalidad superexclusiva, no se permite ningún otro acceso a los espacios de tablas. El usuario que invoca la función de inmovilización (el inmovilizador) tiene acceso exclusivo a la tabla y a los espacios de tablas.

RESET

Especifica que el estado de los espacios de tablas debe restablecerse a

normal. El estado de inmovilización no se puede restablecer si la conexión que emitió la petición de inmovilizar todavía está activa.

Ejemplo

Inmovilizar los espacios de tablas que contienen la tabla de personal.

```
CALL SYSPROC.ADMIN_CMD( 'quiesce tablespaces for table staff share' )
```

Notas de uso

Este mandato no se soporta para tablas declaradas temporales.

Una inmovilización es un bloqueo permanente. Su ventaja consiste en que perduran las anomalías de las transacciones, las anomalías de conexión e incluso las anomalías del sistema (por ejemplo anomalía de alimentación o rearmar).

Una inmovilización es propiedad de una conexión. Si se pierde la conexión, la inmovilización permanece, pero no tiene propietario y se denomina *inmovilización ficticia*. Por ejemplo, si un paro de alimentación ha hecho que se interrumpiera una operación de carga durante la fase de supresión, los espacios de tablas para la tabla cargada quedarán en estado de inmovilización exclusiva pendiente de supresión. Al reiniciarse la base de datos, esta inmovilización será una inmovilización sin propietario (o ficticia). Para eliminar de una inmovilización ficticia se necesita una conexión con el mismo ID de usuario que se haya utilizado al establecer la modalidad de inmovilización.

Para eliminar una inmovilización ficticia:

1. Conéctese a la base de datos con el mismo ID de usuario utilizado al establecer la modalidad de inmovilización.
2. Utilice el mandato LIST TABLESPACES para determinar qué espacio de tablas está inmovilizado.
3. Vuelva a inmovilizar el espacio de tablas utilizando el estado de inmovilización actual. Por ejemplo:

```
CALL SYSPROC.ADMIN_CMD('quiesce tablespaces for table mytable exclusive' )
```

Una vez realizada, la nueva conexión es propietaria de la inmovilización y se puede reiniciar la operación de carga.

Existe un límite de cinco inmovilizadores en un espacio de tablas en un momento determinado.

Un inmovilizador puede actualizar el estado de un espacio de tablas de un estado menos restrictivo a otro más restrictivo (por ejemplo, S a U o U a X). Si un usuario solicita un estado más bajo que uno que ya se mantiene, se vuelve al estado original. Los estados no pueden disminuir de nivel.

El estado de la ejecución del mandato se devuelve en el área de comunicaciones SQL (SQLCA) obtenida a partir de la sentencia CALL.

Mandato REDISTRIBUTE DATABASE PARTITION GROUP utilizando el procedimiento ADMIN_CMD

Redistribuye datos en las particiones de base de datos de un grupo de particiones de base de datos. La distribución de datos de destino puede ser uniforme (valor por omisión) o puede especificarla el usuario para satisfacer determinados requisitos del sistema.

El mandato REDISTRIBUTE DATABASE PARTITION GROUP redistribuye datos en todas las particiones de un grupo de particiones de base de datos. Esto afecta a todos los objetos presentes en el grupo de particiones de base de datos y no puede restringirse a un único objeto.

Ámbito

Este mandato afecta a todas las particiones de base de datos del grupo de particiones de base de datos.

Autorización

Se necesita una de las autorizaciones siguientes:

- SYSADM
- SYCTRL
- DBADM

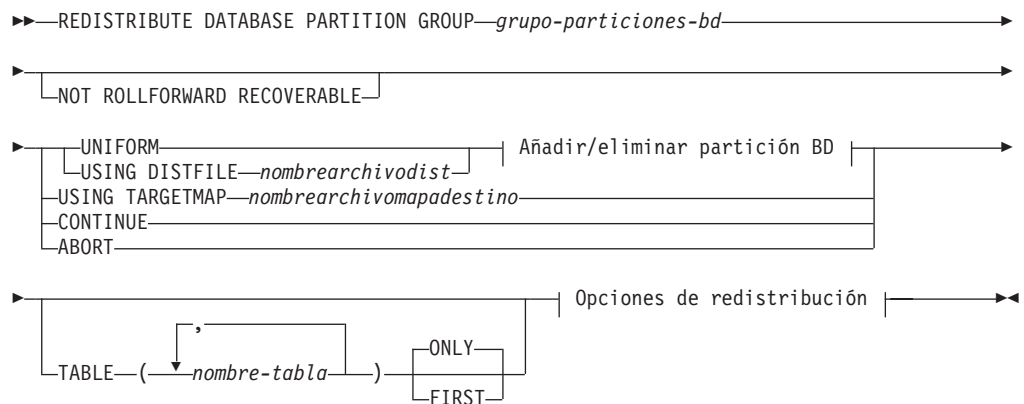
Asimismo, también es necesario uno de los grupos de autorizaciones siguientes:

- Privilegios DELETE, INSERT y SELECT en todas las tablas del grupo de particiones de base de datos que se está redistribuyendo
- Autorización DATAACCESS

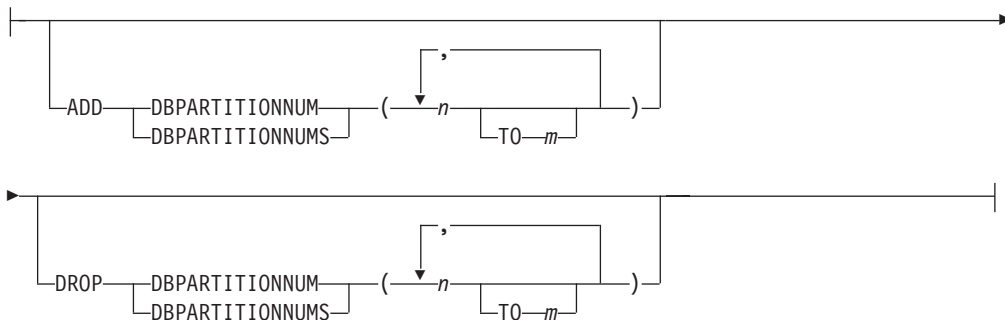
Conexión necesaria

Conexión a la partición del catálogo.

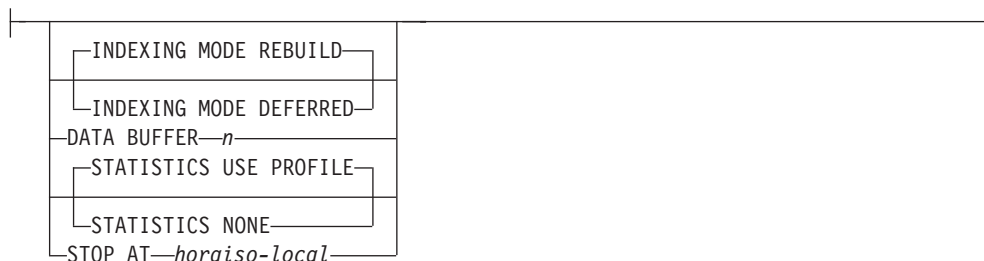
Sintaxis del mandato



Añadir/eliminar partición BD:



Opciones de redistribución:



Parámetros del mandato

DATABASE PARTITION GROUP *grupo-particiones-bd*

Nombre del grupo de particiones de base de datos. Este nombre de una sola parte identifica un grupo de particiones de base de datos descrito en la tabla de catálogos SYSCAT.DBPARTITIONGROUPS. El grupo de particiones de base de datos no puede estar sometido actualmente a una redistribución.

Nota: Las tablas de los grupos de particiones de base de datos IBMGROUP y IBMTEMPGROUP no se pueden redistribuir.

NOT ROLLFORWARD RECOVERABLE

Cuando se utiliza esta opción, el mandato `REDISTRIBUTE DATABASE PARTITION GROUP` no se puede recuperar en avance.

- Los datos se mueven masivamente en lugar de mediante operaciones de inserción y supresión internas. Esto reduce el número de veces que se debe explorar una tabla y acceder a ella, lo que da como resultado un mejor rendimiento.
- Ya no se necesitan registros de anotaciones para cada una de las operaciones de inserción y supresión. Esto significa que ya no necesita gestionar grandes cantidades de espacio de registro activo y espacio de archivado de registro en el sistema al realizar la redistribución de datos. Esto es especialmente beneficioso si, en el pasado, un espacio de registro activo grande y los requisitos de almacenamiento forzaban a dividir una sola operación de redistribución de datos en múltiples tareas de redistribución más pequeñas, para lo que podría necesitar incluso más tiempo para completar la operación de redistribución de datos de un extremo a otro.
- Al utilizar el mandato `REDISTRIBUTE DATABASE PARTITION GROUP` con la opción `NOT ROLLFORWARD RECOVERABLE`, la operación de redistribución utiliza la opción `INDEXING MODE DEFERRED` para las

tablas que contienen columnas XML. Si una tabla no contiene una columna XML, la operación de redistribución utiliza la modalidad de creación de índices especificada al emitir el mandato.

Cuando *no* se utiliza esta opción, se realiza un registro extensivo de todo el movimiento de filas que se efectúa para que se pueda recuperar la base de datos posteriormente en caso de que se produzca una interrupción, un error o por otra necesidad de la actividad.

UNIFORM

Especifica que los datos se distribuyen uniformemente entre las particiones hash (es decir, se supone que cada partición hash tiene el mismo número de filas), pero no se correlaciona el mismo número de particiones hash con cada partición de base de datos. Después de la redistribución, todas las particiones de base de datos del grupo de particiones de base de datos tienen aproximadamente el mismo número de particiones hash.

USING DISTFILE *nombrearchivodist*

Si se desvía la distribución de los valores clave de distribución, utilice esta opción para obtener una redistribución uniforme de los datos entre las particiones de base de datos de un grupo de particiones de base de datos.

Utilice el *nombrearchivodist* para indicar la distribución actual de los datos entre las 32.768 particiones hash.

Utilice cuentas de filas, volúmenes de bytes o cualquier otra medida para indicar la cantidad de datos representados por cada partición hash. El programa de utilidad lee el valor entero asociado con una partición como el valor relativo de dicha partición. Cuando se especifica un *nombrearchivodist*, el programa de utilidad genera una correlación de distribución de destino y la utiliza para redistribuir los datos entre las particiones de base de datos del grupo de particiones de base de datos de la manera más uniforme posible. Después de la redistribución, el valor relativo de cada partición de base de datos del grupo de particiones de base de datos es aproximadamente el mismo (el valor relativo de una partición de base de datos es la suma de los valores relativos de todas las particiones hash que se correlacionan con dicha partición de base de datos).

Por ejemplo, el archivo de distribución de entrada podría contener entradas como las siguientes:

```
10223
1345
112000
0
100
...
```

En el ejemplo, la partición hash 2 tiene un valor relativo de 112.000 y la partición 3 (con un valor relativo de 0) no tiene correlación de datos con ella en absoluto.

El *nombrearchivodist* debe contener 32.768 valores enteros positivos en formato de carácter. La suma de los valores debe ser inferior o igual a 4.294.967.295.

Hay que incluir el nombre de vía de acceso completo del *nombrearchivodist*, y el *nombrearchivodist* debe existir en el servidor y ser accesible desde la partición conectada.

USING TARGETMAP *nombrearchivomapadestino*

El archivo especificado en *nombrearchivomapadestino* se utiliza como

correlación de distribución de destino. La redistribución de datos se realiza de acuerdo con este archivo. Hay que incluir el nombre de vía de acceso completo del *nombreamapadestino* y el *nombreamapadestino* debe existir en el servidor y ser accesible desde la partición conectada.

Si una partición de base de datos incluida en la correlación de destino no está en el grupo de particiones de base de datos, se devolverá un error. Emita la sentencia ALTER DATABASE PARTITION GROUP ADD DBPARTITIONNUM antes de ejecutar el mandato REDISTRIBUTE DATABASE PARTITION GROUP.

Si una partición de base de datos, excluida de la correlación de destino, *está* en el grupo de particiones de base de datos, esa partición de base de datos no se incluirá en el particionamiento. Dicha partición de base de datos se puede descartar utilizando la sentencia ALTER DATABASE PARTITION GROUP DROP DBPARTITIONNUM antes o después del mandato REDISTRIBUTE DATABASE PARTITION GROUP.

CONTINUE

Continúa una operación REDISTRIBUTE DATABASE PARTITION GROUP que se ha detenido o ha fallado anteriormente. Si no se ha producido ninguna, se devuelve un error.

ABORT

Cancela una operación REDISTRIBUTE DATABASE PARTITION GROUP que se ha detenido o ha fallado anteriormente. Si no se ha producido ninguna, se devuelve un error.

ADD

DBPARTITIONNUM *n*

TO *m*

n o *n TO m* especifica una lista o listas de números de particiones de base de datos que deben añadirse al grupo de particiones de base de datos. Las particiones especificadas no deben estar ya definidas en el grupo de particiones de base de datos (SQLSTATE 42728). Es equivalente a ejecutar la sentencia ALTER DATABASE PARTITION GROUP con la cláusula ADD DBPARTITIONNUM especificada.

DBPARTITIONNUMS *n*

TO *m*

n o *n TO m* especifica una lista o listas de números de particiones de base de datos que deben añadirse al grupo de particiones de base de datos. Las particiones especificadas no deben estar ya definidas en el grupo de particiones de base de datos (SQLSTATE 42728). Es equivalente a ejecutar la sentencia ALTER DATABASE PARTITION GROUP con la cláusula ADD DBPARTITIONNUM especificada.

Nota: Cuando se añade una partición de base de datos utilizando esta opción, los contenedores de espacios de tablas se basan en los contenedores del espacio de tablas correspondiente en la partición existente con el número más bajo del grupo de particiones de la base de datos. Como resultado se obtendría un conflicto de nombres entre los contenedores, algo que podría suceder si las

nuevas particiones se realizan en la misma máquina física que los contenedores existentes; esta opción no debe utilizarse. En su lugar, debe utilizarse la sentencia ALTER DATABASE PARTITION GROUP con la opción WITHOUT TABLESPACES antes de emitir el mandato REDISTRIBUTE DATABASE PARTITION GROUP. A continuación, los contenedores de espacios de tablas se pueden crear manualmente especificando nombres apropiados.

DROP

DBPARTITIONNUM *n*

TO *m*

n o *n* **TO** *m* especifica una lista o listas de números de particiones de base de datos que deben descartarse del grupo de particiones de base de datos. Las particiones especificadas no deben estar ya definidas en el grupo de particiones de base de datos (SQLSTATE 42729). Esto equivale a ejecutar la sentencia ALTER DATABASE PARTITION GROUP con la cláusula DROP DBPARTITIONNUM especificada.

DBPARTITIONNUMS *n*

TO *m*

n o *n* **TO** *m* especifica una lista o listas de números de particiones de base de datos que deben descartarse del grupo de particiones de base de datos. Las particiones especificadas no deben estar ya definidas en el grupo de particiones de base de datos (SQLSTATE 42729). Esto equivale a ejecutar la sentencia ALTER DATABASE PARTITION GROUP con la cláusula DROP DBPARTITIONNUM especificada.

TABLE *nombretabla*

Especifica un orden de tabla para el proceso de redistribución.

ONLY Si el orden de tabla va seguido por la palabra clave **ONLY** (que es el valor por omisión), entonces, sólo se redistribuirán las tablas especificadas. El resto de las tablas se puede procesar más adelante mediante mandatos posteriores REDISTRIBUTE CONTINUE. Es el valor por omisión.

FIRST Si el orden de tabla va seguido por la palabra clave **FIRST**, entonces, se distribuirán las tablas especificadas en el orden indicado y el resto de tablas del grupo de particiones de base de datos se redistribuirán en orden aleatorio.

INDEXING MODE

Este parámetro especifica cómo se mantienen los índices durante la redistribución cuando se especifica la opción NOT ROLLFORWARD RECOVERABLE. Los valores válidos son:

REBUILD

Los índices se volverán a crear desde el principio. No es necesario que los índices sean válidos para utilizar esta opción. Como resultado de esto, las páginas de índices se agruparán juntas en el disco.

DEFERRED

El mandato Redistribute no intentará mantener ningún índice. Los

índices se marcarán como que necesitan una renovación. El primer acceso a dichos índices puede que fuerce a volver a crear, o puede que se vuelvan a crear los índices cuando se reinicie la base de datos.

Nota: Para las tablas no MDC, si hay índices no válidos en las tablas, el mandato REDISTRIBUTE DATABASE PARTITION GROUP los volverá a crear automáticamente si no se especifica **INDEXING MODE DEFERRED**. Para una tabla MDC, aunque se especifique **INDEXING MODE DEFERRED**, se vuelve a crear un índice compuesto que no es válido antes de que empiece la redistribución porque el programa de utilidad necesita que el índice compuesto procese una tabla MDC.

DATA BUFFER *n*

Especifica el número de páginas de 4 KB que se utilizan como espacio de almacenamiento intermedio para transferir datos dentro del programa de utilidad. Si el valor especificado es menor que el valor mínimo soportado, se utiliza el valor mínimo y no se devolverá ningún aviso. Esta memoria se asigna directamente desde el almacenamiento dinámico del programa de utilidad, cuyo tamaño puede modificarse mediante el parámetro de configuración de base de datos **util_heap_sz**. Si no se especifica un valor, el programa de utilidad calcula un valor por omisión inteligente durante la ejecución al principio del proceso de cada tabla. De forma más específica, el valor por omisión es utilizar el 50% de la memoria disponible en la pila del programa de utilidad en el momento en que empieza la redistribución de la tabla y tener en cuenta también diversas propiedades de dicha tabla.

STOP AT *horaiso-local*

Cuando se especifica esta opción, antes de empezar la redistribución de datos para cada tabla, *horaiso-local* se comparará con la indicación de fecha y hora local actual. Si la *horaiso-local* es igual o anterior a la indicación de fecha y hora local actual, el programa de utilidad se detendrá con un mensaje de error. El proceso de redistribución de datos de las tablas en progreso en el momento de la detención se completará sin interrupción. No empezará ningún proceso de redistribución de datos de las tablas nuevo. Las tablas no procesadas se pueden redistribuir utilizando la opción **CONTINUE**. Este valor *horaiso-local* se especifica como indicación de fecha y hora, una serie de caracteres de 7 partes que identifica una fecha y hora combinadas. El formato es *aaaa-mm-dd-hh.mm.ss.nnnnnn* (año, mes, día, hora, minutos, segundos, microsegundos), expresado en hora local.

STATISTICS

Esta opción especifica que el programa de utilidad debe recopilar estadísticas para las tablas que tengan un perfil de estadísticas. Especificar esta opción es más eficaz que emitir por separado el mandato RUNSTATS después de que se haya completado la redistribución.

USE PROFILE

Se recopilarán estadísticas para las tablas con un perfil de estadísticas. Para las tablas sin un perfil de estadísticas, no se llevará a cabo ninguna acción. Es el valor por omisión.

NONE

No se recopilarán estadísticas para las tablas.

Ejemplos: pasos de redistribución

Es posible que desee añadir o descartar un nodo de un grupo de nodos. A continuación se indican los pasos para añadir un nodo nuevo a un grupo de nodos y redistribuir los datos. La partición de base de datos no está en la correlación de distribución, pero los contenedores correspondientes a los espacios de tablas del grupo de particiones de base de datos se han creado; la partición de base de datos se añade a la correlación de distribución cuando la operación de redistribución del grupo de particiones de base de datos finaliza satisfactoriamente.

1. Identifique los grupos de nodos que necesitarán redistribuirse. En este documento, el grupo de nodos que se tiene que redistribuir es "sampleNodegrp".
2. Identifique los objetos que deberían deshabilitarse o eliminarse antes de la redistribución.
 - a. Replique las MQT: no se da soporte a este tipo de MQT como parte del programa de utilidad REDISTRIBUTE. Debe descartarlas antes de ejecutar la redistribución y volver a crearlas posteriormente.

```
SELECT tabschema, tabname
FROM syscat.tables
WHERE partition_mode = 'R'
```

- b. Supervisores de sucesos de escritura en tabla: debe inhabilitar los supervisores de sucesos de escritura en tabla activados automáticamente que contengan una tabla que resida en el grupo de particiones de base de datos que se va a redistribuir.

```
SELECT distinct evmonname
FROM syscat.eventtables E
JOIN syscat.tables T on T.tabname = E.tabname
AND T.tabschema = E.tabschema
JOIN syscat.tablespace S on S.tbSPACE = T.tbSPACE
AND S.ngname = 'sampleNodegrp'
```

- c. Tablas de Explain: se recomienda crear las tablas de Explain en un grupo de nodos de partición única. Sin embargo, si están definidos en un grupo de nodos que requiere redistribución, puede preferir descartarlos antes de la redistribución y volver a definirlos una vez terminada la redistribución, en el caso de que no sea necesario mantener los datos generados hasta la fecha.
- d. Modalidad de acceso a la tabla y estado de la carga: asegúrese de que todas las tablas de los grupos de nodos que se van a redistribuir tienen modalidad de acceso completo y no tienen el estado de carga pendiente o carga en proceso.

```
SELECT distinct trim(T.creator) || '\'.' || trim(T.name)
AS name, T.access_mode, A.load_status
FROM sysibm.systables T, sysibm.sysnodegroups N, sysibmadm.admintabinfo A
WHERE T.pmap_id = N.pmap_id
AND A.tabschema = T.creator
AND A.tabname = T.name
AND N.name = 'sampleNodegrp'
AND (T.access_mode <> 'F' or A.load_status is not null)
```

- e. Perfiles de estadísticas: las estadísticas de tabla se pueden actualizar como parte del proceso de redistribución si se ha definido un perfil de estadísticas para la tabla. Al utilizar el programa de utilidad REDISTRIBUTE para actualizar las estadísticas de una tabla se reduce la E/S, dado que se exploran todos los datos para la redistribución y no es necesario ejecutar exploraciones adicionales de los datos para RUNSTATS.

```

RUNSTATS on table schema.table
USE PROFILE runstats_profile
SET PROFILE ONLY

```

3. Revise la configuración de la base de datos. **util_heap_sz** es esencial para procesar el traslado de datos entre particiones de base de datos. Asigne el máximo de memoria posible a **util_heap_sz** mientras dure la redistribución. Es necesario disponer de un **sortheap** suficiente si se realiza la recreación de los índices como parte de la redistribución. Aumente **util_heap_sz** y **sortheap** según sea necesario para mejorar el rendimiento de la redistribución.
4. Recupere los valores de configuración de la base de datos que se usarán para las nuevas particiones de base de datos. Al añadir particiones de base de datos, se emplea la configuración de base de datos por omisión. Por ello, es importante actualizar la configuración de base de datos en los nodos nuevos antes de emitir el mandato REDISTRIBUTE para garantizar que la configuración queda equilibrada en todo el almacén.

```

SELECT name,
CASE WHEN deferred_value_flags = 'AUTOMATIC'
THEN deferred_value_flags
ELSE substr(deferred_value,1,20)
END AS deferred_value
FROM sysibmadm.dbcfg
WHERE dbpartitionnum = nodo-existente
AND deferred_value != ''
AND name NOT IN ('hadr_local_host','hadr_local_svc','hadr_peer_window',
'hadr_remote_host','hadr_remote_inst','hadr_remote_svc',
'hadr_syncmode','hadr_timeout','backup_pending','codepage',
'codeset','collate_info','country','database_consistent',
'database_level','hadr_db_role','log_retain_status',
'loghead','logpath','multipage_alloc','numsegs','pagesize',
'release','restore_pending','restrict_access',
'rollfwd_pending','territory','user_exit_status',
'number_compat','varchar2_compat','database_memory')

```

5. Antes de iniciar el proceso de redistribución, realice una copia de seguridad de la base de datos (o de los espacios de tablas de los grupos de nodos que se van a redistribuir) para asegurarse de disponer de un punto de recuperación reciente.
6. Defina las BCU de datos nuevas en DB2 actualizando para ello el archivo `db2nodes.cfg` y agregando las especificaciones de partición de base de datos de las BCU de datos nuevas, y defina las particiones de base de datos nuevas para DB2 con el mandato `ADD NODE WITHOUT TABLESPACES`.

```

db2start nodenum x export DB2NODE=x
db2 add node without tablespaces
db2stop nodenum x

```

Nota: si no es el primer puerto lógico en la BCU de datos, ejecute el arranque y la parada del primer número de puerto lógico antes y después de la secuencia de mandatos anterior para determinar los puertos lógicos posteriores.

7. Defina los contenedores de espacio de tablas temporal del sistema en las particiones de base de datos que acaba de definir.

```

ALTER TABLESPACE nombre_espacio_tablas
ADD información_contenedor
ON dbpartitionnums (x to y)

```

8. Añada las particiones de base de datos lógicas nuevas a los grupos de particiones de base de datos que abarcan las BCU de datos.

```

ALTER DATABASE PARTITION GROUP nombre_espacio_tablas
ADD dbpartitionnums (x to y)
WITHOUT TABLESPACES

```


- Defina los contenedores de espacio de tablas de datos permanentes en las particiones de base de datos que acaba de definir.

```
ALTER TABLESPACE nombre_espacio_tablas
  ADD información_contenedor
  ON dbpartitionnums (x to y)
```

- Aplique los valores de configuración de base de datos recuperados en el paso 4 a las particiones de base de datos nuevas (o emita un único mandato UPDATE DB CFG para todas las particiones de base de datos que utilizan el nuevo soporte para vista única de configuración de DB2 9.5).
- Capture la definición de las MQT replicadas existentes en los grupos de particiones de base de datos que se van a redistribuir y descártelas.

```
db2look -d nombrebd -e -z
  esquema -t nombres_tablas_MQT_replicadas
  -o repMQTs.clp
```

- Deshabilite todos los supervisores de sucesos de escritura en tabla existentes en los grupos de particiones de base de datos que se van a redistribuir.

```
SET EVENT MONITOR nombre_supervisor STATE 0
```

- Ejecute el programa de utilidad REDISTRIBUTE para realizar la redistribución uniforme en todas las particiones de base de datos. A continuación se muestra el mandato de redistribución sencillo:

```
REDISTRIBUTE DATABASE PARTITION GROUP sampleNodegrp
  NOT ROLLFORWARD RECOVERABLE uniform;
```

Los usuarios deberán considerar la posibilidad de especificar una lista de tablas como entrada del mandato REDISTRIBUTE para determinar el orden en el que se procesarán las tablas. El programa de utilidad REDISTRIBUTE trasladará los datos (comprimidos y compactos). Opcionalmente, se volverán a crear los índices y se actualizarán las estadísticas, si se han definido los perfiles de estadísticas. Por consiguiente, puede ejecutarse el script siguiente en lugar del mandato anterior:

```
REDISTRIBUTE DATABASE PARTITION GROUP sampleNodegrp
  NOT ROLLFORWARD RECOVERABLE uniform
  TABLE (tab1, tab2,...) FIRST;
```

Consecuencias de utilizar la opción NOT ROLLFORWARD RECOVERABLE

Cuando se emite el mandato REDISTRIBUTE DATABASE PARTITION GROUP y se especifica la opción NOT ROLLFORWARD RECOVERABLE, se utiliza una estrategia de registro mínimo que minimiza la grabación de registros de anotación para cada fila que se mueve. Este tipo de registro cronológico es importante para el uso de la operación de redistribución ya que un modelo que registre por completo todo el movimiento de datos podría, para sistemas grandes, exigir una cantidad demasiado grande de espacio de registro activo y permanente, y por lo general, tendría menos rendimiento. No obstante, es importante que los usuarios tengan en cuenta que como resultado de este modelo de registro cronológico mínimo, el mandato REDISTRIBUTE DATABASE PARTITION GROUP *no* es recuperable en avance. Esto significa que en cualquier operación que como resultado se obtenga un avance de la base de datos mediante una operación de redistribución todas las tablas implicadas en la operación de redistribución quedarán en el estado UNAVAILABLE (no disponible). Estas tablas sólo se pueden descartar, lo que significa que no hay ningún modo de recuperar los datos de estas tablas. Esta es la razón por la que, para las bases de datos recuperables, el programa de utilidad REDISTRIBUTE DATABASE PARTITION GROUP, cuando se emite con la opción NOT ROLLFORWARD RECOVERABLE, pone todos los espacios de tablas que toca en estado de copia de seguridad pendiente, lo que fuerza al usuario a hacer copia

de seguridad de todos los espacios de tablas redistribuidos al final de una operación de redistribución satisfactoria. Con una copia de seguridad realizada después de la operación de redistribución, el usuario no debería tener necesidad de avanzar por la propia operación de redistribución.

Hay una consecuencia muy importante de la falta de recuperabilidad en avance del programa de utilidad de redistribución que el usuario debería tener presente: si el usuario decide permitir actualizaciones en las tablas de la base de datos (incluso las tablas ajenas al grupo de partición de la base de datos que se está redistribuyendo) mientras se ejecuta la operación de redistribución, incluido el periodo al final de la redistribución en que el usuario realiza una copia de seguridad de los espacios de tablas afectados por la redistribución, dichas actualizaciones se pueden perder en el caso de una avería grave; por ejemplo, si se destruye un contenedor de bases de datos. La razón de dichas actualizaciones es que la operación de redistribución no es recuperable en avance. Si fuera necesario restaurar la base de datos de una copia de seguridad realizada antes de la operación de redistribución, no sería posible realizar un avance por las anotaciones cronológicas con el fin de reproducir las actualizaciones que se realizaron durante la operación de redistribución sin avanzar tampoco por la redistribución que, tal y como se ha descrito antes, deja las tablas redistribuidas en el estado UNAVAILABLE (no disponible). Por lo tanto, lo único que se puede hacer en este caso es restaurar la base de datos de la copia de seguridad realizada antes de la redistribución sin avanzar. A continuación, se puede volver a realizar la operación de redistribución. Desafortunadamente, todas las actualizaciones producidas durante la operación de distribución original se pierden.

La importancia de este punto no puede exagerarse. Para poder estar seguro de que no se perderán actualizaciones durante una operación de redistribución, debe cumplirse una de las situaciones siguientes:

- El usuario evita realizar actualizaciones durante la operación del mandato REDISTRIBUTE DATABASE PARTITION GROUP, inclusive el periodo después de que finalice el mandato, en el que se realizan copias de seguridad de los espacios de tablas afectados.
- Las actualizaciones que se aplican durante la operación de redistribución proceden de una fuente repetible, lo que significa que se pueden volver a aplicar en cualquier momento. Por ejemplo, si la fuente de las actualizaciones son los datos que están almacenados en un archivo y se aplican actualizaciones durante el proceso por lotes, entonces claramente, incluso en el caso de una anomalía que exija una restauración de la base de datos, las actualizaciones no se perderían ya que simplemente podrían aplicarse de nuevo en cualquier momento.

Por lo que respecta a permitir actualizaciones en la base de datos durante la operación de redistribución, el usuario debe decidir si estas actualizaciones son adecuadas o no en su caso basándose en si las actualizaciones pueden o no repetirse después de una restauración de la base de datos, si es necesario.

Nota: No es cierto que cada anomalía durante la operación del mandato REDISTRIBUTE DATABASE PARTITION GROUP genere este problema. De hecho, la mayoría no. El mandato REDISTRIBUTE DATABASE PARTITION GROUP es completamente reinicializable, lo que significa que si el programa de utilidad falla en medio del trabajo, se puede continuar o bien terminar anormalmente con facilidad mediante las opciones **CONTINUE** o **ABORT**. Las anomalías mencionadas arriba son anomalías que requieren que el usuario realice una restauración a partir de la copia de seguridad realizada antes de la operación de redistribución.

Ejemplos

Redistribuir el grupo de particiones de base de datos DBPG_1 proporcionando la distribución actual de datos mediante un archivo de distribución de datos, `distfile_for_dbpg_1`, y desplazando los datos a dos nuevas particiones de base de datos, 6 y 7.

```
CALL SYSPROC.ADMIN_CMD('REDISTRIBUTE DATABASE PARTITION GROUP DBPG_1
  USING DISTFILE /home/user1/data/distfile_for_dbpg_1
  ADD DATABASE PARTITION (6 TO 7)')
```

Notas de uso

- Cuando se especifica la opción **NOT ROLLFORWARD RECOVERABLE** y la base de datos es recuperable, la primera vez que el programa de utilidad accede a un espacio de tablas, se pone en estado **BACKUP PENDING**. Todas las tablas de ese espacio de tablas pasarán a ser de sólo lectura hasta que se haga una copia de seguridad del espacio de tablas, que sólo puede hacerse cuando todas las tablas de un espacio de tablas han sido redistribuidas.
- Cuando se ejecuta una operación de redistribución, produce un archivo de registro de sucesos que contiene información general sobre la operación de redistribución e información sobre la hora inicial y final de cada tabla procesada. Este archivo de registro de sucesos se graba en el servidor:

- Las opciones El directorio `homeinst/sqllib/redist` de los sistemas Linux y UNIX, utilizando el formato siguiente para los subdirectorios y el nombre de archivo: `nombre-base-datos.nombre-grupo-partición-base-datos.indicación-fecha-hora.log`.
- El directorio `DB2INSTPROF\instance\redist` de los sistemas operativos Windows (donde `DB2INSTPROF` es el valor de la variable de registro `DB2INSTPROF`), utilizando el formato siguiente para los subdirectorios y el nombre de archivo: `nombre-base-datos.nombre-grupo-partición-base-datos.indicación-fecha-hora.log`.
- El valor de indicación de fecha y hora es la hora en la que se emitió el mandato.

Para obtener más información sobre el archivo de anotaciones cronológicas de sucesos de redistribución, consulte el tema “Archivo de registro de sucesos de redistribución”.

- Este programa de utilidad realiza operaciones **COMMIT** intermitentes durante el proceso. Esto puede hacer que las conexiones de tipo 2 reciban un error `SQL30090N`.
- Todos los paquetes que tienen una dependencia en una tabla que ha sido sometida a una redistribución se invalidan. Se recomienda volver a vincular explícitamente dichos paquetes después de que se haya completado la operación de redistribución de grupo de particiones de base de datos. La revinculación explícita elimina el retardo inicial en la ejecución de la primera petición SQL para el paquete no válido. El archivo de mensajes de redistribución contiene una lista de todas las tablas que han sido sometidas a redistribución.
- Por omisión, el programa de utilidad de redistribución actualizará las estadísticas para las tablas que tengan un perfil de estadísticas. Para las tablas sin un perfil de estadísticas, se recomienda actualizar por separado la tabla y las estadísticas de índice de estas tablas, llamando a la API `db2Runstats` o emitiendo el mandato `RUNSTATS` después de que se haya completado la operación de redistribución.
- Los grupos de particiones de base de datos que contienen tablas de consulta materializadas duplicadas o tablas definidas con `DATA CAPTURE CHANGES` no se pueden redistribuir.

- La redistribución no está permitida si existen espacios de tablas temporales de usuarios con tablas temporales declaradas existentes o tablas temporales creadas en el grupo de particiones de base de datos.
- Las opciones, como por ejemplo **INDEXING MODE**, se ignoran sin aviso en las tablas en que no se aplican. Por ejemplo, **INDEXING MODE** se ignorará en las tablas sin índices.
- El estado de la ejecución del mandato se devuelve en el área de comunicaciones SQL (SQLCA) obtenida a partir de la sentencia CALL.
- El archivo al que se hace referencia en **USING DISTFILE** *nombreachivodist* o **USING TARGETMAP** *nombreachivomapedestino*, debe hacer referencia a un archivo del servidor.
- Antes de iniciar una operación de redistribución, asegúrese de que no hay tablas en estado Pendiente de carga. Se puede comprobar el estado de las tablas utilizando el mandato LOAD QUERY.
- El mandato REDISTRIBUTE DATABASE PARTITION GROUP puede resultar anómalo (SQLSTATE 55071) si una petición para añadir un servidor de particiones de base de datos está pendiente o en curso. Este comando también puede resultar anómalo (SQLSTATE 55077) si se añade en línea un servidor de particiones de base de datos nuevo a la instancia y no todas las aplicaciones saben de la existencia del servidor de particiones de base de datos nuevo.

Compatibilidades

Las tablas que contienen columnas XML que utilizan el formato de registro de DB2 Versión 9.5 o anteriores no se puede redistribuir. Utilice el procedimiento almacenado ADMIN_MOVE_TABLE para migrar la tabla al formato nuevo.

Para la compatibilidad con versiones anteriores a la Versión 8:

- La palabra clave **NODEGROUP** puede sustituirse por **DATABASE PARTITION GROUP**.

Mandato REORG INDEXES/TABLE utilizando el procedimiento ADMIN_CMD

Reorganiza un índice o una tabla

Puede reorganizar todos los índices definidos en una tabla reconstruyendo los datos de índice en páginas no fragmentadas y físicamente continuas. O una alternativa es la opción de reorganizar índices específicos en una tabla particionada de rangos.

Si especifica la opción CLEANUP ONLY de la cláusula de índice, la limpieza se realizará sin reconstruir los índices. Este mandato no puede utilizarse en índices de tablas temporales declaradas o de tablas temporales creadas (SQLSTATE 42995).

La opción de tabla reorganiza una tabla reconstruyendo las filas para eliminar los datos fragmentados y compactando la información.

Ámbito

Este mandato afecta a todas las particiones de base de datos del grupo de particiones de base de datos.

Autorización

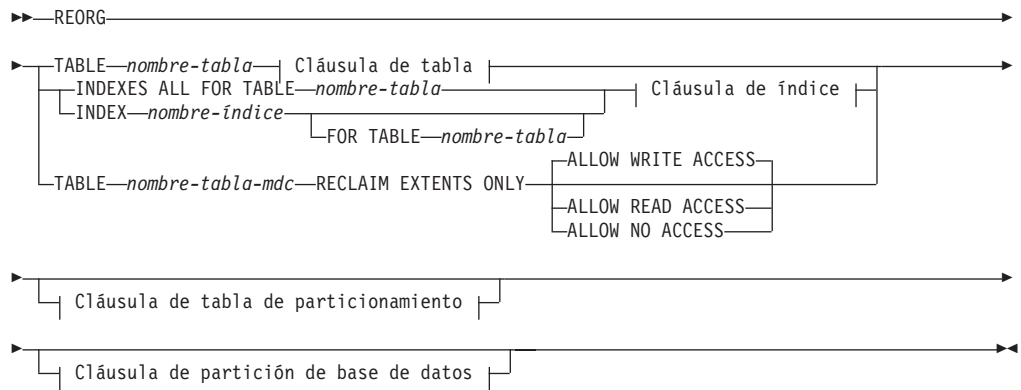
Una de las siguientes:

- *sysadm*
- *sysctrl*
- *sysmaint*
- *dbadm*
- *sqladm*
- Privilegio CONTROL en la tabla.

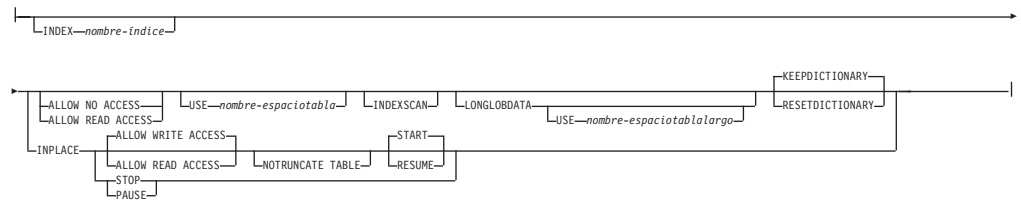
Conexión necesaria

Base de datos

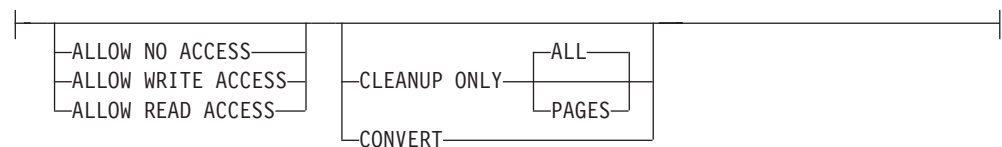
Sintaxis del mandato



Cláusula de tabla:



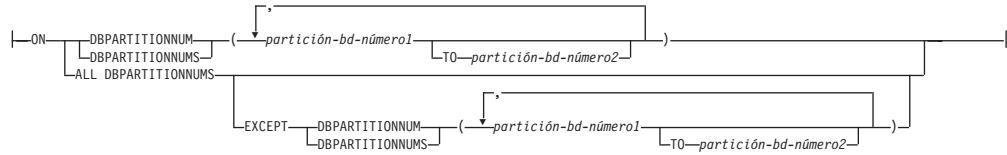
Cláusula de índice:



Cláusula de particionamiento de tabla:



Cláusula de partición de base de datos:



Parámetros del mandato

INDEXES ALL FOR TABLE *nombre-tabla*

Especifica la tabla cuyos índices se han de reorganizar. La tabla puede estar en una base de datos local o remota.

INDEX *nombre-índice*

Especifica un índice individual que se debe reorganizar en una tabla particionada. La reorganización de los índices individuales *solo* se admiten para índices no particionados en una tabla particionada. Este parámetro no está soportado para los índices de bloque.

FOR TABLE *nombre-tabla*

Especifica la ubicación de nombre de tabla del índice individual que se va a reorganizar en una tabla particionada. Este parámetro es opcional, dado que los nombres de índice son exclusivos en toda la base de datos.

Para REORG INDEXES cuando se especifica la opción ON DATA PARTITION, la cláusula de acceso solo se aplica a la partición definida. Los usuarios pueden leer y escribir en el resto de la tabla mientras se reorganizan los índices en la partición especificada. Esta situación también se aplica a los niveles de acceso por omisión.

ALLOW NO ACCESS

Para REORG INDEX y REORG INDEXES, especifica que ningún otro usuario puede acceder a la tabla o partición de datos mientras los índices se están reorganizando.

ALLOW READ ACCESS

Para REORG INDEX y REORG INDEXES, especifica que otros usuarios pueden tener acceso de solo lectura a la tabla o partición de datos mientras los índices se están reorganizando. Este nivel de acceso no se admite en REORG INDEXES de una tabla particionada a menos que se haya especificado la opción CLEANUP ONLY o la opción ON DATA PARTITION.

ALLOW WRITE ACCESS

Para REORG INDEX y REORG INDEXES, especifica que otros usuarios pueden leer y escribir en la tabla o partición de datos mientras los índices se están reorganizando.

Este nivel de acceso no se admite en tablas clustering de múltiples dimensiones (MDC) o en índices ampliados a menos que se especifique la opción CLEANUP ONLY. Este nivel de acceso tampoco no se admite en tablas particionadas cuando se utiliza el mandato REORG INDEXES a menos que se haya especificado la opción CLEANUP ONLY o la opción ON DATA PARTITION.

Cuando no se ha especificado la modalidad ACCESS, se selecciona para la tabla o la partición de datos de la forma siguiente:

Tabla 61. Acceso a la tabla por omisión basado en el mandato, tipo de tabla y parámetros adicionales especificados para la cláusula de índice:

| Mandato | Tipo de tabla | Cláusula de particionamiento de tabla | Parámetros adicionales especificados para la cláusula de índice | Modalidad de acceso por omisión |
|---------------|-----------------------|---|---|---------------------------------|
| REORG INDEXES | Tabla no particionada | No aplicable | Cualquiera | ALLOW READ ACCESS |
| REORG INDEX | Tabla particionada | No aplicable | Cualquiera | ALLOW READ ACCESS |
| REORG INDEXES | Tabla particionada | Ninguna | Ninguno especificado | ALLOW NO ACCESS |
| REORG INDEXES | Tabla particionada | ON DATA PARTITION | Ninguno especificado | ALLOW READ ACCESS |
| REORG INDEXES | Tabla particionada | Con o sin la cláusula ON DATA PARTITION | se ha especificado CLEANUP ONLY | ALLOW READ ACCESS |

CLEANUP ONLY

Cuando se solicite CLEANUP ONLY, se realizará una limpieza en vez de una reorganización completa. Los índices no se reconstruirán y las páginas liberadas estarán disponibles para que los índices definidos puedan volver a utilizarlas solo en esta tabla.

La opción CLEANUP ONLY PAGES buscará y liberará las páginas pseudovacías confirmadas. Una página pseudovacia confirmada es aquella en que todas las claves de la página están marcadas como suprimidas y se sabe que todas estas supresiones están confirmadas. El número de páginas pseudovacías de un índice puede determinarse ejecutando RUNSTATS y consultando la columna NUM EMPTY LEAFS de SYSCAT.INDEXES. La opción PAGES limpiará NUM EMPTY LEAFS si están determinadas para la confirmación.

La opción CLEANUP ONLY ALL liberará las páginas pseudovacías confirmadas, y eliminará las claves pseudosuprimidas confirmadas de las páginas que no estén pseudovacías. Esta opción intentará asimismo fusionar páginas hojas adyacentes si al hacerlo se producirá una página hoja fusionada que dispondrá de al menos un espacio libre igual a PCTFREE en dicha página hoja fusionada, donde PCTFREE es el porcentaje de espacio libre definido para el índice durante su creación. El valor por omisión de PCTFREE es el diez por ciento. Si pueden fusionarse dos páginas, se liberará una de ellas. El número de claves pseudosuprimidas de un índice, excluyendo las que están en páginas pseudovacías, puede determinarse ejecutando RUNSTATS y seleccionando NUMRIDS DELETED de SYSCAT.INDEXES. La opción ALL limpiará NUMRIDS DELETED y NUM EMPTY LEAFS si están determinadas para la confirmación.

ALL Especifica que los índices deben limpiarse eliminando las claves pseudosuprimidas confirmadas y las páginas pseudovacías confirmadas.

PAGES

Especifica que las páginas pseudovacías confirmadas deben eliminarse del árbol del índice. Esta acción no limpiará las claves pseudosuprimidas en las páginas que no estén pseudovacías. Dado que solo se comprueban las páginas hojas pseudovacías, en la mayor parte de los casos es un método considerablemente más rápido que utilizar la opción ALL.

CONVERT

Convierte índices de tipo 1 en índices de tipo 2. Si el índice ya es de tipo 2, esta opción no tiene ningún efecto.

En la versión 9.7, se han dejado de mantener los índices de tipo 1 y todos los índices que se crean son del tipo 2. Como resultado, la opción CONVERT queda en desuso.

Todos los índices creados antes de la versión 8 son índices de tipo 1. Antes de la versión 9.7, todos los índices creados por la versión 8 y posterior son índices del tipo 2, excepto si ha creado un índice en una tabla que ya tenía un índice de tipo 1. En tal caso, el nuevo índice también será del tipo 1. Esto ya no ocurre en la versión 9.7 porque todos los índices que se crean son del tipo 2.

Utilice la opción ALLOW READ ACCESS o ALLOW WRITE ACCESS para que otras transacciones de sólo lectura o de lectura-grabación tengan acceso a la tabla mientras se reorganizan los índices. Aunque ALLOW READ ACCESS y ALLOW WRITE ACCESS permiten el acceso a la tabla, durante el periodo en el que las copias reorganizadas de los índices quedan disponibles, no se permite ningún acceso a la tabla.

TABLE *nombre-tabla-mdc* RECLAIM EXTENTS ONLY

Especifica la tabla de clustering de múltiples dimensiones (MDC) que debe reorganizarse para reclamar extensiones que ya no se utilizan. Se puede utilizar el nombre o el alias con el formato: *esquema.nombre-tabla*. El *esquema* es el nombre de usuario bajo el que se ha creado la tabla. Si se omite el nombre de esquema, se supone que se trata del esquema por omisión.

Para REORG TABLE RECLAIM EXTENTS ONLY cuando se especifica la opción ON DATA PARTITION, la cláusula de acceso solo se aplica a la partición definida. Los usuarios pueden leer y escribir en el resto de la tabla mientras se reclaman las extensiones en la partición especificada. Esta situación también se aplica a los niveles de acceso por omisión.

ALLOW NO ACCESS

Para REORG TABLE RECLAIM EXTENTS ONLY, especifica que ningún otro usuario puede acceder a la tabla mientras se reclaman las extensiones.

ALLOW READ ACCESS

Para REORG TABLE RECLAIM EXTENTS ONLY, especifica que otros usuarios pueden tener acceso de solo lectura a la tabla mientras se reclaman las extensiones.

ALLOW WRITE ACCESS

Para REORG TABLE RECLAIM EXTENTS ONLY, especifica que otros usuarios pueden leer y escribir en la tabla mientras se reclaman las extensiones.

TABLE *nombre-tabla*

Especifica la tabla que se debe reorganizar. La tabla puede estar en una base de datos local o remota. Se puede utilizar el nombre o el alias con el

formato: *esquema.nombre-tabla*. El *esquema* es el nombre de usuario bajo el que se ha creado la tabla. Si se omite el nombre de esquema, se supone que se trata del esquema por omisión.

Para las tablas de tipo, el nombre de tabla especificado debe ser el nombre de la tabla raíz de la jerarquía.

No se puede especificar un índice para la reorganización de una tabla de clúster multidimensional (MDC). La reorganización de tablas in situ no se puede utilizar para las tablas MDC.

INDEX *nombre-índice*

Especifica el índice a utilizar al reorganizar la tabla. Si no especifica el nombre totalmente calificado con el formato:

esquema.nombre-índice, se toma el esquema por omisión. El *esquema* es el nombre de usuario bajo el cual se ha creado el índice. El gestor de bases de datos utiliza el índice para volver a ordenar físicamente los registros de la tabla que está reorganizando.

En el caso de una reorganización de tablas in situ, si se define un índice de clúster en la tabla y se especifica un índice, tiene que ser un índice de clúster. Si no se especifica la opción in place (in situ), se utilizará cualquier índice especificado. Si no se especifica el nombre de un índice, los registros se reorganizan sin importar el orden. No obstante, si la tabla tiene definido un índice de clúster y no se especifica ningún índice, se utiliza el índice de clúster para crear el clúster de la tabla. No se puede especificar un índice si se está reorganizando una tabla MDC.

ALLOW NO ACCESS

Especifica que ningún otro usuario puede acceder a la tabla mientras se esté reorganizando. Cuando se reorganiza una tabla particionada sin la cláusula Particionamiento de tabla, este es el valor por omisión.

ALLOW READ ACCESS

Permite solo el acceso de lectura a la tabla durante la reorganización. Es el valor por omisión para una tabla no particionada o para una tabla particionada que utilice la cláusula Particionamiento de tabla.

Cuando se especifica la opción ON DATA PARTITION para un REORG TABLE de una tabla particionada de rango, la cláusula de acceso se aplica solo a la partición definida. Si todos los índices de la tabla están particionados, los usuarios pueden leer y escribir en el resto de la tabla mientras se reorganiza la partición y se reconstruyen los índices. Sin embargo, si hay índices no particionados definidos en la tabla, el acceso a la tabla completa se reajusta en ALLOW NO ACCESS. En esta situación, los índices no particionados se reconstruyen junto con la partición especificada.

INPLACE

Reorganiza la tabla al tiempo que permite el acceso de los usuarios.

Puede realizarse una reorganización de tablas in situ solo en tablas no particionadas y en tablas que no son MDC con índices de tipo 2, pero sin índices ampliados y sin índices definidos en las columnas XML de la tabla. La reorganización de tablas sólo se puede realizar sobre tablas que tengan al menos tres páginas.

La reorganización de tablas in situ se lleva a cabo de modo asíncrono y es posible que no se haga efectiva inmediatamente.

ALLOW READ ACCESS

Permite solo el acceso de lectura a la tabla durante la reorganización.

ALLOW WRITE ACCESS

Permite el acceso de grabación a la tabla durante la reorganización. Este es el comportamiento por omisión.

NOTRUNCATE TABLE

No se trunca la tabla después de la reorganización in situ. Durante el truncamiento, la tabla tiene un bloqueo S.

START

Inicia el proceso de REORG in situ. Dado que es el valor por omisión, esta palabra clave es opcional.

STOP Detiene el proceso de REORG in situ en el punto actual.

PAUSE

Suspende o hace una pausa momentánea de REORG in situ.

RESUME

Continúa o reanuda una reorganización de tablas in situ tras una pausa previa. Cuando se reanuda la reorganización en línea, si desea tener las mismas opciones que las que había al pausar la reorganización, debe especificar de nueva esas opciones mientras reanuda.

USE *nomb-esptb*

Especifica el nombre de un espacio de tablas temporal del sistema donde se puede almacenar una copia temporal de la tabla que se está reorganizando. Si no se proporciona un nombre de espacios de tablas, el gestor de bases de datos almacena una copia de trabajo de la tabla en los espacios de tablas que contienen la tabla que se está reorganizando.

En el caso de un objeto de tabla de 8 KB, 16 KB o 32 KB, si el tamaño de página del espacio de tablas temporal del sistema que especifique no coincide con el tamaño de página de los espacios de tablas en los que residen los datos de tabla, el producto de base de datos DB2 intentará localizar un espacio de tablas temporal del tamaño correcto de los objetos LONG/LOB. Dicho espacio de tablas ya debe existir para que la reorganización sea satisfactoria.

Si tiene dos espacios de tablas temporales cuyo tamaño de página sea idéntico y especifica uno de ellos en la cláusula USE, se utilizarán por turno circular si existe un índice en la tabla que se reorganiza. Supongamos que tiene dos espacios de tablas, *tempspace1* y *tempspace2*, los dos con el mismo tamaño de página, y que especifica *tempspace1* en el mandato REORG con la opción USE. La primera vez que realice REORG, se utilizará *tempspace1*. La segunda vez, se utilizará *tempspace2*. La tercera vez, se utilizará *tempspace1*, etcétera. Para evitar este comportamiento, debe eliminar uno de los espacios de tablas temporales.

Para tablas particionadas, el espacio de tablas se utiliza como almacenamiento temporal para la reorganización de particiones de

datos en la tabla. La reorganización de la tabla particionada completa reorganiza una única partición de datos a la vez. La cantidad de espacio que se necesita es igual al tamaño de la partición de datos más grande de la tabla, no al de toda la tabla.

Si no suministra un nombre de espacio de tablas para una tabla particionada, se utilizará el espacio de tablas en el que se encuentra cada partición de datos para el almacenamiento temporal de esa partición de datos. En el espacio de tablas de cada partición de datos debe haber suficiente espacio libre para que quepa una copia de la partición de datos.

INDEXSCAN

Para un REORG de clúster, se utilizará una exploración de índice para reordenar los registros de tabla. Para reorganizar las filas de la tabla, acceda a la tabla mediante un índice. El método por omisión es explorar la tabla y clasificar el resultado para reorganizar la tabla, utilizando los espacios de tablas temporales como sea necesario. Aunque las claves de índice están dispuestas en el orden de clasificación, la exploración y la clasificación suelen ser más rápidas que captar filas leyendo primero el identificador de fila desde un índice.

LONGLOBDATA

Los datos de campo largo y LOB se han de reorganizar.

Esto no es obligatorio, aunque la tabla contenga columnas largas o LOB. El valor por omisión es evitar la reorganización de estos objetos porque consume tiempo y no mejora el clúster. No obstante, ejecutar una reorganización con la opción LONGLOBDATA en las tablas con columnas XML reclamará el espacio no utilizado y por consiguiente, reducirá el tamaño del objeto de almacenamiento XML.

Este parámetro es necesario cuando se convierten datos LOB existentes en datos LOB en línea.

USE *nomb-esptblargos*

Este es un parámetro opcional, que permite especificar el nombre de un espacio de tablas temporal que sirva para reconstruir datos largos. Si no se especifica un espacio de tablas temporal para el objeto tabla ni para los objetos largos, los objetos se construirán en el espacio de tablas en el que residen actualmente. Si se especifica un espacio de tablas temporal para la tabla, pero no se especifica este parámetro, se utilizará el espacio de tablas empleado para la reorganización de datos básica, a menos que los tamaños de página sean distintos. En esta situación, el sistema de base de datos DB2 intentará elegir un contenedor temporal que tenga el tamaño de página apropiado para crear en él los objetos largos.

Si se especifica USE *nomb-estblargos*, también debe especificarse USE *nomb-esptb*. De lo contrario, se omite el argumento *nomb-esptblargos*.

KEEPDICTIONARY

Si el atributo COMPRESS de la tabla es YES y la tabla tiene un diccionario de compresión, no se construye un nuevo diccionario. Todas las filas procesadas durante la reorganización están sujetas a la compresión mediante el diccionario existente. Si el atributo COMPRESS es YES y no existe un diccionario de compresión para la

tabla, sólo se creará uno (y la tabla comprimida) en este caso de ejemplo, si la tabla tiene un tamaño determinado (aproximadamente de 1 a 2 MB) y existen suficientes datos dentro de esta tabla. Si, en cambio, establece explícitamente REORG RESETDICTIONARY, se crea un diccionario siempre y cuando haya como mínimo 1 fila en la tabla. Si el atributo COMPRESS para la tabla es NO y la tabla tiene un diccionario de compresión, el proceso de reorganización conservará el diccionario, y todas las filas de la tabla recién organizada estarán en formato no comprimido. No se pueden comprimir algunos datos como los datos LOB que no están almacenados en la fila de tabla base.

Si no se especifica la opción LONGLOBDATA, solo se reorganizan los datos de fila de la tabla. En la tabla siguiente se describe el comportamiento de la sintaxis KEEPDICTIONARY en el mandato REORG cuando no se especifica la opción LONGLOBDATA.

Tabla 62. REORG KEEPDICTIONARY

| Comprimir | Diccionario existe | Resultado; consecuencia |
|-----------|--------------------|--|
| S | S | Se conserva el diccionario; se comprimen las filas. |
| S | N | Crear diccionario; filas comprimidas |
| N | S | Conservar diccionario; todas las filas sin comprimir |
| N | N | Sin efecto; todas las filas sin comprimir |

En la tabla siguiente se describe el comportamiento de la sintaxis KEEPDICTIONARY en el mandato REORG cuando se especifica la opción LONGLOBDATA.

Tabla 63. REORG KEEPDICTIONARY cuando se especifica la opción LONGLOBDATA.

| Comprimir | Existe el diccionario de datos de fila de tabla | Existe el diccionario de objetos de almacenamiento XML ¹ | Diccionario de compresión | Compresión de datos |
|-----------|---|---|---|--|
| S | S | S | Conservar diccionarios. | Los datos existentes se comprimen. Los datos nuevos se comprimirán. |
| S | S | N | Conservar el diccionario de filas de tabla y crear un diccionario de objetos de almacenamiento XML. | Los datos existentes se comprimen. Los datos nuevos se comprimirán. |
| S | N | S | Crear diccionario de filas de tabla y conservar el diccionario XML. | Los datos existentes se comprimen. Los datos nuevos se comprimirán. |
| S | N | N | Crear diccionarios de filas de tabla y XML. | Los datos existentes se comprimen. Los datos nuevos se comprimirán. |
| N | S | S | Conservar diccionarios de filas de tablas y XML. | Los datos de la tabla no se comprimen. Los datos nuevos no se comprimirán. |

Tabla 63. REORG KEEPDICTIONARY cuando se especifica la opción LONGLOBDATA. (continuación)

| Comprimir | Existe el diccionario de datos de fila de tabla | Existe el diccionario de objetos de almacenamiento XML ¹ | Diccionario de compresión | Compresión de datos |
|-----------|---|---|--|--|
| N | S | N | Conservar diccionario de filas de tabla. | Los datos de la tabla no se comprimen. Los datos nuevos no se comprimirán. |
| N | N | S | Conservar diccionario XML. | Los datos de la tabla no se comprimen. Los datos nuevos no se comprimirán. |
| N | N | N | Sin efecto. | Los datos de la tabla no se comprimen. Los datos nuevos no se comprimirán. |

Nota:

1. Puede crearse un diccionario de compresión para el objeto de almacenamiento XML de una tabla únicamente si se añaden columnas XML a la tabla en DB2 V9.7 o posterior, o si la tabla se migra mediante el procedimiento ONLINE_TABLE_MOVE almacenado.

Para cualquier reinicialización o truncamiento de una tabla (como por ejemplo, la operación de sustitución), si el atributo de compresión para la tabla es NO, se eliminará el diccionario, en caso de que exista. Por el contrario, si hay un diccionario y el atributo de compresión de la tabla es YES, un truncamiento ahorrará el diccionario y no lo eliminará. El diccionario se anota en su totalidad para la recuperación y para el soporte futuro con cambios de captura de datos (es decir, duplicación).

RESETDICTIONARY

Si el atributo COMPRESS de la tabla es YES, se construye un nuevo diccionario de compresión de filas. Todas las filas procesadas durante la reorganización están sujetas a la compresión mediante el nuevo diccionario. Este diccionario sustituye a cualquier otro diccionario anterior. Si el atributo COMPRESS de la tabla es NO y la tabla no tiene un diccionario de compresión existente, el proceso de reorganización eliminará el diccionario y todas las filas de la tabla recién reorganizada estarán en formato no comprimido. No se pueden comprimir algunos datos como los datos LOB que no están almacenados en la fila de tabla base.

Si no se especifica la opción LONGLOBDATA, solo se reorganizan los datos de fila de la tabla. En la tabla siguiente se describe el comportamiento de la sintaxis RESETDICTIONARY en el mandato REORG cuando no se especifica la opción LONGLOBDATA.

Tabla 64. REORG RESETDICTIONARY

| Comprimir | Diccionario existe | Resultado; consecuencia |
|-----------|--------------------|---|
| S | S | Crear diccionario nuevo*; filas comprimidas. Si la opción DATA CAPTURE CHANGES se especifica en las sentencias CREATE TABLE o ALTER TABLE, el diccionario actual se conserva (se conoce como <i>diccionario histórico de compresión</i>). |
| S | N | Crear diccionario nuevo; filas comprimidas |
| N | S | Eliminar diccionario; todas las filas sin comprimir. Si la opción DATA CAPTURE NONE se especifica en las sentencias CREATE TABLE o ALTER TABLE, también se elimina el <i>diccionario histórico de compresión</i> correspondiente a la tabla especificada. |
| N | N | Sin efecto; todas las filas sin comprimir |

* - Si hay un diccionario y se habilita el atributo de compresión, pero en la actualidad no hay datos en la tabla, la operación RESETDICTIONARY conservará el diccionario existente. Las filas que tengan un tamaño más pequeño en la longitud mínima de registro interno y las filas que no muestren que se acorta la longitud de registro cuando se intenta comprimir las se consideran "insuficientes" en este caso.

En la tabla siguiente se describe el comportamiento de la sintaxis RESETDICTIONARY en el mandato REORG cuando se especifica la opción LONGLOBDATA.

Tabla 65. REORG RESETDICTIONARY cuando se especifica la opción LONGLOBDATA.

| Comprimir | Existe el diccionario de datos de fila de tabla | Existe el diccionario de objetos de almacenamiento XML ¹ | Diccionario de datos | Compresión de datos |
|-----------|---|---|--|--|
| S | S | S | Crear diccionarios ^{2 3} . | Los datos existentes se comprimen. Los datos nuevos se comprimirán. |
| S | S | N | Crear nuevo diccionario de filas de tabla y crear nuevo diccionario XML ³ . | Los datos existentes se comprimen. Los datos nuevos se comprimirán. |
| S | N | S | Crear diccionario de datos de filas de tabla y crear nuevo diccionario XML. | Los datos existentes se comprimen. Los datos nuevos se comprimirán. |
| S | N | N | Crear diccionarios. | Los datos existentes se comprimen. Los datos nuevos se comprimirán. |
| N | S | S | Elimina diccionarios. Los datos existentes y nuevos no se comprimen. | Los datos de tabla existentes no se comprimen. Los datos nuevos no se comprimirán. |
| N | S | N | Eliminar diccionario de filas de tabla. Todos los datos están sin comprimir. | Los datos de tabla existentes no se comprimen. Los datos nuevos no se comprimirán. |

Tabla 65. REORG RESETDICTIONARY cuando se especifica la opción LONGLOBDATA. (continuación)

| Comprimir | Existe el diccionario de datos de fila de tabla | Existe el diccionario de objetos de almacenamiento XML ¹ | Diccionario de datos | Compresión de datos |
|-----------|---|---|--|--|
| N | N | S | Elimina el diccionario de objetos de almacenamiento XML. | Los datos de tabla existentes no se comprimen. Los datos nuevos no se comprimirán. |
| N | N | N | Sin efecto. | Los datos de tabla existentes no se comprimen. Los datos nuevos no se comprimirán. |

Notas:

1. Puede crearse un diccionario de compresión para el objeto de almacenamiento XML de una tabla únicamente si se añaden columnas XML a la tabla en DB2 V9.7 o posterior, o si la tabla se migra mediante un movimiento de tabla en línea.
2. Si hay un diccionario y se habilita el atributo de compresión, pero en la actualidad no hay datos en la tabla, la operación RESETDICTIONARY conservará el diccionario existente. Las filas que tengan un tamaño más pequeño en la longitud mínima de registro interno y las filas que no muestren que se acorta la longitud de registro cuando se intenta comprimir las se consideran insuficientes en este caso.
3. Si la opción DATA CAPTURE CHANGES se especifica en las sentencias CREATE TABLE o ALTER TABLE, el diccionario de datos actual se conserva (se conoce como *diccionario de compresión histórica*).

ALL DBPARTITIONNUMS

Especifica que la operación ha de realizarse en todas las particiones de base de datos especificadas en el archivo db2nodes.cfg. Éste es el valor por omisión si no se especifica una cláusula de nodo.

EXCEPT

Especifica que la operación ha de realizarse en todas las particiones de base de datos especificadas en el archivo db2nodes.cfg, salvo las especificadas en la lista de nodos.

ON DATA PARTITION *nombre-partición*

Especifica la partición de datos para la reorganización.

Esta opción sólo se utiliza con RECLAIM EXTENTS ONLY.

Si el nombre de la partición de datos no existe para la tabla especificada, REORG falla y se devuelve SQL0204N.

ON DBPARTITIONNUM | ON DBPARTITIONNUMS

Realiza la operación en un conjunto de particiones de base de datos.

partición-bd-núm-1

Especifica un número de partición de base de datos en la lista de particiones de base de datos.

partición-bd-número2

Especifica el segundo número de partición de base de

datos, de manera que todas las particiones de base de datos desde *partición-bd-núm-1* hasta *partición-bd-núm-2* inclusive se incluyen en la lista de particiones de base de datos.

Ejemplo

Reorganizar las tablas de un grupo de particiones de base de datos que consta de las particiones 1, 3 y 4.

```
CALL SYSPROC.ADMIN_CMD ('REORG TABLE employee  
INDEX empid ON DBPARTITIONNUM (1,3,4)')
```

Notas de uso

Restricciones:

- El estado de la ejecución del mandato se devuelve en el área de comunicaciones SQL (SQLCA) obtenida a partir de la sentencia CALL.
- El programa de utilidad REORG emite una sentencia COMMIT al principio de la operación que, en el caso de conexiones de Tipo 2, hace que el procedimiento devuelva SQL30090N con el código de razón 2.
- El programa de utilidad REORG no permite utilizar apodos.
- El mandato REORG TABLE no se admite en tablas temporales declaradas o en tablas temporales creadas.
- El mandato REORG TABLE no se puede usar en las vistas.
- La reorganización de una tabla no es compatible con las tablas agrupadas en clúster de rangos porque el área de rango de la tabla siempre permanece agrupada en clústeres.
- No se puede usar REORG TABLE en una tabla particionada de un espacio de tablas DMS mientras se esté haciendo una copia de seguridad en línea de CUALQUIER espacio de tablas en el que resida la tabla, incluidos los LOB y los índices.
- REORG TABLE no puede utilizar un índice basado en una extensión de índice.
- Si una tabla está en un estado pendiente de reorganización, no está permitido realizar una reorganización in situ en la tabla.
- Para tablas particionadas:
 - La tabla debe tener una ACCESS_MODE en SYSCAT.TABLES igual a Acceso Completo.
 - La reorganización se salta las particiones de datos que tienen un estado restringido debido a una operación de enlace o desenlace. Si se especifica la cláusula Particionamiento de tabla, dicha partición debe ser completamente accesible.
 - Si se produce un error durante la reorganización de la tabla, algunos índices o particiones de índice pueden invalidarse. Los índices no particionados de la tabla quedarán marcados como no válidos si la reorganización ha alcanzado o pasado la fase de sustitución para la primera partición de datos. Las particiones de índice de cualquier partición de datos que ya haya alcanzado o pasado la fase de sustitución quedarán marcadas como no válidas. Los índices se volverán a generar en los siguientes accesos a la tabla o a la partición de datos.
 - Si se produce un error durante la reorganización del índice cuando se utiliza la modalidad de acceso ALLOW NONE, algunos índices de la tabla pueden invalidarse. Para índices RID no particionados en la tabla, solo el índice que

se está reorganizando en el momento de producirse el error quedará invalidado. Para tablas MDC con índices de bloques no particionados, es posible que uno o más índices de bloques quede invalidado en caso de producirse un error. Para los índices particionados, solo quedará invalidado el objeto de índice de la partición de datos que se está reorganizando. Los índices marcados como no válidos se volverán a generar en los siguientes accesos a la tabla o a la partición de datos.

La información acerca del progreso actual de la reorganización de la tabla se graba en el archivo histórico para la actividad de la base de datos. El archivo histórico contiene un registro para cada suceso de reorganización. Para ver este archivo, ejecute el mandato LIST HISTORY para la base de datos que contiene la tabla que está reorganizando.

También puede utilizar instantáneas de tabla para supervisar el progreso de la reorganización de la tabla. Los datos de supervisión de reorganización de tabla se registran independientemente del valor del Conmutador de tabla de supervisor de base de datos.

Si se produce un error, se grabará un vuelco SQLCA en el archivo histórico. Para una reorganización de tablas in situ, el estado se registrará como PAUSED.

Cuando una tabla indexada se ha modificado muchas veces, los datos de los índices pueden quedar fragmentados. Si la tabla tiene un clúster respecto a un índice, la tabla y el índice pueden salir del orden del clúster. Ambos factores pueden afectar negativamente al rendimiento de las exploraciones que utilizan el índice y pueden influir en la eficacia de la captación previa de páginas de índice. Se pueden utilizar REORG INDEX o REORG INDEXES para reorganizar uno o todos los índices de una tabla. La reorganización de índices eliminará la fragmentación y restaurará el clúster físico en las páginas hoja. Utilice el mandato REORGCHK como ayuda para determinar si un índice se debe reorganizar. Asegúrese de completar todas las operaciones de base de datos y liberará todos los bloqueos antes de invocar la reorganización de índices. Esto puede realizarse emitiendo un mandato COMMIT después de cerrar todos los cursores abiertos WITH HOLD o emitiendo un mandato ROLLBACK.

Una reorganización de tabla clásica (reorganización fuera de línea) vuelve a construir los índices durante la última fase de la reorganización. Sin embargo, la reorganización de tabla in situ (reorganización en línea) no vuelve a construir los índices. Es recomendable emitir un mandato REORG INDEXES después de que finalice una reorganización de tabla in situ. Una reorganización de tabla in situ es asíncrona, por lo tanto se debe tener cuidado de garantizar que la reorganización de tabla in situ se haya completado antes de emitir el mandato REORG INDEXES. La emisión del mandato REORG INDEXES antes de que se complete la reorganización de tabla in situ, puede causar que falle la reorganización (SQLCODE -2219).

Las tablas que se han modificado tantas veces que los datos están fragmentados y el rendimiento de acceso es notablemente lento, son candidatas para el mandato REORG TABLE. También se debe invocar este programa de utilidad después de modificar la longitud en línea de una columna de tipo estructurado con el fin de aprovechar la longitud en línea modificada. Utilice el mandato REORGCHK para determinar si una tabla necesita reorganización. Asegúrese de completar todas las operaciones de base de datos y liberar todos los bloqueos antes de invocar REORG TABLE. Esto puede realizarse emitiendo un COMMIT después de cerrar todos los cursores abiertos WITH HOLD o emitiendo un ROLLBACK. Después de

reorganizar una tabla, utilice RUNSTATS para actualizar las estadísticas de tabla, y REBIND para volver a vincular los paquetes que utilizan dicha tabla. El programa de utilidad de reorganización cerrará implícitamente todos los cursores.

Si la tabla contiene un formato de filas mixto porque se ha activado o desactivado la compresión de valores de tabla, una reorganización de tabla fuera de línea puede convertir todas las filas existentes al formato de fila de destino.

Si la tabla está distribuida en varias particiones de base de datos y la reorganización de la tabla o índice falla en cualquiera de las particiones de base de datos afectadas, solo se retrotraerá la reorganización de la tabla o índices en las particiones de base de datos anómalas.

Si la reorganización no es satisfactoria, no se deberán suprimir los archivos temporales. El gestor de bases de datos utiliza dichos archivos para recuperar la base de datos.

Si se especifica el nombre de un índice, el gestor de bases de datos reorganiza los datos de acuerdo con el orden del índice. Para maximizar el rendimiento, especifique un índice que se utilice con frecuencia en consultas de SQL. Si *no* se especifica el nombre de un índice y existe un índice de clúster, los datos se ordenarán de acuerdo al índice de clúster.

El valor PCTFREE de una tabla determina la cantidad de espacio libre designado por página. Si no se ha establecido el valor, el programa de utilidad llenará tanto espacio como sea posible en cada página.

Para realizar una recuperación en avance del espacio de tablas después de una reorganización de tabla, tanto los espacios de tablas normales como los de gran tamaño se deberán habilitar para la recuperación en avance.

Si la tabla contiene columnas LOB que no utilizan la opción COMPACT, el objeto de almacenamiento LOB DATA puede ser significativamente mayor después de la reorganización de la tabla. Esto puede ser el resultado del orden en el que se han reorganizado las filas y se han utilizado los tipos de espacios de tablas (SMS o DMS).

Se pueden volver a crear índices sobre datos XML mediante el mandato REORG INDEXES/TABLE. Para obtener información detallada, consulte “Volver a crear índices sobre datos XML”.

Mandato RESET ALERT CONFIGURATION utilizando el procedimiento ADMIN_CMD

Restablece los valores del indicador de salud para objetos específicos a los valores por omisión actuales para ese tipo de objeto o restablece los valores por omisión actuales del indicador de salud para un tipo de objeto a los valores por omisión de instalación.

Importante: Este mandato o API ha quedado en desuso y puede que se elimine en un futuro release ya que el Supervisor de salud ha quedado en desuso en la Versión 9.7. Para obtener más información, consulte el tema “El Supervisor de salud ha quedado en desuso” de la publicación *Novedades en DB2 Versión 9.7*.

Autorización

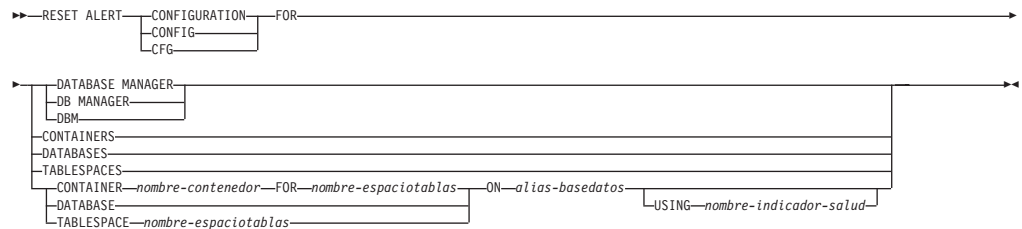
Una de las siguientes:

- *sysadm*
- *sysmaint*
- *sysctrl*

Conexión necesaria

Base de datos

Sintaxis del mandato



Parámetros del mandato

DATABASE MANAGER | DB MANAGER | DBM

Restablece los valores de alerta para el gestor de bases de datos.

CONTAINERS

Restablece los valores de alerta para todos los contenedores de espacios de tablas gestionados por el gestor de bases de datos a los valores por omisión de instalación. Estos son los valores que se aplican a todos los contenedores de espacios de tablas que no tienen valores personalizados. Los valores personalizados se definen utilizando la cláusula `CONTAINER nombre-contenedor FOR nombre-espaciotablas ON alias-basedatos`.

DATABASES

Restablece valores de alerta para todas las bases de datos gestionadas por el gestor de bases de datos. Estos son los valores que se aplican a todas las bases de datos que no tienen valores personalizados. Los valores personalizados se definen utilizando la cláusula `DATABASE ON alias-basedatos`.

TABLESPACES

Restablece los valores de alerta por omisión para todos los espacios de tablas gestionados por el gestor de bases de datos a los valores por omisión de instalación. Estos son los valores que se aplican a todos los espacios de tablas que no tienen valores personalizados. Los valores personalizados se definen utilizando la cláusula `TABLESPACE nombre-espaciotablas ON alias-basedatos`.

CONTAINER nombre-contenedor FOR nombre-espaciotablas ON alias-basedatos

Restablece los valores de alerta para el contenedor de espacio de tablas denominado *nombre-contenedor*, para el espacio de tablas especificado utilizando la cláusula `FOR nombre-espaciotablas`, en la base de datos especificada utilizando la cláusula `ON alias-basedatos`. Si este contenedor de espacio de tablas tiene valores personalizados, estos se eliminan y se utilizan los valores por omisión de los contenedores de espacios de tablas actuales.

DATABASE ON *alias-basedatos*

Restablece los valores de alerta para la base de datos especificada utilizando la cláusula **ON** *alias-basedatos*. Si esta base de datos tiene valores personalizados, se eliminan estos valores y se utilizan los valores por omisión de instalación.

TABLESPACE *nombre-espaciotablas* **ON** *alias-basedatos*

Restablece los valores de alerta para el espacio de tablas denominado *nombre-espaciotablas*, en la base de datos especificada utilizando la cláusula **ON** *alias-basedatos*. Si este espacio de tablas tiene valores personalizados, se eliminan estos valores y se utilizan los valores por omisión de instalación.

USING *nombre-indicador-salud*

Especifica el conjunto de indicadores de salud para los que se restablecerá la configuración de alerta. Los nombres de indicador de salud se componen de un identificador de objeto de dos letras, seguido de un nombre que describe lo que mide el indicador. Por ejemplo:

```
db.sort_privmem_util
```

Si esta opción no se especifica, todos los indicadores de salud del objeto o el tipo de objeto especificado se restablecerán.

Ejemplo

Restablece los valores de alerta correspondientes al gestor de bases de datos propietario de la base de datos que contiene el procedimiento **ADMIN_CMD**.

```
CALL SYSPROC.ADMIN_CMD( 'reset alert cfg for dbm' )
```

Notas de uso

El estado de la ejecución del mandato se devuelve en el área de comunicaciones SQL (SQLCA) obtenida a partir de la sentencia **CALL**.

El *alias-basedatos* debe representar una base de datos local definida en el catálogo existente en el servidor, porque el procedimiento **ADMIN_CMD** solo se ejecuta en el servidor.

Mandato RESET DATABASE CONFIGURATION utilizando el procedimiento ADMIN_CMD

Restablece la configuración de una base de datos específica a los valores por omisión del sistema.

Ámbito

Este mandato solo afecta a la partición de base de datos con la que se conecta la aplicación.

Autorización

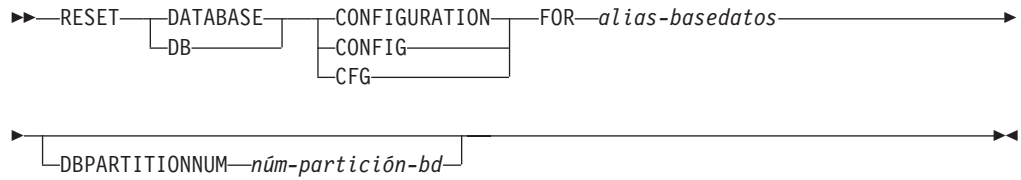
Una de las siguientes:

- SYSADM
- SYSTRM
- SYSMINT

Conexión necesaria

Base de datos

Sintaxis del mandato



Parámetros del mandato

FOR *alias-basedatos*

Especifica el alias de la base de datos cuya configuración debe restablecerse a los valores por omisión del sistema. El alias de base de datos debe estar definido en el catálogo existente en el servidor y debe hacer referencia a una base de datos local del servidor.

DBPARTITIONNUM *núm-partición-bd*

Si se tiene que aplicar un restablecimiento de configuración de base de datos a una partición de base de datos específica, se puede utilizar este parámetro. Si no se proporciona este parámetro, el restablecimiento tendrá efecto sobre todas las particiones de la base de datos.

Ejemplo

Restablecer la configuración de una base de datos catalogada con el alias SAMPLE en el servidor

```
CALL SYSPROC.ADMIN_CMD( 'reset db cfg for SAMPLE' )
```

Notas de uso

Para ver o imprimir una lista de los parámetros de configuración de la base de datos, utilice la vista de administración de SYSIBMADM.DBCFG.

Para cambiar el valor de un parámetro configurable, utilice el mandato UPDATE DATABASE CONFIGURATION.

Los cambios efectuados en el archivo de configuración de base de datos solo entran en vigor después de cargarse en la memoria. Para que esto pueda producirse, todas las aplicaciones deben desconectarse de la base de datos.

Si se produce un error, el archivo de configuración de base de datos no se modifica.

El archivo de configuración de base de datos no se puede restablecer si la suma de comprobación no es válida. Esto podría producirse si se modifica el archivo de configuración de base de datos sin utilizar el mandato apropiado. Si sucede esto, se deberá restaurar la base de datos para restablecer el archivo de configuración de la misma.

El mandato RESET DATABASE CONFIGURATION restablecerá los parámetros de configuración de base de datos a los valores de configuración por omisión

documentados, donde el valor de **auto_runstats** será ON. **Self_tuning_mem** restablecerá a ON en los entornos de bases de datos no particionadas y restablecerá a OFF en los entornos de bases de datos particionadas.

El estado de la ejecución del mandato se devuelve en el área de comunicaciones SQL (SQLCA) obtenida a partir de la sentencia CALL.

El *alias-basedatos* debe representar una base de datos local definida en el catálogo existente en el servidor, porque el procedimiento ADMIN_CMD solo se ejecuta en el servidor.

Mandato RESET DATABASE MANAGER CONFIGURATION utilizando el procedimiento ADMIN_CMD

Restablece los parámetros del archivo de configuración del gestor de bases de datos en los valores por omisión del sistema para la instancia que contiene la base de datos conectada actualmente. Los valores se restablecen por el tipo de nodo.

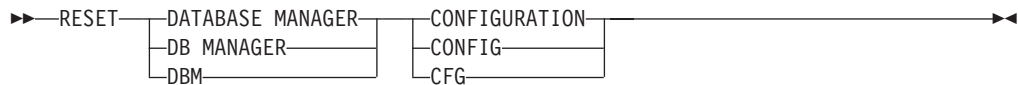
Autorización

SYSADM

Conexión necesaria

Base de datos

Sintaxis del mandato



Parámetros del mandato

Ninguna

Ejemplo

Restablecer la configuración de la instancia que contiene la base de datos a la que pertenece el procedimiento almacenado ADMIN_CMD.

```
CALL SYSPROC.ADMIN_CMD( 'reset dbm cfg' )
```

Notas de uso

Este mandato restablece todos los parámetros establecidos por el programa de instalación. Esto podría hacer que se devolvieran mensajes de error al reiniciar DB2. Por ejemplo, si se restablece el parámetro **svcname**, el usuario recibirá el mensaje de error SQL5043N cuando intente reiniciar DB2.

Antes de ejecutar este mandato, guarde la salida de la vista administrativa SYSIBMADM.DBMCFG en un archivo para poder consultar los valores existentes. Los valores individuales se pueden actualizar a continuación utilizando el mandato UPDATE DATABASE MANAGER CONFIGURATION mediante el procedimiento ADMIN_CMD.

No se recomienda que el usuario modifique el parámetro **svcename**, establecido por el programa de instalación.

Para ver o imprimir una lista de los parámetros de configuración del gestor de bases de datos, utilice la vista de administración SYSIBMADM.DBMCFG. Para cambiar el valor de un parámetro configurable, utilice el mandato UPDATE DATABASE MANAGER CONFIGURATION mediante el procedimiento ADMIN_CMD.

Para obtener más información acerca de estos parámetros, consulte la lista de resumen de parámetros de configuración y los parámetros individuales.

Algunos cambios efectuados en el archivo de configuración del gestor de bases de datos solo entran en vigor después de haberse cargado en la memoria. Para obtener más información sobre qué parámetros se pueden configurar en línea y cuáles no, vea el resumen de parámetros de configuración. Los parámetros de configuración de servidor que no se restablecen inmediatamente, se restablecen durante la ejecución de db2start. Para un parámetro de configuración de cliente, los parámetros se restablecen la vez siguiente que se reinicia la aplicación. Si el cliente es el procesador de línea de mandatos, es necesario invocar TERMINATE.

Si se produce un error, el archivo de configuración del gestor de bases de datos no se modifica.

El archivo de configuración del gestor de bases de datos no puede restablecerse si la suma de comprobación no es válida. Esto podría producirse si editara manualmente el archivo de configuración y no utiliza el mandato adecuado. Si la suma de comprobación no es válida, debe volver a crear la instancia.

Mandato REWIND TAPE utilizando el procedimiento ADMIN_CMD

Rebobina cintas para las operaciones de copia de seguridad y restauración en los dispositivos de cinta de modalidad continua. Este mandato sólo está soportado en los sistemas operativos Windows.

Autorización

Una de las siguientes:

- *sysadm*
- *sysctrl*
- *sysmaint*

Conexión necesaria

Base de datos

Sintaxis del mandato

```
➤➤ REWIND TAPE [ON dispositivo] ➤➤
```

Parámetros del mandato

ON dispositivo

Especifica un nombre de dispositivo de cinta válido. El valor por omisión es `\\.\TAPE0`. El dispositivo especificado debe ser relativo al servidor.

Ejemplo

Rebobinar la cinta en el dispositivo llamado `'\\.\TAPE1'`.

```
CALL SYSPROC.ADMIN_CMD( 'rewind tape on \\.\TAPE1' )
```

Notas de uso

El estado de la ejecución del mandato se devuelve en el área de comunicaciones SQL (SQLCA) obtenida a partir de la sentencia CALL.

Mandato RUNSTATS utilizando el procedimiento ADMIN_CMD

Actualiza las estadísticas sobre las características de una tabla y/o los índices asociados o de las vistas estadísticas. Estas características incluyen el número de registros, el número de páginas y el promedio de longitud de registro. El optimizador utiliza estas estadísticas al determinar las vías de acceso a los datos.

En el caso de una tabla, habrá que llamar a este programa de utilidad cuando se hayan hecho muchas actualizaciones de la tabla o después de reorganizar la tabla. En el caso de una vista estadística, habrá que llamar a este programa de utilidad cuando los cambios realizados en tablas subyacentes hayan afectado notablemente a las filas devueltas por la vista. La vista debe haberse habilitado con anterioridad para utilizarla en la optimización de consultas con el mandato ALTER VIEW.

Ámbito

Este mandato puede emitirse desde cualquier partición de base de datos del archivo `db2nodes.cfg`. Puede utilizarse para actualizar los catálogos en la partición de base de datos de catálogo.

En el caso de las tablas, el mandato reúne estadísticas para una tabla en la partición de base de datos desde la que se invoca. Si la tabla no existe en esa partición de base de datos, se selecciona la primera partición de base de datos del grupo de particiones de base de datos.

En el caso de las vistas, el mandato reúne estadísticas utilizando los datos de las tablas en todas las particiones de base de datos participantes.

Autorización

En el caso de las tablas, una de las siguientes:

- *sysadm*
- *sysctrl*
- *sysmaint*
- *dbadm*
- *sqladm*
- Privilegio CONTROL sobre la tabla
- Autorización LOAD

No se necesita ningún privilegio explícito para utilizar este mandato en ninguna tabla temporal global declarada que exista en su conexión.

En el caso de las vistas estadísticas, una de las siguientes:

- *sysadm*
- *sysctrl*
- *sysmaint*
- *dbadm*
- *sqladm*
- Privilegio CONTROL sobre la vista estadística

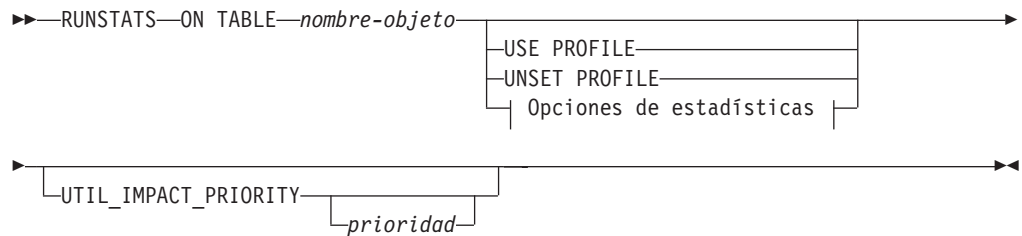
Además, hay que poseer los privilegios pertinentes para acceder a las filas desde la vista estadística. Concretamente, para cada tabla, vista estadística o apodo al que se haga referencia en la definición de la vista estadística, el usuario debe tener uno de los siguientes privilegios:

- *dataaccess*
- CONTROL
- SELECT

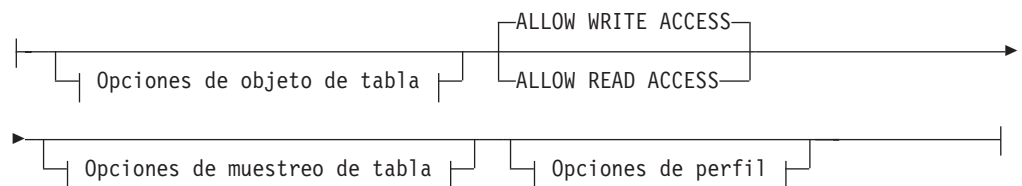
Conexión necesaria

Base de datos

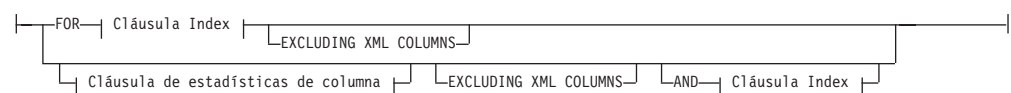
Sintaxis del mandato



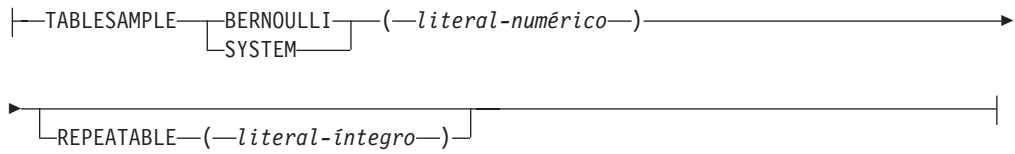
Opciones de estadísticas:



Opciones de objeto de tabla:



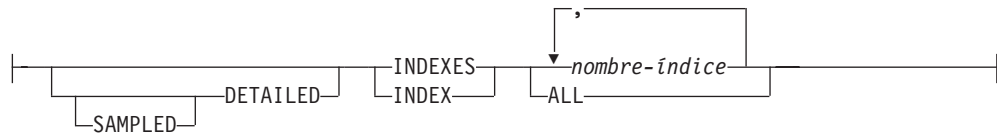
Opciones de muestreo de tabla:



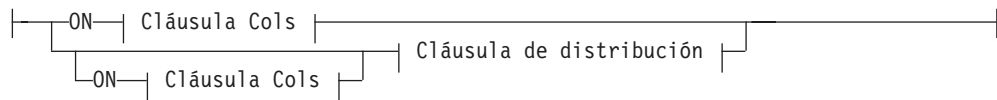
Opciones de perfil:



Cláusula de índice:



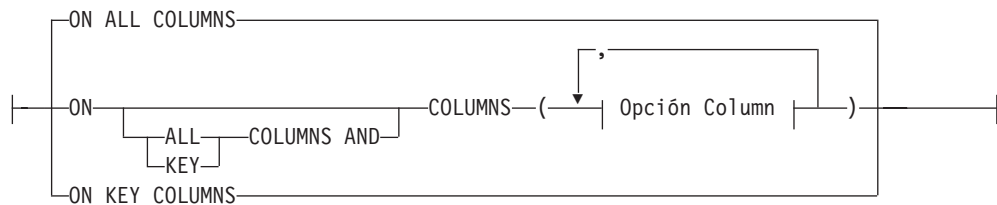
Cláusula de estadísticas de columna:



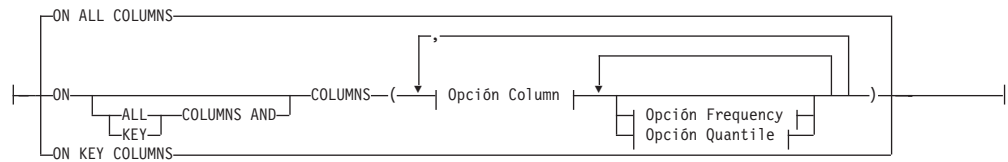
Cláusula de distribución:



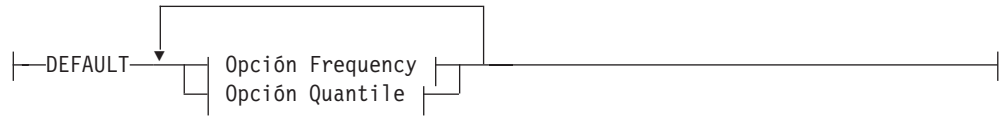
Cláusula En cols:



Cláusula En CoIs Dist:



Opción de Dist por omisión:



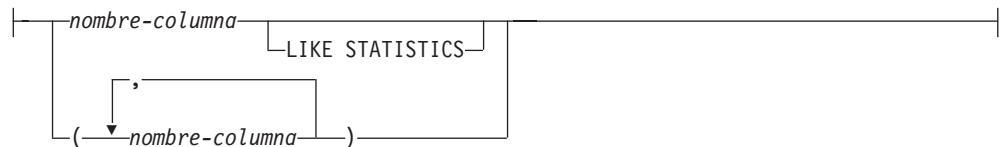
Opción de frecuencia:



Opción de cuantil:



Opción de columna:



Parámetros del mandato

nombre-objeto

Identifica la tabla o la vista estadística en la que se deben reunir estadísticas. No debe ser una tabla de la jerarquía. Para las tablas de tipo, *nombre-objeto* debe ser el nombre de la tabla raíz de la jerarquía de la tabla. Debe utilizarse el nombre totalmente calificado o el alias con el formato: *esquema.nombre-objeto*. El esquema es el nombre de usuario con el que se ha creado la tabla.

nombre-índice

Identifica un índice existente definido en la tabla. Se debe utilizar el nombre totalmente calificado con el formato *esquema.nombre-índice*. Esta opción no se puede usar para las vistas.

USE PROFILE

Esta opción permite que RUNSTATS emplee un perfil de estadísticas almacenado con anterioridad para reunir estadísticas sobre una tabla o una vista estadística. El perfil de estadísticas se crea utilizando las opciones de SET PROFILE y se actualiza utilizando las opciones de UPDATE PROFILE.

UNSET PROFILE

Especifique esta opción para eliminar un perfil de estadísticas existente. Por ejemplo,

```
runstats on tablemyschema.mytable unset profile
```

FOR INDEXES

Reúne y actualiza las estadísticas únicamente para los índices. Si en la tabla no se han recopilado estadísticas de tabla con anterioridad, también se reúnen las estadísticas básicas de tabla. Estas estadísticas básicas no incluyen ninguna estadística de distribución. Esta opción no se puede usar para las vistas.

AND INDEXES

Reúne y actualiza las estadísticas para la tabla y para los índices. Esta opción no se puede usar para las vistas.

DETAILED

Calcula las estadísticas de índice ampliadas. Son las estadísticas CLUSTERFACTOR y PAGE_FETCH_PAIRS, que se reúnen para índices relativamente grandes. Esta opción no se puede usar para las vistas.

SAMPLED

Esta opción, cuando se utiliza con la opción DETAILED, permite que RUNSTATS emplee una técnica de muestreo de CPU al compilar las estadísticas de índice ampliadas. Si no se especifica la opción, se examinará cada entrada del índice para calcular las estadísticas de índice ampliadas. Esta opción no se puede usar para las vistas.

ON ALL COLUMNS

Para recopilar estadísticas de todas las columnas candidatas, utilice la cláusula ON ALL COLUMNS. Las columnas se pueden especificar para la recopilación básica de estadísticas (cláusula En Cols) o junto con la cláusula WITH DISTRIBUTION (cláusula En Cols Dist). La especificación ON ALL COLUMNS es la opción por omisión si no se especifica ninguna de las cláusulas específicas de columnas.

Si se especifica en la cláusula En Cols, todas las columnas sólo tendrán recopiladas estadísticas básicas de columna, a menos que se seleccionen columnas específicas como parte de la cláusula WITH DISTRIBUTION. De estas columnas especificadas como parte de la cláusula WITH DISTRIBUTION se reunirán también estadísticas básicas y de distribución.

Si se especifica WITH DISTRIBUTION ON ALL COLUMNS, se reúnen estadísticas básicas y de distribución para todas las columnas elegibles. Cualquier dato que se especifique en la cláusula En Cols es redundante y por lo tanto, no es necesario.

ON COLUMNS

Esta cláusula permite especificar al usuario una lista de columnas para la que se reunirán estadísticas. Si especifica un grupo de columnas, se reunirá el número de valores diferenciados para el grupo. Cuando se ejecuta RUNSTATS en una tabla sin reunir estadísticas de índice, y se especifica un subconjunto de columnas de las que hay que reunir estadísticas, sucede lo siguiente:

1. Las estadísticas de columnas no especificadas en el mandato RUNSTATS , pero que son la primera columna de un índice, NO se restablecen.
2. Las estadísticas de las demás columnas no especificadas en el mandato RUNSTATS se restablecen.

Esta cláusula se puede utilizar en la cláusula En Colas y en la cláusula En Cols Dist. Actualmente la recopilación de estadísticas de distribución para un grupo de columnas no está soportada.

Si se especifican columnas de tipo XML en un grupo de columnas, las columnas de tipo XML se ignorarán de cara a la recopilación de valores diferenciados para el grupo. Sin embargo, se recopilarán estadísticas de columnas XML para las columnas de tipo XML del grupo de columnas.

EXCLUDING XML COLUMNS

Esta cláusula le permite omitir todas las columnas de tipo XML de la recopilación de estadísticas. Esta cláusula facilita la recopilación de estadísticas en las columnas no XML, porque la inclusión de datos XML puede exigir más recursos del sistema. La cláusula EXCLUDING XML COLUMNS tiene prioridad sobre las otras cláusulas que especifiquen columnas XML para la recopilación de estadísticas. Por ejemplo, si se utiliza la cláusula EXCLUDING XML COLUMNS y también se especifican columnas de tipo XML con la cláusula ON COLUMNS, o se utiliza la cláusula ON ALL COLUMNS, se ignorarán todas las columnas de tipo XML durante la recopilación de estadísticas.

ON KEY COLUMNS

En lugar de listar columnas específicas, se puede elegir la opción de reunir estadísticas en columnas que componen todos los índices definidos en la tabla. Se supone que las columnas críticas de las consultas también son las que se utilizan para crear índices en la tabla. Si no hay ningún índice en la tabla, es como si fuese una lista vacía y no se reunirán estadísticas de columna. Se puede usar en la cláusula-en-cols o en la cláusula-en-cols-dist. Es redundante en la cláusula-en-cols si se especifica en ambas cláusulas, ya que la cláusula WITH DISTRIBUTION sirve para especificar la recopilación de ambas estadísticas, las básicas y las de distribución. Por definición, las columnas de tipo XML no son columnas de clave, y la cláusula ON KEY COLUMNS no las incluirá en la recopilación de estadísticas. Esta opción no se puede usar para las vistas.

nombre-columna

Nombre de una columna de la tabla o de la vista estadística. Si se especifica el nombre de una columna inelegible para la recopilación de estadísticas, tal como una columna inexistente o un nombre de columna mal escrito, se devolverá el error (-205). Se pueden especificar dos listas de columnas, una sin distribución y una con distribución. Si se especifica la columna en la lista que no está asociada a la cláusula WITH DISTRIBUTION, solo se reunirán las estadísticas básicas de columna. Si la columna aparece en ambas listas, se reunirán estadísticas de distribución (a menos que se establezca que NUM_FREQVALUES y NUM_QUANTILES sean igual a cero).

NUM_FREQVALUES

Define el número máximo de valores de frecuencia que se van a reunir. Se puede especificar para una columna individual en la cláusula ON COLUMNS. Si el valor no se especifica para una columna individual, el valor de límite de frecuencia se tomará del valor especificado en la cláusula DEFAULT. Si allí tampoco se especifica, el número máximo de valores de frecuencia que se han de reunir será el establecido en el parámetro NUM_FREQVALUES de configuración de base de datos.

NUM_QUANTILES

Define el número máximo de valores de cuantil de distribución que se van a reunir. Se puede especificar para una columna individual en la cláusula ON COLUMNS. Si el valor no se especifica para una columna individual, el valor de límite de cuantil se tomará del valor especificado en la cláusula DEFAULT. Si allí tampoco se especifica, el número máximo de valores de

cuantil que se han de reunir será el establecido en el parámetro NUM_QUANTILES de configuración de base de datos.

WITH DISTRIBUTION

Esta cláusula especifica que deben reunirse estadísticas tanto básicas como de distribución en las columnas. Si no se especifica la cláusula ON COLUMNS, se reúnen estadísticas de distribución en todas las columnas de la tabla o de la vista estadística (excluidas las columnas inelegibles, como las de CLOB y LONG VARCHAR). Si se especifica la cláusula ON COLUMNS, solo se reúnen estadísticas de distribución en la lista de columnas proporcionada (excluidas las inelegibles para la recopilación de estadísticas). Si no se especifica la cláusula, solo se reunirán estadísticas básicas.

Actualmente, la recopilación de estadísticas de distribución en grupos de columnas no está soportada; las estadísticas de distribución no se reunirán cuando se especifiquen grupos de columnas en la cláusula WITH DISTRIBUTION ON COLUMNS.

DEFAULT

Si se especifica NUM_FREQVALUES o NUM_QUANTILES, estos valores se utilizarán para determinar el número máximo de estadísticas de frecuencia y de cuantiles que se reunirán para las columnas, si no se especifican para columnas individuales en la cláusula ON COLUMNS. Si no se ha especificado la cláusula DEFAULT, los valores utilizados serán los de los parámetros de configuración de base de datos correspondientes.

LIKE STATISTICS

Cuando se especifica esta opción, se reúnen estadísticas de columna adicionales. Son las estadísticas SUB_COUNT y SUB_DELIM_LENGTH de SYSSTAT.COLUMNS. Las estadísticas se recopilan por columnas de tipo CHAR y VARCHAR con un atributo de página de códigos del juego de caracteres de un solo byte (SBCS), FOR BIT DATA o UTF-8. Los utiliza el optimizador de consultas para mejorar las estimaciones de selectividad en los predicados de tipo "column LIKE '%xyz'" y "column LIKE '%xyz%'".

ALLOW WRITE ACCESS

Especifica que otros usuarios pueden leer y grabar en la(s) tabla(s) mientras se calculan las estadísticas. En el caso de las vistas estadísticas, estas son las tablas base a las que se hace referencia en la definición de las vistas.

No se recomienda utilizar la opción ALLOW WRITE ACCESS para tablas en las que se vayan a producir inserciones, actualizaciones o supresiones simultáneamente. Primero el mandato RUNSTATS calcula estadísticas de la tabla y a continuación, calcula estadísticas de índice. Los cambios en el estado de la tabla entre el momento en el que se recopilan estadísticas de la tabla y el índice pueden producir incoherencias. Si bien es importante tener estadísticas actualizadas para la optimización de consultas, también es importante tener estadísticas coherentes. Por lo tanto, las estadísticas deben recopilarse en el momento en que las inserciones, actualizaciones o supresiones estén bajo mínimos.

ALLOW READ ACCESS

Especifica que otros usuarios pueden tener acceso solo de lectura a la(s) tabla(s) mientras se calculan las estadísticas. En el caso de las vistas estadísticas, estas son las tablas base a las que se hace referencia en la definición de las vistas.

TABLESAMPLE BERNOULLI

Esta opción permite que RUNSTATS recoja estadísticas en una muestra de las filas de la tabla o de la vista estadística. El muestreo de BERNOULLI considera cada fila individualmente, incluida la fila con la probabilidad $P/100$ (donde P es el valor de literal numérico) y excluyéndola con la probabilidad $1-P/100$. Por lo tanto, si literal-numérico se evalúa de modo que tenga el valor 10, lo que representa una muestra del diez por ciento, cada fila se incluirá con la probabilidad 0,1 y se excluirá con la probabilidad 0,9. A menos que se especifique la cláusula opcional REPEATABLE, cada ejecución de RUNSTATS producirá habitualmente una muestra diferente de la tabla. Todas las páginas de datos se recuperarán mediante una exploración de la tabla pero solo se utilizará para la recopilación de estadísticas el porcentaje de filas especificado mediante el parámetro literal-numérico.

TABLESAMPLE SYSTEM

Esta opción permite que RUNSTATS recoja estadísticas en una muestra de las páginas de datos de la(s) tabla(s). El muestreo de SYSTEM considera cada página individualmente, incluyendo cada página con la probabilidad $P/100$ (donde P es el valor de literal numérico) y excluyéndola con la probabilidad $1-P/100$. A menos que se especifique la cláusula opcional REPEATABLE, cada ejecución de RUNSTATS producirá habitualmente una muestra diferente de la tabla. El tamaño del ejemplo se controla mediante el parámetro literal-numérico entre paréntesis, que representa un porcentaje aproximado P de la tabla que se va a devolver. Solo se recuperará y se utilizará para la recopilación de estadísticas un porcentaje de las páginas de datos especificado mediante el parámetro literal-numérico.

En vistas estadísticas, el muestreo de SYSTEM está restringido a las vistas cuyas definiciones son una selección en una sola tabla base. Si la vista contiene varias tablas, el muestreo de SYSTEM también es posible si:

- Las tablas se unen utilizando predicados de igualdad en todas las columnas de clave primaria y clave foránea incluidas en una restricción de integridad referencial definida entre las tablas.
- Ninguna condición de búsqueda filtra las filas en las tablas padre de la relación; y
- Puede identificarse entre todas las tablas una única tabla hija que a su vez no sea una tabla padre.

Si la vista de estadísticas no cumple esas condiciones, en su lugar se utilizará el muestreo BERNOULLI y se enviará un mensaje de aviso (SQL2317W).

REPEATABLE (*literal-entero*)

Al añadir la cláusula REPEATABLE a la cláusula TABLESAMPLE se garantiza que las ejecuciones repetidas de RUNSTATS devuelvan la misma muestra. El parámetro *literal-entero* es un entero no negativo que representa el generador que se debe utilizar para el muestreo. Si se pasa un generador negativo se producirá un error (SQL1197N). La muestra establecida aún podría variar entre invocaciones repetibles de RUNSTATS si la actividad con respecto a la tabla o a la vista estadística ha provocado cambios en los datos de la tabla o de la vista estadística desde la última vez que se ejecutó TABLESAMPLE REPEATABLE. Además, el método con el que se obtuvo el muestreo, especificado por la palabra clave BERNOULLI o SYSTEM, también debe ser el mismo para garantizar unos resultados coherentes.

literal-numérico

El parámetro literal-numérico especifica el tamaño de la muestra que se

debe obtener, como un porcentaje P. Este valor debe ser un número positivo inferior o igual a 100 y puede estar entre 1 y 0. Por ejemplo, el valor 0,01 representa una centésima de un porcentaje, lo que significa que se tomará un muestreo de 1 fila entre 10.000 como promedio. DB2 tratará el valor 0 ó 100 como si no se especificara el muestreo, independientemente de si se ha especificado TABLESAMPLE BERNOULLI o TABLESAMPLE SYSTEM. DB2 tratará un valor superior a 100 o inferior a 0 como un error (SQL1197N).

SET PROFILE NONE

Especifica que no se establecerá ningún perfil de estadística para esta invocación de RUNSTATS.

SET PROFILE

Permite a RUNSTATS generar y almacenar un perfil de estadísticas específico en las tablas de catálogos del sistema y ejecuta las opciones del mandato RUNSTATS para recopilar estadísticas.

SET PROFILE ONLY

Permite a RUNSTATS generar y almacenar un perfil de estadísticas específico en las tablas de catálogos del sistema sin ejecutar las opciones de mandato RUNSTATS.

UPDATE PROFILE

Permite a RUNSTATS modificar un perfil de estadísticas existente en las tablas de catálogos del sistema, y ejecuta las opciones del mandato RUNSTATS del perfil de estadísticas actualizado para reunir estadísticas.

UPDATE PROFILE ONLY

Permite a RUNSTATS modificar un perfil de estadísticas existente en las tablas de catálogos del sistema sin ejecutar las opciones del mandato RUNSTATS del perfil de estadísticas actualizado.

UTIL_IMPACT_PRIORITY *prioridad*

Especifica que RUNSTATS se regulará al nivel especificado por *prioridad*. *prioridad* es un número entre 1 y 100; 100 representa la prioridad más alta y 1 representa la más baja. La prioridad especifica la cantidad de regulación a que está sujeto el programa de utilidad. Todos los programas de utilidad con la misma prioridad se someten a la misma cantidad de regulación y los programas de utilidad con prioridades inferiores se regulan más que los que tienen prioridades superiores. Si *prioridad* no se especifica, RUNSTATS tendrá la prioridad por omisión de 50. Si se omite la palabra clave UTIL_IMPACT_PRIORITY, el programa de utilidad RUNSTATS se invocará sin que sea posible la regulación. Si se especifica la palabra clave UTIL_IMPACT_PRIORITY, pero se establece que el parámetro de configuración **util_impact_lim** sea igual a 100, el programa de utilidad se ejecutará sin regular. Esta opción no se puede usar para las vistas.

En una base de datos particionada, el mandato RUNSTATS, cuando se usa en tablas, reúne las estadísticas en una sola partición de base de datos. Si la partición de base de datos desde la que se ejecuta el mandato RUNSTATS tiene una partición de la tabla, el mandato se ejecutará en esa partición de base de datos. De lo contrario, el mandato se ejecutará en la primera partición de base de datos del grupo de las particiones de base de datos entre las que está particionada la tabla.

Ejemplo

Reunir estadísticas sobre todas las columnas utilizadas en índices y sobre todos los índices.


```
CALL SYSPROC.ADMIN_CMD ('RUNSTATS ON TABLE db2user.employee
ON KEY COLUMNS and INDEXES ALL')
```

Notas de uso

1. Cuando hay particiones desenlazadas en una tabla particionada, las claves de índice que aún pertenecen a las particiones de datos desenlazadas que exigen una limpieza no se contarán como parte de las claves en las estadísticas. Estas claves no se cuentan porque son invisibles y han dejado de formar parte de la tabla. Al final se eliminarán del índice mediante la limpieza asíncrona del índice. Como resultado, las estadísticas recopiladas antes de ejecutar la limpieza asíncrona del índice serán engañosas. Si se emite el mandato RUNSTATS antes de que finalice la limpieza asíncrona del índice, es probable que se genere una falsa alarma para la reorganización del índice o para la limpieza del índice basadas en las estadísticas inexactas. Una vez ejecutada la limpieza asíncrona del índice, todas las claves de índice que todavía pertenezcan a las particiones de datos desenlazadas que exijan una limpieza se eliminarán, y este proceso puede evitar la necesidad de reorganizar el índice.

En el caso de las tablas particionadas, conviene que emita el mandato RUNSTATS en cuanto se haya terminado una limpieza asíncrona del índice para generar estadísticas de índices exactas en presencia de las particiones de datos desenlazadas. Para determinar si hay o no particiones de datos desenlazadas en la tabla, puede comprobar el campo de estado en la tabla SYSDATAPARTITIONS y buscar el valor I (limpieza del índice) o D (desenlazado con MQT dependiente).

El mandato RUNSTATS recopila estadísticas para todas las particiones de índice de un índice particionado. Las estadísticas de la vista SYSTAT.INDEXES para el índice particionado representan una partición de índice, excepto para las estadísticas FIRSTKEYCARD, FIRST2KEYCARD, FIRST3KEYCARD, FIRST4KEYCARD y FULLKEYCARD. Dado que estas estadísticas se utilizan en estimaciones de cardinalidad, son para todo el índice y no para una partición de índice. Las estadísticas de distribución (valores frecuentes y cuantiles) no se recopilan para índices particionados, pero se recogerán si se ejecuta RUNSTATS en la tabla. Es posible que las estadísticas de las columnas iniciales de un índice particionado no sean tan precisas como las estadísticas de las columnas iniciales de un índice no particionado.

2. El estado de la ejecución del mandato se devuelve en el área de comunicaciones SQL (SQLCA) obtenida a partir de la sentencia CALL.
3. Conviene ejecutar el mandato RUNSTATS:
 - En las tablas que se han modificado de forma considerable (por ejemplo, si se ha realizado un gran número de actualizaciones, si se ha insertado o suprimido una cantidad notable de datos o si se ha ejecutado LOAD sin la opción de estadísticas durante LOAD).
 - En las tablas que se han reorganizado (utilizando REORG, REDISTRIBUTE DATABASE PARTITION GROUP).
 - En las tablas cuyas filas se han comprimido.
 - Cuando se ha creado un índice nuevo.
 - Antes de vincular aplicaciones cuyo rendimiento sea importante.
 - Al cambiar la cantidad de captación previa.
 - En las vistas estadísticas cuyas tablas subyacentes se han modificado sustancialmente, a fin de cambiar las filas devueltas por la vista.
 - Después de ejecutar LOAD con la opción STATISTICS, utilice el programa de utilidad RUNSTATS para recopilar estadísticas sobre columnas XML. Las

estadísticas de columnas XML no se recopilan nunca durante el proceso de LOAD, ni siquiera cuando LOAD se ejecute con la opción STATISTICS. Cuando el programa de utilidad RUNSTATS se utiliza solo para recopilar estadísticas de columnas XML, se retienen las estadísticas existentes de columnas no XML que se hayan recopilado mediante LOAD o mediante una ejecución anterior del programa de utilidad RUNSTATS. Cuando se hayan recopilado anteriormente estadísticas para algunas columnas XML, esas estadísticas se descartarán si el mandato actual no recopila ninguna estadística para esa columna XML, o las estadísticas se sustituyen si el mandato actual recopila estadísticas para la columna XML.

4. Las opciones elegidas deben depender de la tabla y la aplicación específicas. En general:
 - Si la tabla es muy importante en las consultas importantes, es relativamente pequeña o no cambia demasiado y no hay mucha actividad en el propio sistema, puede que valga la pena el esfuerzo de reunir estadísticas con la mayor cantidad posible de detalles.
 - Si el tiempo para reunir estadísticas está limitado, si la tabla es relativamente grande o se actualiza con frecuencia, puede ser preferible ejecutar RUNSTATS limitado al conjunto de las columnas que se utilizan en los predicados. De esta manera, podrá ejecutar más a menudo el mandato RUNSTATS.
 - Si el tiempo para reunir estadísticas es muy limitado y el esfuerzo que supone adaptar el mandato RUNSTATS en una tabla según la base de la tabla es un problema importante, estudie la posibilidad de reunir estadísticas solo para las columnas "KEY". Se supone que el índice contiene el conjunto de columnas importantes para la tabla y es más probable que aparezca en predicados.
 - Si el tiempo para recopilar estadísticas es muy limitado y se deben reunir estadísticas de tabla, puede utilizar la opción TABLESAMPLE para recopilar estadísticas sobre un subconjunto de los datos de la tabla.
 - Si hay muchos índices en la tabla y la información DETAILED (ampliada) en los índices puede mejorar los planes de acceso, plantéese la posibilidad de usar la opción SAMPLED con el fin de reducir el tiempo necesario para reunir estadísticas.
 - Si hay alguna desviación en ciertas columnas y predicados del tipo "column = constant", puede ser preferible especificar un valor de NUM_FREQVALUES que sea mayor para esa columna
 - Reúna estadísticas de distribución para todas las columnas que se utilizan en predicados de igualdad y para las que puede haber una desviación en la distribución de los valores.
 - Para las columnas que tienen predicados de rango (por ejemplo, "column >= constant", "column BETWEEN constant1 AND constant2") o del tipo "column LIKE '%xyz'", puede ser preferible especificar un valor de NUM_QUANTILES que sea mayor.
 - Si el espacio de almacenamiento es motivo de preocupación y no se puede disponer de mucho tiempo para reunir estadísticas, no especifique valores elevados de NUM_FREQVALUES o NUM_QUANTILES para las columnas que no se utilicen en predicados.
 - Si se solicitan estadísticas de índice, y no se han ejecutado nunca estadísticas para la tabla donde reside el índice, se calculan estadísticas tanto para la tabla como para los índices.
 - Si no se necesitan estadísticas de columnas XML en la tabla, se puede usar la opción EXCLUDING XML COLUMNS para excluir todas las columnas

XML. Esta opción tiene prioridad sobre las demás cláusulas que especifiquen columnas XML para la recopilación de estadísticas.

5. Después de haber ejecutado el mandato, tenga en cuenta lo siguiente:
 - Hay que emitir un COMMIT para liberar los bloqueos.
 - Para permitir que se generen planes de acceso nuevos, los paquetes que hacen referencia a la tabla de destino deben volverse a vincular.
 - La ejecución del mandato en partes de la tabla puede producir incoherencias como resultado de la actividad en la tabla desde que se emitió el mandato por última vez. En este caso, se devuelve un mensaje de aviso. La emisión de RUNSTATS en la tabla solo podría provocar incoherencias entre las estadísticas a nivel de tabla y a nivel de índice. Por ejemplo, podría reunir estadísticas a nivel de índice en una tabla y, posteriormente, suprimir un número significativo de filas de la tabla. Si luego solo emite RUNSTATS en la tabla, la cardinalidad de la tabla podría ser menor que FIRSTKEYCARD, lo que es una incoherencia. De la misma manera, si reúne estadísticas en un índice nuevo al crearlo, las estadísticas a nivel de tabla pueden ser incoherentes.
6. El mandato RUNSTATS descartará las estadísticas de distribución reunidas con anterioridad si se solicitan estadísticas de tabla. Por ejemplo, RUNSTATS ON TABLE, o RUNSTATS ON TABLE ... AND INDEXES ALL, hará que se descarten las estadísticas de distribución recogidas con anterioridad. Si el mandato se ejecuta únicamente sobre índices, se retendrán las estadísticas de distribución recopiladas con anterioridad. Por ejemplo, RUNSTATS ON TABLE ... FOR INDEXES ALL hará que se retengan las estadísticas de distribución recogidas con anterioridad. Si el mandato RUNSTATS solo se ejecuta en columnas XML, se retendrán las estadísticas básicas de columna y las estadísticas de distribución recopiladas con anterioridad. Cuando se hayan recopilado anteriormente estadísticas para algunas columnas XML, esas estadísticas se descartarán si el mandato actual no recopila ninguna estadística para esa columna XML, o las estadísticas se sustituyen si el mandato actual recopila estadísticas para la columna XML.
7. Para las tablas agrupadas en clúster de rangos hay un índice especial generado por el sistema en las tablas de catálogos que representa la propiedad de ordenación del rango de las tablas agrupadas en clúster de rangos. Cuando se recopilan estadísticas sobre este tipo de tabla, si la tabla se debe incluir como parte de la recopilación de estadísticas, también se recopilarán estadísticas para el índice generado por el sistema. Las estadísticas reflejan el acceso rápido a las búsquedas de rangos representando el índice como un índice de dos niveles con tantas páginas como la tabla de datos base y agrupando perfectamente los datos base en clústeres a lo largo del orden del índice.
8. En la cláusula En Dist Cols de la sintaxis de mandatos, los parámetros Opción de frecuencia y Opción de cuantil no están soportados actualmente para la columna GROUPS. Estas opciones están soportadas para columnas individuales.
9. Hay tres estadísticas de captación previa que no se pueden calcular si se trabaja en modalidad DMS. Cuando observe las estadísticas de índice en los catálogos de índice, verá un valor -1 para las estadísticas siguientes:
 - AVERAGE_SEQUENCE_FETCH_PAGES
 - AVERAGE_SEQUENCE_FETCH_GAP
 - AVERAGE_RANDOM_FETCH_PAGES
10. El muestreo de RUNSTATS mediante TABLESAMPLE sólo se realiza con páginas de datos de tablas y no con páginas de índice. Cuando se necesitan estadísticas y también muestreos de índice, se exploran todas las páginas de

índice para recopilar estadísticas. TABLESAMPLE solo es aplicable en la recopilación de estadísticas de tabla. No obstante, se puede realizar una recopilación más eficaz de estadísticas de índice detalladas mediante la opción SAMPLED DETAILED. Es un método de muestreo distinto al que emplea TABLESAMPLE y solo se aplica al conjunto detallado de estadísticas de índice.

11. Se puede establecer o actualizar un perfil de estadísticas para la tabla o vista estadística especificada en el mandato RUNSTATS utilizando las opciones para establecer perfiles o actualizar perfiles. El perfil de estadísticas se almacena en un formato de tipo serie visible, que representa el mandato RUNSTATS, en la columna STATISTICS_PROFILE de la tabla de catálogos del sistema SYSIBM.SYSTABLES.
12. La recopilación de estadísticas en las columnas de tipo XML se rige mediante dos valores de registro de sistemas de base de datos DB2: DB2_XML_RUNSTATS_PATHID_K y DB2_XML_RUNSTATS_PATHVALUE_K. Estos dos parámetros se parecen al parámetro NUM_FREQVALUES en el sentido de que especifican los valores de número de frecuencia que hay que recopilar. Si no se establecen, se usará un valor por omisión igual a 200 para ambos parámetros.
13. RUNSTATS adquiere un bloqueo de tabla IX en SYSTABLES y un bloqueo U en la fila de la tabla en la que se reúnen estadísticas, al principio de RUNSTATS. Las operaciones todavía pueden leer en SYSTABLES que incluye la fila con el bloqueo U. Las operaciones de grabación también son posibles, siempre y cuando no se produzcan con respecto a la fila que tiene el bloqueo U. Sin embargo, otro lector o grabador no podrá adquirir el bloqueo S sobre SYSTABLES debido al bloqueo IX de RUNSTATS'.
14. No se recopilan estadísticas para columnas con tipos estructurados. Si se especifican, las columnas con estos tipos de datos se ignoran.
15. Sólo se recopilan AVGCOLLEN y NUMNULLS para las columnas con tipos de datos LOB o LONG.
16. AVGCOLLEN representa el espacio promedio en bytes si la columna se guarda en la memoria de base de datos o en una tabla temporal. Este valor representa la longitud del descriptor de datos para tipos de datos LOB o LONG, excepto cuando los datos LOB están en línea en la página de datos.

Nota: El espacio promedio necesario para almacenar la columna en el disco puede ser diferente del valor representado por esta estadística.

Mandato SET TAPE POSITION utilizando el procedimiento ADMIN_CMD

Establece que las posiciones de las cintas para las operaciones de copia de seguridad y restauración sean dispositivos de cinta en modalidad continua. Este mandato sólo está soportado en los sistemas operativos Windows.

Autorización

Una de las siguientes:

- *sysadm*
- *sysctrl*
- *sysmaint*

Conexión necesaria

Base de datos

Sintaxis del mandato

► SET TAPE POSITION *ON dispositivo* TO *posición* ►

Parámetros del mandato

ON *dispositivo*

Especifica un nombre de dispositivo de cinta válido. El valor por omisión es `\\.\TAPE0`. El dispositivo especificado debe ser relativo al servidor.

TO *posición*

Especifica la marca en la que debe colocarse la cinta. DB2 para Windows graba una marca de cinta después de cada imagen de copia de seguridad. Un valor de 1 especifica la primera posición, 2 especifica la segunda posición, etc. Si la cinta se coloca en la marca de cinta 1, por ejemplo, el archivo 2 queda colocado para restaurarse.

Ejemplo

Puesto que DB2 graba una marca de cinta después de cada imagen de copia de seguridad, si se especifica la posición 1 la cinta se moverá al principio del segundo archivado presente en ella.

```
CALL SYSPROC.ADMIN_CMD( 'set tape position to 1' )
```

Notas de uso

El estado de la ejecución del mandato se devuelve en el área de comunicaciones SQL (SQLCA) obtenida a partir de la sentencia CALL.

Mandato UNQUIESCE DATABASE utilizando el procedimiento ADMIN_CMD

Restaura el acceso de usuario a las bases de datos que se han inmovilizado para el mantenimiento u otras razones. UNQUIESCE restaura el acceso de usuario sin necesitar la conclusión y el reinicio de la base de datos.

A menos que se designe específicamente, ningún usuario salvo los que disponen de *sysadm*, *sysmaint* o *sysctrl* tiene acceso a una base de datos mientras esté inmovilizada. Por consiguiente, es necesario realizar UNQUIESCE para restaurar el acceso general a una base de datos inmovilizada.

Ámbito

UNQUIESCE DB restaura el acceso de los usuarios a todos los objetos de la base de datos inmovilizada.

Para detener la instancia y eliminar su inmovilización y la de todas sus bases de datos, emita el mandato `db2stop`. Al detener y reiniciar DB2 se eliminará la inmovilización de todas las instancias y bases de datos.

Autorización

Una de las siguientes:

Para eliminar la inmovilización a nivel de base de datos:

- *sysadm*
- *dbadm*

Sintaxis del mandato

►►—UNQUIESCE—DB—◄◄

Conexión necesaria

Base de datos

Parámetros del mandato

DB Eliminar la inmovilización de la base de datos. El acceso de usuarios se restaurará en todos los objetos de la base de datos.

Ejemplos

Eliminación de la inmovilización de una base de datos

```
CALL SYSPROC.ADMIN_CMD( 'unquiesce db' )
```

Este mandato eliminará la inmovilización de la base de datos que se haya inmovilizado anteriormente.

Notas de uso

El estado de la ejecución del mandato se devuelve en el área de comunicaciones SQL (SQLCA) obtenida a partir de la sentencia CALL.

Mandato UPDATE ALERT CONFIGURATION utilizando el procedimiento ADMIN_CMD

Actualiza los valores de configuración de alertas para los indicadores de salud.

Importante: Este mandato o API ha quedado en desuso y puede que se elimine en un futuro release ya que el Supervisor de salud ha quedado en desuso en la Versión 9.7. Para obtener más información, consulte el tema “El Supervisor de salud ha quedado en desuso” de la publicación *Novedades en DB2 Versión 9.7*.

Autorización

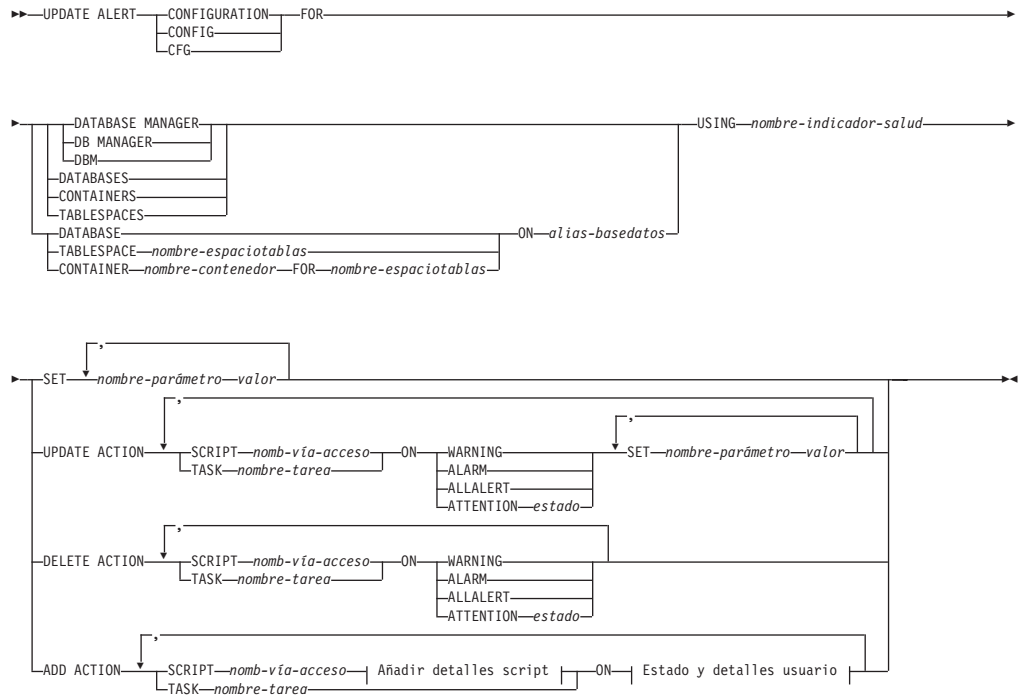
Una de las siguientes:

- *sysadm*
- *sysmaint*
- *sysctrl*

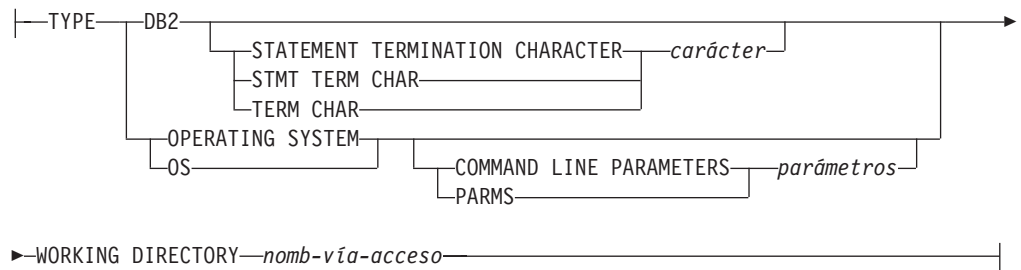
Conexión necesaria

Base de datos

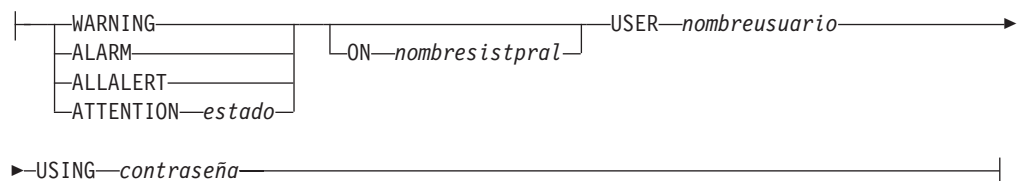
Sintaxis del mandato



Añadir detalles script:



Estado y detalles usuario:



Parámetros del mandato

DATABASE MANAGER

Actualiza valores de alerta para el gestor de bases de datos.

DATABASES

Actualiza valores de alerta para todas las bases de datos gestionadas por el gestor de bases de datos. Estos son los valores que se aplican a todas las bases de datos que no tienen valores personalizados. Los valores personalizados se definen utilizando la cláusula `DATABASE ON alias-basedatos`.

CONTAINERS

Actualiza valores de alerta para todos los contenedores de espacios de tablas gestionados por el gestor de bases de datos. Estos son los valores que se aplican a todos los contenedores de espacios de tablas que no tienen valores personalizados. Los valores personalizados se definen utilizando la cláusula `CONTAINER nombre-contenedor ON alias-basedatos`.

TABLESPACES

Actualiza valores de alerta para todos los espacios de tablas gestionados por el gestor de bases de datos. Estos son los valores que se aplican a todos los espacios de tablas que no tienen valores personalizados. Los valores personalizados se definen utilizando la cláusula `TABLESPACE nombre-espaciotablas ON alias-basedatos`.

DATABASE ON *alias-basedatos*

Actualiza los valores de alerta para la base de datos especificada utilizando la cláusula `ON alias-basedatos`. Si esta base de datos tiene valores personalizados, estos alteran temporalmente los valores para todas las bases de datos para la instancia, que se especifica mediante el parámetro `DATABASES`.

CONTAINER *nombre-contenedor* FOR *nombre-espaciotablas* ON *alias-basedatos*

Actualiza los valores de alerta para el contenedor de espacio de tablas denominado *nombre-contenedor*, para el espacio de tablas especificado utilizando la cláusula `FOR nombre-espaciotablas`, en la base de datos especificada utilizando la cláusula `ON alias-basedatos`. Si este contenedor de espacio de tablas tiene valores personalizados, prevalecerán sobre los valores de todos los contenedores de espacios de tablas de la base de datos, que se especifica utilizando el parámetro `CONTAINERS`.

TABLESPACE *nombre-espaciotablas* ON *alias-basedatos*

Actualiza los valores de alerta para el espacio de tablas denominado *nombre*, en la base de datos especificada utilizando la cláusula `ON alias-basedatos`. Si este espacio de tablas tiene valores personalizados, prevalecerán sobre los valores de todos los espacios de tablas de la base de datos que se especifica utilizando el parámetro `TABLESPACES`.

USING *nombre-indicador-salud*

Especifica el conjunto de indicadores de salud para los que se actualizará la configuración de alerta. Los nombres de indicador de salud se componen de un identificador de objeto de dos letras, seguido de un nombre que describe lo que mide el indicador. Por ejemplo:

```
db.sort_privmem_util
```

SET *valor nombre-parámetro*

Actualiza el elemento de configuración de alertas, *nombre-parámetro*, del indicador de salud al valor especificado. *nombre-parámetro* debe ser uno de los siguientes:

- **ALARM**: el *valor* es una unidad de indicador de salud.
- **WARNING**: el *valor* es una unidad de indicador de salud.
- **SENSITIVITY**: el *valor* se expresa en segundos.

- ACTIONSENABLED: el *valor* puede ser YES o NO.
- THRESHOLDSCHECKED: el *valor* puede ser YES o NO.

La lista de las posibles unidades de indicador de salud de la versión de DB2 específica se puede reunir ejecutando la consulta:

```
SELECT SUBSTR(UNIT,1,80) AS UNIT
FROM TABLE(HEALTH_GET_IND_DEFINITION(' ')) AS T GROUP BY UNIT
```

UPDATE ACTION SCRIPT *nomb-vía-acceso* ON [WARNING | ALARM | ALLALERT | ATTENTION *estado*]

Especifica que los atributos de script del script predefinido con el nombre de vía de acceso absoluta *nomb-vía-acceso* se actualizará según la cláusula siguiente:

SET *valor nombre-parámetro*

Actualiza el atributo de script, *nombre-parámetro*, al valor especificado. *nombre-parámetro* debe ser uno de los siguientes:

- SCRIPTTYPE
- WORKINGDIR
- TERMCHAR
- CMDLINEPARMS

Los tipos válidos son OS o DB2.

Los parámetros de línea de mandatos que especifique para el script del sistema operativo precederán los parámetros suministrados por omisión. Los parámetros que se envían al script del sistema operativo son:

- Lista de parámetros suministrados por el usuario
- Nombre corto del indicador de salud
- Nombre totalmente calificado del objeto
- Valor de indicador de salud
- Estado de alerta

- USERID
- PASSWORD
- SYSTEM

UPDATE ACTION TASK *nombre-tarea* ON [WARNING | ALARM | ALLALERT | ATTENTION *estado*]

Especifica que los atributos de tarea de la tarea con el nombre *nombre* se actualizará según la cláusula siguiente:

SET *valor nombre-parámetro*

Actualiza el atributo de tarea, *nombre-parámetro*, al valor especificado. *nombre-parámetro* debe ser uno de los siguientes:

- USERID
- PASSWORD
- SYSTEM

DELETE ACTION SCRIPT *nomb-vía-acceso* ON [WARNING | ALARM | ALLALERT | ATTENTION *estado*]

Elimina el script de acción con el nombre de vía de acceso absoluta *nombrevía-acceso* de la lista de scripts de acción de alerta.

DELETE ACTION TASK *nombre-tarea* ON [WARNING | ALARM | ALLALERT | ATTENTION *estado*]

Elimina la tarea de acción llamada *nombre* de la lista de tareas de acción de alerta.

ADD ACTION SCRIPT *nomb-vía-acceso* ON [WARNING | ALARM | ALLALERT | ATTENTION *estado*]

Especifica que se va a añadir un nuevo script de acción con el nombre de vía de acceso absoluta *nomb-vía-acceso*, cuyos atributos se proporcionan según lo siguiente:

TYPE Un script de acción debe ser un script de Mandato de DB2 o un script de sistema operativo:

- DB2
- OPERATING SYSTEM

Si es un script de Mandato de DB2, la cláusula siguiente permite especificar opcionalmente el carácter, *carácter*, que se utiliza en el script para terminar las sentencias:

STATEMENT TERMINATION CHARACTER ;

Si es un script de sistema operativo, la cláusula siguiente permite especificar opcionalmente los parámetros de línea de mandatos, *parámetros*, que se pasarían al script con la invocación: COMMAND LINE PARAMETERS *parámetros*.

WORKING DIRECTORY *nomb-vía-acceso*

Especifica el nombre de vía de acceso, nombre-vía, del directorio en el que se ejecutará el script.

USER *nombusuario* **USING** *contraseña*

Especifica la cuenta de usuario, *nombusuario*, y la contraseña asociada, *contraseña*, bajo la que se ejecutará el script. Al utilizar la opción ADD ACTION, el *nombusuario* y la *contraseña* pueden quedar expuestos en la red (donde el *nombusuario* y la *contraseña* se envían sin cifrado), el archivo de anotaciones cronológicas db2diag, los archivos de rastreo, el archivo de vuelco, el supervisor de instantáneas (instantánea SQL dinámica), las instantáneas del supervisor del sistema, una serie de supervisores de sucesos (como sentencia, punto muerto), el Query Patroller, las tablas de Explain, la salida de db2pd (como los mecanismos de antememoria de paquete y tiempo de espera excedido de bloqueo) y los registros de auditoría de db2.

ADD ACTION TASK *nombre* ON [WARNING | ALARM | ALLALERT | ATTENTION *estado*]

Especifica que se va a añadir una nueva tarea, llamada *nombre*, para ejecutarla en (ON) la condición especificada.

ON [WARNING | ALARM | ALLALERT | ATTENTION *estado*]

Especifica la condición en la que se ejecutará la acción o tarea. Para los indicadores de salud (HI) basados en el umbral, es WARNING o ALARM. Para los HI basados en el estado, puede ser un estado numérico tal como está documentado para cada HI basado en el estado (por ejemplo, para el indicador de salud ts.ts_op_status, consulte el elemento del supervisor tablespace_state para los estados de espacios de tablas) o un identificador de texto para este estado.

ATTENTION *estado*

A continuación se suministran valores numéricos válidos para algunos de los estados del indicador de salud de la base de datos a modo de ejemplo para la opción del mandato ADD ACTION SCRIPT CLP:

- 0 - Activo; Normal (ACTIVE)

- 1 - Inmovilización pendiente (QUIESCE_PEND)
- 2 - Inmovilizado (QUIESCED)
- 3 - Avance (ROLLFWD)

Los indicadores de salud adicionales basados en el estado se definen con los archivos de cabecera sqlmon.h y sqlutil.h.

El mandato UPDATE ALERT CFG invocado por el procedimiento almacenado ADMIN_CMD da soporte a un valor numérico o a un identificador de texto para *estado*. Los valores numéricos e identificadores de texto válidos para algunos estados de indicador de salud adicionales, a modo de ejemplo para el indicador de salud de estado operativo del espacio de tablas (ts.ts_op_status) son:

- 0x1 - QUIESCED_SHARE
- 0x2 - QUIESCED_UPDATE
- 0x4 - QUIESCED_EXCLUSIVE

Utilizar el mandato UPDATE ALERT CFG y los anteriores valores de indicador de estado, la siguiente entrada de línea de mandato, ADD ACTION SCRIPT ... ON ATTENTION 2

es equivalente a

```
ADD ACTION SCRIPT ... ON ATTENTION QUIESCED_UPDATE
```

Además, para el indicador de salud de estado operativo del espacio de tablas (ts.ts_op_status), se pueden especificar varios estados utilizando un solo valor numérico mediante los estados OR'ing juntos. Por ejemplo, puede especificar el estado 7 (= 0x1 + 0x2 + 0x4), la acción se realizará cuando el espacio de tablas entre en uno cualquiera de los estados Inmovilizado: SHARE, Inmovilizado: UPDATE o Inmovilizado: EXCLUSIVE. Como alternativa, podría especificar QUIESCED_SHARE, QUIESCED_UPDATE y QUIESCED_EXCLUSIVE en tres ejecuciones de mandatos UPDATE ALERT CFG separadas.

Ejemplo

Añadir una acción para el indicador db.log_fs_util que ejecutará el script /home/test/scripts/logfsutilact cuando existe una alarma en el sistema cuyo nombre de sistema principal es 'plato'.

```
CALL SYSPROC.ADMIN_CMD( 'update alert cfg for databases using
db.log_fs_util add action script /home/test/scripts/logfsutilact
type os command line parameters "param1 param2" working
directory /tmp on alarm on plato user dricard using mypasswv' )
```

Para comprobar la configuración de alerta después de haberla establecido, se pueden utilizar las funciones de tabla HEALTH_GET_IND_DEFINITION y HEALTH_GET_ALERT_ACTION_CFG, de la siguiente manera:

```
SELECT OBJECTTYPE, ID, CONDITION, ACTIONTYPE,
SUBSTR(ACTIONNAME,1,50) AS ACTION_NAME
FROM TABLE(SYSPROC.HEALTH_GET_ALERT_ACTION_CFG('DB','G',' ',''))
AS ALERT_ACTION_CFG
```

El siguiente ejemplo muestra una salida de esta consulta:

| OBJECTTYPE | ID | CONDITION | ACTIONTYPE | ACTION_NAME |
|------------|------|-----------|------------|------------------------------------|
| DB | 1006 | ALARM | S | /home/dricard/scripts/logfsutilact |

1 registro(s) seleccionado(s).

Notas de uso

Para la opción ADD ACTION, el *nombreusuario* y la *contraseña* pueden exponerse en diversos lugares donde se captura el texto de la sentencia SQL:

- la red (nombreusuario/contraseña se pasan a través del cable no cifrado)
- archivo de anotaciones cronológicas db2diag
- archivos de rastreo
- archivo de vuelco
- supervisor de instantáneas (instantánea SQL dinámica)
- instantáneas del supervisor del sistema
- un número de supervisores de sucesos (sentencia, punto muerto)
- Query Patroller
- tablas de Explain
- salida de db2pd (mecanismos de antememoria de paquete y tiempo de espera excedido de bloqueo, entre otros)
- registros de auditoría de DB2

El estado de la ejecución del mandato se devuelve en el área de comunicaciones SQL (SQLCA) obtenida a partir de la sentencia CALL.

El *alias-basedatos* debe estar definido en el catálogo en el servidor y ser local para el servidor.

El *nombre-vía* debe ser un nombre totalmente calificado de vía de acceso del servidor.

Mandato UPDATE CONTACT utilizando el procedimiento ADMIN_CMD

Actualiza los atributos de un contacto definido en el sistema local. Un contacto es un usuario al que el Planificador y el Supervisor de salud envían mensajes. Para crear un contacto, utilice el mandato ADD CONTACT. El valor del parámetro de configuración **contact_host** del Servidor de administración de bases de datos (DAS) determina si la lista es local o global.

Autorización

Ninguna

Conexión necesaria

Base de datos. El DAS debe estar en ejecución.

Sintaxis del mandato

```
►► UPDATE CONTACT nombre USING palabraclave valor ◀◀
```

Parámetros del mandato

UPDATE CONTACT *nombre*

Nombre del contacto que se actualizará.

USING *palabraclave valor*

Especifica el parámetro de contacto que hay que actualizar (*palabraclave*) y el valor con el que se establecerá (*valor*). El conjunto válido de palabras clave es:

ADDRESS

Dirección de correo electrónico que utiliza el servidor SMTP para enviar la notificación.

TYPE Si la dirección corresponde a una dirección de correo electrónico o a un buscpersonas.

MAXPAGELEN

Número máximo de caracteres que el buscpersonas puede aceptar.

DESCRIPTION

Texto descriptivo del contacto. Tiene una longitud máxima de 128 caracteres.

Ejemplo

Actualizar la dirección de usuario 'test' con 'newaddress@test.com'.

```
CALL SYSPROC.ADMIN_CMD( 'update contact test using address newaddress@test.com' )
```

Notas de uso

El DAS ya se debe haber creado y estar en ejecución.

El estado de la ejecución del mandato se devuelve en el área de comunicaciones SQL (SQLCA) obtenida a partir de la sentencia CALL.

Mandato UPDATE CONTACTGROUP utilizando el procedimiento ADMIN_CMD

Actualiza los atributos de un grupo de contactos definido en el sistema local. Un grupo de contactos es una lista de direcciones a los que el Planificador y el Supervisor de salud deben enviar notificaciones. El valor del parámetro de configuración **contact_host** del Servidor de administración de bases de datos (DAS) determina si la lista es local o global.

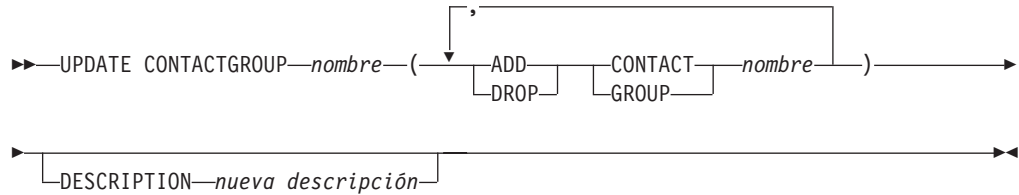
Autorización

Ninguna

Conexión necesaria

Base de datos. El DAS debe estar en ejecución.

Sintaxis del mandato



Parámetros del mandato

CONTACTGROUP *nombre*

Nombre del grupo de contactos que se actualizará.

ADD CONTACT *nombre*

Especifica el nombre del nuevo contacto que se añadirá al grupo. Se puede definir un contacto con el mandato ADD CONTACT después de haberlo añadido a un grupo.

DROP CONTACT *nombre*

Especifica el nombre de un contacto del grupo que se descartará del grupo.

ADD GROUP *nombre*

Especifica el nombre del nuevo grupo de contactos que se añadirá al grupo.

DROP GROUP *nombre*

Especifica el nombre de un grupo de contactos que se descartará del grupo.

DESCRIPTION *nueva descripción*

Opcional. Texto descriptivo nuevo del grupo de contactos.

Ejemplo

Añadir el contacto llamado 'cname2' al grupo de contactos llamado 'gname1':

```
CALL SYSPROC.ADMIN_CMD( 'update contactgroup gname1 add contact cname2' )
```

Notas de uso

El DAS ya se debe haber creado y estar en ejecución.

El estado de la ejecución del mandato se devuelve en el área de comunicaciones SQL (SQLCA) obtenida a partir de la sentencia CALL.

Mandato UPDATE DATABASE CONFIGURATION utilizando el procedimiento ADMIN_CMD

Modifica entradas individuales de un archivo de configuración de base de datos específico.

En cada partición de base de datos en la que se ha creado la base de datos reside un archivo de configuración de base de datos.

Ámbito

Este mandato actualiza todas las particiones de base de datos por omisión, excepto cuando se especifica DBPARTITIONNUM para actualizar únicamente una partición de base de datos.

Autorización

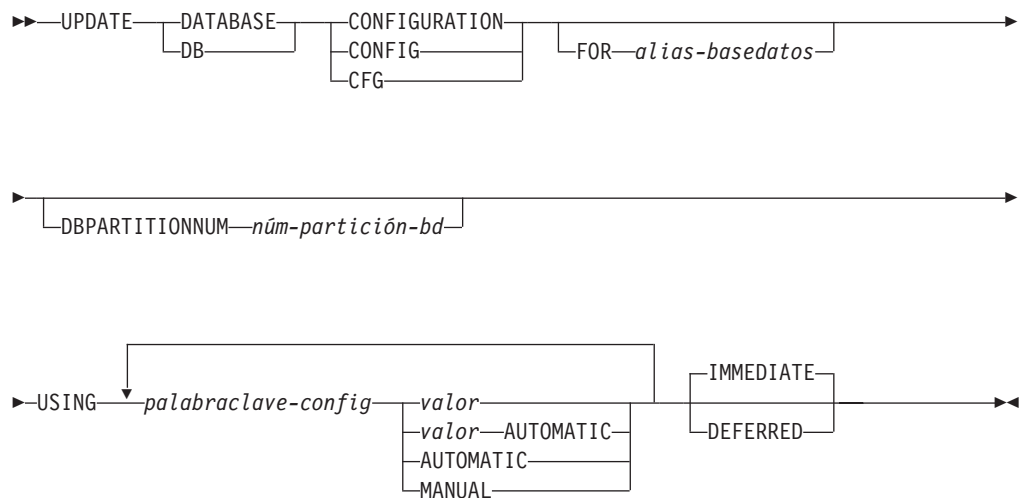
Una de las siguientes:

- *sysadm*
- *sysctrl*
- *sysmaint*

Conexión necesaria

Base de datos. La conexión de base de datos debe ser local con respecto a la instancia que contiene la base de datos conectada.

Sintaxis del mandato



Parámetros del mandato

AUTOMATIC

Algunos parámetros de configuración pueden establecerse en `AUTOMATIC`, para permitir que DB2 ajuste automáticamente estos parámetros para que reflejen las necesidades de recursos actuales. Para obtener una lista de parámetros de configuración que den soporte a la palabra clave `AUTOMATIC`, consulte el resumen de parámetros de configuración. Si se especifica un valor junto a la palabra clave `AUTOMATIC`, es posible que influya en los cálculos automáticos. Para obtener detalles específicos sobre este comportamiento, consulte la documentación para el parámetro de configuración.

Nota: Los parámetros de configuración **appl_memory**, **logindexbuild**, **max_log** y **num_log_span** sólo se pueden establecer en AUTOMATIC utilizando el procesador de línea de mandatos.

DEFERRED

Efectuar los cambios solo en el archivo de configuración, de forma que los cambios entren en vigor la próxima vez que reactive la base de datos.

FOR *alias-basedatos*

Especifica el alias de la base de datos cuya configuración debe actualizarse. No es necesario especificar el alias de base de datos cuando ya se ha establecido una conexión de base de datos. El alias de la base de datos debe definirse localmente en el servidor. Puede actualizar el archivo de configuración para otra base de datos que reside debajo de la misma instancia de base de datos. Por ejemplo, si únicamente está conectado a la base de datos db11 y emite `update db config for alias db22 using immediate:`

- Si no hay ninguna conexión activa en db22, la actualización se realizará correctamente porque sólo es necesario actualizar el archivo de configuración. Una conexión nueva (que activará la base de datos) reflejará el nuevo cambio en la memoria.
- Si no hay conexiones activas en db22 de otras aplicaciones, la actualización funcionará en el disco pero no en la memoria. Recibirá un aviso que indica que es necesario reiniciar la base de datos.

DBPARTITIONNUM *núm-partición-bd*

Si se tiene que aplicar una actualización de configuración de base de datos a una partición de base de datos específica, se puede utilizar este parámetro. Si no se proporciona este parámetro, la actualización tendrá efecto sobre todas las particiones de la base de datos.

IMMEDIATE

Efectuar los cambios inmediatamente, mientras se esté ejecutando la base de datos. IMMEDIATE es la acción por omisión. Dado que el procedimiento ADMIN_CMD exige una conexión con la base de datos, los cambios entrarán en vigor de inmediato para cualquier parámetro dinámicamente configurable de la base de datos conectada.

También es la cláusula por omisión cuando funciona en la interfaz CLPPlus. Si se utiliza el procesador CLPPlus, no es necesario invocar IMMEDIATE.

MANUAL

Inhabilita el ajuste automático para el parámetro de configuración. El parámetro se establece en su valor interno actual y ya no se actualiza automáticamente.

USING *palabraclave-config valor*

palabraclave-config especifica el parámetro de configuración de base de datos que se debe actualizar. *valor* especifica el valor que se debe asignar al parámetro.

Ejemplo

Establecer que el parámetro **sortheap** de configuración de la base de datos tenga el valor 1000 en la partición de base de datos a la que la aplicación está conectada en este momento.

```
CALL SYSPROC.ADMIN_CMD ('UPDATE DB CFG USING sortheap 1000')
```


Notas de uso

El estado de la ejecución del mandato se devuelve en el área de comunicaciones SQL (SQLCA) obtenida a partir de la sentencia CALL.

El *alias-basedatos* debe ser un nombre de alias que esté definido en el servidor.

El mandato afecta a todas las particiones de base de datos a menos que se especifique DBPARTITIONNUM.

Para ver o imprimir una lista de los parámetros de configuración de la base de datos, utilice la vista de administración de SYSIBMADM.DBCFG.

Para restablecer todos los parámetros de configuración de base de datos en los valores por omisión recomendados, utilice el mandato RESET DATABASE CONFIGURATION mediante el procedimiento ADMIN_CMD.

Para cambiar un parámetro de configuración de base de datos, utilice el mandato UPDATE DATABASE CONFIGURATION mediante el procedimiento ADMIN_CMD. Por ejemplo, para cambiar la modalidad de registro cronológico a “archival logging” en un entorno de base de datos de única partición que contenga una base de datos denominada ZELLMART, utilice:

```
CALL SYSPROC.ADMIN_CMD ('update db cfg for zellmart using logretain recovery')
```

Para comprobar si ha cambiado el parámetro de configuración **logretain**, utilice:

```
SELECT * FROM SYSIBMADM.DBCFG WHERE NAME='logretain'
```

Para actualizar un parámetro de configuración de base de datos en una partición de base de datos específica, puede:

1. establecer la variable DB2NODE en un número de partición de base de datos.
2. conectarse a la partición de base de datos.
3. actualizar los parámetros de configuración de base de datos UPDATE DATABASE CONFIGURATION mediante el procedimiento ADMIN_CMD.
4. desconectarse de la partición de base de datos.

o puede utilizar DBPARTITIONNUM. Por ejemplo, para actualizar la modalidad de registro a una sola partición específica (30) utilizando DBPARTITIONNUM, utilice:

```
CALL SYSPROC.ADMIN_CMD ('update db cfg for zellmart dbpartitionnum 30 using logretain recovery')
```

Para obtener más información acerca de los parámetros de configuración de DB2 y los valores disponibles para cada tipo de nodo de base de datos, vea las descripciones individuales de los parámetros de configuración. Los valores de estos parámetros difieren para cada tipo de nodo de base de datos configurado (servidor, cliente o servidor con clientes remotos).

No todos los parámetros pueden actualizarse.

Algunos cambios en el archivo de configuración de base de datos solo entran en vigor después de haberse cargado en la memoria. Para que esto pueda producirse, todas las aplicaciones deben desconectarse de la base de datos. Para obtener más información sobre qué parámetros se pueden configurar en línea y cuáles no, vea la lista de resumen de parámetros de configuración.

Si se produce un error, el archivo de configuración de base de datos no se modifica. El archivo de configuración de base de datos no se puede actualizar si la suma de comprobación no es válida. Esto podría producirse si se modifica el archivo de configuración de base de datos sin utilizar el mandato apropiado. Si sucede esto, se deberá restaurar la base de datos para restablecer el archivo de configuración de la misma.

Mandato UPDATE DATABASE MANAGER CONFIGURATION utilizando el procedimiento ADMIN_CMD

Modifica entradas individuales del archivo de configuración del gestor de bases de datos para la instancia que contiene la base de datos conectada actualmente..

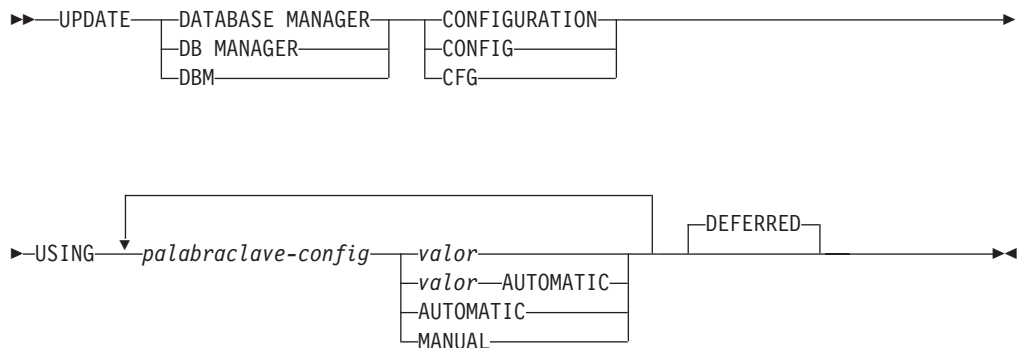
Autorización

sysadm

Conexión necesaria

Base de datos

Sintaxis del mandato



Parámetros del mandato

AUTOMATIC

Algunos parámetros de configuración pueden establecerse en `AUTOMATIC`, para permitir que DB2 ajuste automáticamente estos parámetros para que reflejen las necesidades de recursos actuales. Para obtener una lista de parámetros de configuración que den soporte a la palabra clave `AUTOMATIC`, consulte el resumen de parámetros de configuración. Si se especifica un valor junto a la palabra clave `AUTOMATIC`, es posible que influya en los cálculos automáticos. Para obtener detalles específicos sobre este comportamiento, consulte la documentación para el parámetro de configuración.

Nota: Tenga en cuenta que el parámetro de configuración del gestor de bases de datos `federated_async` sólo puede establecerse en `AUTOMATIC` mediante el procesador de línea de mandatos.

DEFERRED

Efectuar los cambios solo en el archivo de configuración, de forma que los cambios entren en vigor cuando se reinicie la instancia. Es el valor por omisión.

Es la cláusula por omisión cuando funciona en la interfaz CLPPlus. Si se utiliza el procesador CLPPlus, no es necesario invocar DEFERRED.

MANUAL

Inhabilita el ajuste automático para el parámetro de configuración. El parámetro se establece en su valor interno actual y ya no se actualiza automáticamente.

USING *palabraclave-config valor*

Especifica el parámetro de configuración del gestor de bases de datos que se debe actualizar. Para obtener una lista de parámetros de configuración, consulte el resumen de parámetros de configuración. *valor* especifica el valor que se debe asignar al parámetro.

Ejemplo

Actualizar el nivel de diagnóstico a 1 para la configuración del gestor de bases de datos.

```
CALL SYSPROC.ADMIN_CMD('db2 update dbm cfg using DIAGLEVEL 1')
```

Notas de uso

Para ver o imprimir una lista de los parámetros de configuración del gestor de bases de datos, utilice la vista administrativa SYSIBMADM.DBMCFG. Para restablecer los parámetros de configuración del gestor de bases de datos en los valores por omisión recomendados del gestor de bases de datos, utilice el mandato RESET DATABASE MANAGER CONFIGURATION mediante el procedimiento ADMIN_CMD. Para obtener más información acerca de los parámetros de configuración del gestor de bases de datos y los valores de estos parámetros que son adecuados para cada tipo de nodo de base de datos configurado (servidor, cliente o servidor con clientes remotos), vea las descripciones de los parámetros de configuración individuales.

No todos los parámetros pueden actualizarse.

Algunos cambios en el archivo de configuración del gestor de bases de datos solo entran en vigor después de haberse cargado en la memoria. Para obtener más información sobre qué parámetros se pueden configurar en línea y cuáles no, vea el resumen de parámetros de configuración. Los parámetros de configuración de servidor que no se restablecen inmediatamente, se restablecen durante la ejecución de db2start. Para un parámetro de configuración de cliente, los parámetros se restablecen la vez siguiente que se reinicia la aplicación. Si el cliente es el procesador de línea de mandatos, es necesario invocar TERMINATE.

Si se produce un error, el archivo de configuración del gestor de bases de datos no se modifica.

El archivo de configuración del gestor de bases de datos no se puede actualizar si la suma de comprobación no es válida. Esto puede producirse si edita el archivo de configuración del gestor de bases de datos y no utiliza el mandato adecuado. Si la suma de comprobación no es válida, debe volver a instalar el gestor de bases de datos para restablecer el archivo de configuración del gestor de bases de datos.

Cuando se actualizan los parámetros **SVCENAME** o **TPNAME** de configuración del gestor de bases de datos para la instancia actual, si se ha habilitado el soporte de LDAP y hay un servidor LDAP registrado para esta instancia, el servidor LDAP se actualizará con el valor o los valores nuevos.

El estado de la ejecución del mandato se devuelve en el área de comunicaciones SQL (SQLCA) obtenida a partir de la sentencia **CALL**.

Las actualizaciones solo se pueden hacer en la instancia de base de datos que contiene la base de datos conectada.

Si un parámetro permite utilizar la actualización dinámica, se intenta actualizarlo dinámicamente, aunque no se especifique la palabra clave **IMMEDIATE**. La autorización utilizada es el ID de **SYSTEM_USER** actual.

Mandato **UPDATE HEALTH NOTIFICATION CONTACT LIST** utilizando el procedimiento **ADMIN_CMD**

Actualiza la lista de contactos de notificación para las alertas de salud emitidas por una instancia.

Autorización

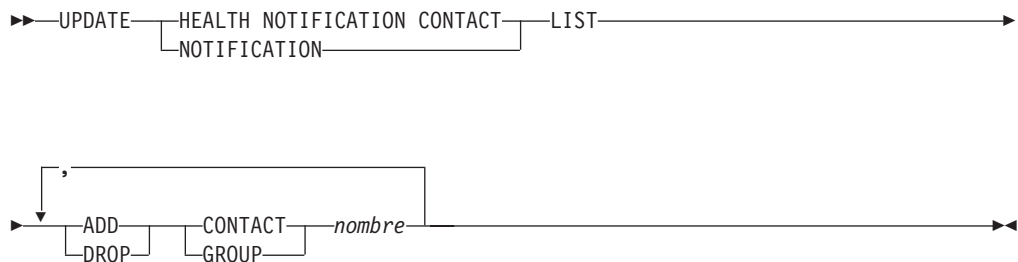
Una de las siguientes:

- sysadm
- sysctrl
- sysmaint

Conexión necesaria

Base de datos

Sintaxis del mandato



Parámetros del mandato

ADD GROUP *nombre*

Añade un nuevo grupo de contactos a los que se notificará la salud de la instancia.

ADD CONTACT *nombre*

Añade un nuevo contacto al que se notificará la salud de la instancia.

DROP GROUP *nombre*

Elimina el grupo de contactos de la lista de contactos a los que se notificará la salud de la instancia.

DROP CONTACT *nombre*

Elimina el contacto de la lista de contactos a los que se notificará la salud de la instancia.

Ejemplo

Añadir un grupo de contactos 'gname1' a la lista de contactos de notificación de salud:

```
CALL SYSPROC.ADMIN_CMD( 'update notification list add group gname1' )
```

Nota de uso

El estado de la ejecución del mandato se devuelve en el área de comunicaciones SQL (SQLCA) obtenida a partir de la sentencia CALL.

Mandato UPDATE HISTORY utilizando el procedimiento ADMIN_CMD

Actualiza la ubicación, tipo de dispositivo, comentario o estado en una entrada del archivo histórico en la partición de base de datos conectada actualmente.

Autorización

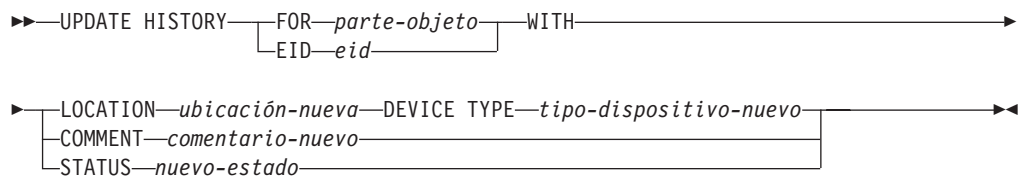
Una de las siguientes:

- *sysadm*
- *sysctrl*
- *sysmaint*
- *dbadm*

Conexión necesaria

Base de datos

Sintaxis del mandato



Parámetros del mandato

FOR *parte-objeto*

Especifica el identificador para la entrada del histórico que debe actualizarse. Es una indicación de fecha y hora con un número de secuencia opcional de 001 a 999. Ese parámetro no se puede utilizar para actualizar el estado de la entrada. Para actualizar el estado de la entrada, especifique en su lugar un EID.

EID *eid*

Especifica el ID de la entrada del histórico.

LOCATION *ubicación-nueva*

Especifica la ubicación física nueva de una imagen de copia de seguridad. La interpretación de este parámetro depende del tipo de dispositivo.

DEVICE TYPE *tipo-dispositivo-nuevo*

Especifica un tipo de dispositivo nuevo para almacenar la imagen de copia de seguridad. Los tipos de dispositivo válidos son:

- D Disco
- K Disquete
- T Cinta
- A Tivoli Storage Manager
- F Copia de seguridad instantánea
- U Salida de usuario
- P Área de conexión de memoria
- N Dispositivo nulo
- X XBSA
- Q Sentencia de SQL
- O Otro

COMMENT *comentario-nuevo*

Especifica un comentario nuevo para describir la entrada.

STATUS *nuevo-estado*

Especifica un nuevo estado para una entrada. Sólo se puede actualizar el estado de las entradas de copia de seguridad. Los valores válidos son:

- A Activa. La imagen de copia de seguridad está en la cadena de anotación cronológica activa. La mayoría de las entradas están activas.
- I Inactiva. Las imágenes de copia de seguridad que ya no se corresponden con la secuencia de anotaciones cronológicas actuales, que también se denomina cadena de anotaciones cronológicas actuales, están marcadas como inactivas.
- E Caducada. Las imágenes de copia de seguridad que ya no son necesarias porque hay más de NUM_DB_BACKUPS activas, están marcadas como caducadas.
- D Suprimida. Las imágenes de copia de seguridad que ya no están disponibles para su recuperación deberían marcarse como ya suprimidas.
- X No suprimir. Las entradas del archivo histórico de recuperación que están marcadas como DB2HISTORY_STATUS_DO_NOT_DELETE no se recortarán mediante las llamadas al mandato PRUNE HISTORY, ejecutando el procedimiento ADMIN_CMD con PRUNE HISTORY, las llamadas a la API db2Prune ni el recorte del archivo histórico de recuperación automatizada. Puede utilizar el estado DB2HISTORY_STATUS_DO_NOT_DELETE para impedir que las entradas del archivo de recuperación de claves se recorten y que se supriman los objetos de recuperación asociados a ellos. Sólo los archivos de registro, las imágenes de copia de seguridad y las

imágenes de copia de carga están marcados como DB2HISTORY_STATUS_DO_NOT_DELETE.

Ejemplo

Para actualizar la entrada del archivo histórico para la copia de seguridad entera de la base de datos realizada el 13 de abril de 1997 a las 10:00 a.m., entre:

```
CALL SYSPROC.ADMIN_CMD('update history
  for 199704131000000001 with location
  /backup/dbbackup.1 device type d')
```

Notas de uso

El propósito principal del archivo histórico de la base de datos es registrar información, pero los datos contenidos en el histórico son utilizados directamente por operaciones de restauración automática. Durante cualquier restauración donde se especifique la opción AUTOMATIC, el programa de utilidad de restauración utilizará y hará referencia al histórico de imágenes de copia de seguridad y a sus ubicaciones para satisfacer la petición de restauración automática. Si se va a utilizar la función de restauración automática y se ha cambiado la ubicación de las imágenes de copia de seguridad desde que se crearon, se recomienda que el registro del histórico de la base de datos para estas imágenes se actualice para reflejar la ubicación actual. Si la ubicación de las imágenes de copia de seguridad en el histórico de la base de datos no está actualizada, la restauración automática no podrá ubicar las imágenes de copia de seguridad, pero todavía se pueden utilizar satisfactoriamente los mandatos de restauración manual.

El estado de la ejecución del mandato se devuelve en el área de comunicaciones SQL (SQLCA) obtenida a partir de la sentencia CALL.

parte-objeto o *eid* deben hacer referencia a las entradas del histórico de anotaciones en la partición de base de datos conectada.

Mandato UPDATE STMM TUNING DBPARTITIONNUM mediante el procedimiento ADMIN_CMD

Actualizar la partición de base de datos de ajuste STMM (Self Tuning Memory Manager) preferida por el usuario.

Autorización

Los privilegios que posee el ID de autorización de la sentencia debe tener al menos una de las autorizaciones siguientes:

- DBADM
- DATAACCESS
- SQLADM

Conexión necesaria

Base de datos

Sintaxis del mandato

►►—UPDATE—STMM—TUNING—DBPARTITIONNUM—*núm-partición*—◀◀

Parámetro del mandato

núm-partición

núm-partición es un entero. Si se utiliza un número de partición de base de datos no existente o -1, DB2 seleccionará automáticamente una partición de base de datos adecuada en la que ejecutar el ajustador de memoria STMM.

Ejemplo

Actualizar la partición de base de datos de ajuste STMM (Self Tuning Memory Manager) preferida por el usuario a la partición de base de datos 3.

```
CALL SYSPROC.ADMIN_CMD( 'update stmm tuning dbpartitionnum 3' )
```

Notas de uso

El proceso de ajuste STMM comprueba periódicamente si se ha cambiado el valor del número de partición de base de datos de ajuste STMM preferido por el usuario. El proceso de ajuste STMM moverá la partición de base de datos de ajuste STMM preferida por el usuario si *núm-partición* existe y es una partición de base de datos activa. Cuando este mandato cambia el número de partición de base de datos de ajuste STMM se produce un cambio inmediato al número de partición de base de datos de ajuste STMM actual.

El estado de la ejecución del mandato se devuelve en el área de comunicaciones SQL (SQLCA) obtenida a partir de la sentencia CALL.

Este mandato confirma sus cambios en el procedimiento ADMIN_CMD.

Función ADMIN_EST_INLINE_LENGTH - Estimar la longitud necesaria en datos en línea

La función ADMIN_EST_INLINE_LENGTH devuelve una estimación de la longitud en línea necesaria para poner en línea los datos almacenados en una columna XML, BLOB, CLOB o DBCLOB.

Si los datos no se pueden poner en línea, la función devuelve un valor negativo.

Si los datos ya están en línea, la función devuelve la longitud real de los datos en línea.

Sintaxis

►►—ADMIN_EST_INLINE_LENGTH—(—*nombre-columna*—)—◀◀

El esquema es SYSIBM.

Valor de retorno

Esta función devuelve un valor INTEGER que representa la longitud en línea estimada (en bytes) de los datos o uno de los valores siguientes:

NULL Indica que las entradas son NULL.

- 1 Indica que los datos no se pueden poner en línea porque no hay una longitud en línea válida que permita poner en línea el valor de la columna.
- 2 Indica que la longitud en línea estimada del documento no se puede determinar porque el documento se insertó y almacenó en un release anterior a DB2 para Linux, UNIX y Windows Versión 9.7.

Parámetros de la función

nombre-columna

Identifica una columna de la tabla base con el tipo de datos XML, BLOB, CLOB o DBCLOB (SQLSTATE 42884). La columna debe hacer referencia, de forma directa o indirecta, a la columna de una tabla base que no se ha generado en función de una expresión (SQLSTATE 42815).

Ejemplo

Ejemplo 1: El ejemplo siguiente devuelve la longitud en línea estimada de tres documentos XML contenidos en la columna XML xml_doc1 de la tabla TAB1.

```
db2 => SELECT PK, ADMIN_IS_INLINED(xml_doc1) as IS_INLINED,
        ADMIN_EST_INLINE_LENGTH(xml_doc1) as EST_INLINE_LENGTH
        from TAB1
```

Esta consulta genera la salida siguiente:

| PK | IS_INLINED | EST_INLINE_LENGTH |
|----|------------|-------------------|
| 1 | 1 | 292 |
| 2 | 0 | 450 |
| 3 | 0 | 454 |

3 registro(s) seleccionado(s).

En el ejemplo, la función ADMIN_IS_INLINED indica que el primer documento está en línea. Por lo tanto, la función ADMIN_EST_INLINE_LENGTH devuelve la longitud real del documento XML en línea. El segundo documento no está en línea, por lo que la función ADMIN_EST_INLINE_LENGTH devuelve la longitud en línea estimada que se precisa para poner en línea el segundo documento XML.

Ejemplo 2: El ejemplo siguiente devuelve la longitud en línea estimada de un documento XML que está contenido en la columna XML xml_doc1 de la tabla TAB1. Este ejemplo incluye un predicado.

```
db2 => SELECT PK, ADMIN_IS_INLINED(xml_doc1) as IS_INLINED,
        ADMIN_EST_INLINE_LENGTH(xml_doc1) as EST_INLINE_LENGTH
        from TAB1 where PK=2
```

Esta consulta genera la salida siguiente:

| PK | IS_INLINED | EST_INLINE_LENGTH |
|----|------------|-------------------|
| 2 | 0 | 450 |

1 registro(s) seleccionado(s).

Ejemplo 3: El ejemplo siguiente devuelve la longitud en línea estimada de tres datos CLOB contenidos en la columna CLOB clob_1 de la tabla TAB1.

```
db2 => SELECT PK, ADMIN_IS_INLINED(clob_1) as IS_INLINED,
        ADMIN_EST_INLINE_LENGTH(clob_1) as EST_INLINE_LENGTH
        from TAB1
```

Esta consulta genera la salida siguiente:

| PK | IS_INLINED | EST_INLINE_LENGTH |
|----|------------|-------------------|
| 1 | 1 | 68 |
| 2 | 0 | 3665 |
| 3 | 0 | -1 |

3 registro(s) seleccionado(s).

Notas de uso

- Las columnas XML solamente están soportadas si los documentos XML se han insertado con DB2 para Linux, UNIX y Windows Versión 9.7 o con una versión posterior. Los documentos XML insertados antes de este release tienen un formato de almacenamiento diferente. Cuando la función `ADMIN_EST_INLINE_LENGTH` encuentra un formato de almacenamiento incorrecto, devuelve el valor -2.
- Si tiene la intención de aumentar la longitud en línea de las columnas, recuerde que esta longitud no se puede reducir.
- Al incrementar la longitud en línea también se incrementa el tamaño total de las filas, lo que puede afectar al rendimiento de las agrupaciones de almacenamientos intermedios. El tamaño total de las filas tiene los límites siguientes.

Tabla 66. Límites del tamaño de fila

| Tamaño de página | Límite del tamaño de fila | Límite de la longitud en línea |
|------------------|---------------------------|--------------------------------|
| 4K | 4.005 | 4001 |
| 8K | 8.101 | 8097 |
| 16K | 16.293 | 16.289 |
| 32K | 32.677 | 32.673 |

- Es probable que la longitud en línea estimada no sea exacta si el tamaño de página del objeto de almacenamiento XML no es el mismo que el tamaño de página de la tabla base.

Función de tabla `ADMIN_GET_DBP_MEM_USAGE` - Obtener el consumo de memoria total por instancia

La función de tabla `ADMIN_GET_DBP_MEM_USAGE` obtiene el consumo de memoria total para una instancia determinada.

La función de tabla `ADMIN_GET_DBP_MEM_USAGE` toma un argumento de entrada `núm-particiónbd` (tipo `INTEGER`) opcional, que especifica un número de partición de base de datos válido y devuelve únicamente estadísticas para la partición de la base de datos individual. Si se omite el argumento, las estadísticas se devolverán para todas las particiones de base de datos activa. Al utilizar la característica de particionamiento de bases de datos (DPF), si especifica -1 o un valor `NULL` para `núm-particiónbd`, los datos se devolverán a partir de la partición conectada actualmente.

Sintaxis

```

▶▶ ADMIN_GET_DBP_MEM_USAGE ( [núm-partición-bd] )

```

El esquema es SYSPROC.

Parámetros de la función de tabla

númparticiónbd

Argumento de entrada de tipo integer que especifica la partición de base de datos desde dónde se tomarán las estadísticas de uso de la memoria. Si se especifica -1 o el valor NULL, se devolverán datos de la partición conectada actualmente.

Autorización

Privilegio EXECUTE en la función ADMIN_GET_DBP_MEM_USAGE.

Información devuelta

Tabla 67. El conjunto de resultados para ADMIN_GET_DBP_MEM_USAGE

| Nombre de columna | Tipo de datos | Descripción |
|-----------------------|---------------|--|
| DBPARTITIONNUM | SMALLINT | El número de partición de bases de datos desde dónde se toman las estadísticas de uso de memoria. |
| MAX_PARTITION_MEM | BIGINT | La cantidad máxima de memoria de instancia (en bytes) que puede consumirse en la partición de bases de datos. |
| CURRENT_PARTITION_MEM | BIGINT | La cantidad de memoria de instancia (en bytes) consumida actualmente en la partición de bases de datos. |
| PEAK_PARTITION_MEM | BIGINT | El punto máximo o la marca de límite superior de consumo de memoria de instancia (en bytes) en la partición de bases de datos. |

Ejemplos

Ejemplo 1: Recuperar estadísticas de uso de memoria de la partición de bases de datos 3

```
SELECT * FROM TABLE (SYSPROC.ADMIN_GET_DBP_MEM_USAGE(3)) AS T
```

```
DBPARTITIONNUM  MAX_PARTITION_MEM  CURRENT_PARTITION_MEM  PEAK_PARTITION_MEM
-----
                3          500000000          381000000          481000000
```

1 registro(s) seleccionado(s).

Ejemplo 2: Recuperar las estadísticas de uso de memoria de la partición conectada actualmente (asumiendo que el usuario esté conectado a la partición 2 de bases de datos.)

```
SELECT * FROM TABLE (SYSPROC.ADMIN_GET_DBP_MEM_USAGE(-1)) AS T
```

```
DBPARTITIONNUM  MAX_PARTITION_MEM  CURRENT_PARTITION_MEM  PEAK_PARTITION_MEM
-----
                2          500000000          381000000          481000000
```

1 registro(s) seleccionado(s).

Ejemplo 3: Recuperar las estadísticas de uso de memoria de todas las particiones

```
SELECT * FROM TABLE (SYSPROC.ADMIN_GET_DBP_MEM_USAGE()) AS T
```

```
DBPARTITIONNUM  MAX_PARTITION_MEM  CURRENT_PARTITION_MEM  PEAK_PARTITION_MEM
-----
                0          500000000          381000000          481000000
```

| | | | |
|---|-----------|-----------|-----------|
| 1 | 500000000 | 381000000 | 481000000 |
| 2 | 500000000 | 381000000 | 481000000 |
| 3 | 500000000 | 381000000 | 481000000 |

4 registro(s) seleccionado(s).

Ejemplo 4: Recuperar las estadísticas de uso de memoria en valores de megabyte (MB)

```
SELECT DBPARTITIONNUM, MAX_PARTITION_MEM/1048576 AS MAX_MEM_MB,
       CURRENT_PARTITION_MEM/1048576 AS CURRENT_MEM_MB, PEAK_PARTITION_MEM/1048576
       AS PEAK_MEM_MB FROM TABLE (SYSPROC.ADMIN_GET_DBP_MEM_USAGE()) AS T
```

| DBPARTITIONNUM | MAX_MEM_MB | CURRENT_MEM_MB | PEAK_MEM_MB |
|----------------|------------|----------------|-------------|
| 0 | 4590 | 1107 | 1107 |
| 1 | 4590 | 1108 | 1108 |
| 2 | 4590 | 1106 | 1106 |

3 registro(s) seleccionado(s).

Función de tabla ADMIN_GET_INDEX_COMPRESS_INFO - Devolver información de índice de compresión

La función de tabla ADMIN_GET_INDEX_COMPRESS_INFO devuelve los potenciales ahorros de compresión de índice para índices no comprimidos o notifica las estadísticas de compresión de índice desde los catálogos.

Sintaxis

```
►►—ADMIN_GET_INDEX_COMPRESS_INFO—(—tipoobjeto—,—esquemaobjeto—,—nombreobjeto—,—númparticiónbd—,—idparticióndatos—)►►
```

El esquema es SYSPROC.

Parámetros de la función de tabla

tipoobjeto

Argumento de entrada de tipo VARCHAR(1) que indica el tipo de objeto. El valor debe ser uno de los siguientes valores sensibles a mayúsculas y minúsculas:

- 'T', NULL o la serie vacía para indicar una tabla
- 'I' para un índice

esquemaobjeto

Parámetro de entrada de tipo VARCHAR(128) sensible a mayúsculas y minúsculas que especifica el esquema de objeto.

Si *tipoobjeto* es 'T', NULL o la serie vacía ("), *esquemaobjeto* indica el esquema de tabla.

- Si *esquemaobjeto* está especificado y *nombreobjeto* es NULL o la serie vacía ("), se devuelve información para todos los índices de todas las tablas en el esquema especificado.
- Si se especifican tanto *esquemaobjeto* como *nombreobjeto*, se devuelve información para todos los índices de la tabla especificada.

Si *tipoobjeto* es 'I', *esquemaobjeto* indica el esquema de índice.

- Si *esquemaobjeto* está especificado y *nombreobjeto* es NULL o la serie vacía ("), se devuelve información para todos los índices en el esquema especificado.
- Si se especifican tanto *esquemaobjeto* como *nombreobjeto*, se devuelve información para el índice especificado.
- Si no se especifica ni *esquemaobjeto* ni *nombreobjeto*, se devuelve información para todos los índices de todos los esquemas.

Si se especifica *nombreobjeto* y no se especifica *esquemaobjeto*, la función devuelve un error de SQL. Se dice que un valor de parámetro no está especificado cuando tiene un valor NULL o el valor de la serie vacía (").

nombreobjeto

Parámetro de entrada de tipo VARCHAR(128) sensible a mayúsculas y minúsculas que especifica el nombre de objeto. Consulte la descripción del parámetro *esquemaobjeto*.

númparticiónbd

Parámetro de entrada de tipo INTEGER que especifica un número de partición de base de datos. Cuando se especifica, se devuelve información solamente para índices que residen en la partición de base de datos especificada. Para especificar que se deben devolver datos para todas las particiones de bases de datos activas, establezca el valor del parámetro *númparticiónbd* en -2 o bien NULL. En entornos de base de datos no particionada, especifique -2 o NULL.

idparticióndatos

Parámetro de entrada de tipo INTEGER que especifica el ID de partición de datos. Cuando se especifica, se devuelve información solamente para particiones de índice definidas en las particiones de datos especificadas. El ID de partición de datos debería corresponderse con el DATAPARTITIONID encontrado en la vista SYSCAT.DATAPARTITIONS. Para especificar que se deben devolver datos para todas las particiones de datos, establezca el valor del parámetro *idparticióndatos* en -2 o bien NULL. Para índices no particionados, especifique -2, 0 o NULL.

Autorización

Privilegio EXECUTE sobre la función de tabla
ADMIN_GET_INDEX_COMPRESS_INFO.

Ejemplo

Después de la migración de base de datos, todos los índices existentes están descomprimidos. Es probable que desee estimar los potenciales ahorros de compresión de índice para los índices existentes en la tabla "S.T1", que tiene un ID de partición de datos 3 y reside en el número de partición de base de datos 2. En este ejemplo, S es el nombre de esquema y T1 es el nombre de tabla, y T1 no está comprimida

```
SELECT compress_attr, iid, dbpartitionnum, index_compressed,
       pct_pages_saved, num_leaf_pages_saved
FROM TABLE(sysproc.admin_get_index_compress_info('S', 'T1', 2, 3)) AS t
```

En el siguiente ejemplo se muestra la salida de esta sentencia:

```
COMPRESS_ATTR      IID DBPARTITIONNUM INDEX_COMPRESSED ...
-----
N                  1          2 N ...
N                  2          2 N ...
```

```

... PCT_PAGES_SAVED NUM_LEAF_PAGES_SAVED
... -----
...          50          200
...          45          150

```

Es posible que decida que los ahorros de compresión merecen la pena y desee habilitar la compresión de índice.

```

ALTER INDEX INDEX1 compress yes
ALTER INDEX INDEX2 compress yes
REORG INDEXES all FOR table S.T1

```

A medida que pasa el tiempo, puede determinar la necesidad de crear índices nuevos para la tabla y estimar los ahorros de compresión de índice para estos índices antes de comprimirlos. También es posible que desee ver las estadísticas de compresión de los índices que ya se han comprimido.

```

SELECT compress_attr, iid, dbpartitionnum, index_compressed,
       pct_pages_saved, num_leaf_pages_saved
FROM TABLE(sysproc.admin_get_index_compress_info('', 'S', 'T1', 2, 3)) AS t

```

En el siguiente ejemplo se muestra la salida de esta sentencia:

```

COMPRESS_ATTR      IID DBPARTITIONNUM INDEX_COMPRESSED ...
-----
Y                  1          2 Y          ...
Y                  2          2 Y          ...
N                  3          2 N          ...
N                  4          2 N          ...
... PCT_PAGES_SAVED NUM_LEAF_PAGES_SAVED
... -----
...          -1          -1
...          -1          -1
...          58          230
...          49          140

```

Dado que los dos primeros índices ya estaban comprimidos, tal como se indica en la columna `index_compressed`, la sentencia devuelve valores tomados de los catálogos del sistema. En este caso, los valores de los catálogos no se han recopilado.

Después de ejecutar `RUNSTATS` en la tabla, la siguiente ejecución de la función de índice generará los resultados corregidos.

```

RUNSTATS ON TABLE S.T1 FOR INDEXES ALL
SELECT compress_attr, iid, dbpartitionnum, index_compressed,
       pct_pages_saved, num_leaf_pages_saved
FROM TABLE(sysproc.admin_get_index_compress_info('', 'S', 'T1', 2, 3)) AS t

```

En el siguiente ejemplo se muestra la salida de esta sentencia:

```

COMPRESS_ATTR      IID DBPARTITIONNUM INDEX_COMPRESSED ...
-----
Y                  1          2 Y          ...
Y                  2          2 Y          ...
N                  3          2 N          ...
N                  4          2 N          ...
... PCT_PAGES_SAVED NUM_LEAF_PAGES_SAVED
... -----
...          50          200
...          45          150
...          58          230
...          49          140

```

Metadatos de la función de tabla ADMIN_GET_INDEX_COMPRESS_INFO

Tabla 68. Metadatos de la función de tabla ADMIN_GET_INDEX_COMPRESS_INFO

| Nombre de columna | Tipo de datos | Descripción |
|----------------------|---------------|--|
| INDSCHEMA | VARCHAR(128) | Nombre del esquema en el que está definido el índice. |
| INDNAME | VARCHAR(128) | Nombre del índice. |
| TABSCHEMA | VARCHAR(128) | Nombre del esquema en el que está definida la tabla. |
| TABNAME | VARCHAR(128) | Nombre de tabla. |
| DBPARTITIONNUM | SMALLINT | Número de partición de base de datos. |
| IID | SMALLINT | Identificador del índice. |
| DATAPARTITIONID | INTEGER | ID de la partición de datos. |
| COMPRESS_ATTR | CHAR(1) | Estado del atributo COMPRESSION en el índice. <ul style="list-style-type: none"> • “Y” = La compresión de índice está habilitada • “N” = La compresión de índice no está habilitada |
| INDEX_COMPRESSED | CHAR(1) | Formato físico del índice. <ul style="list-style-type: none"> • “Y” = El índice está en formato comprimido • “N” = El índice está en formato no comprimido <p>Si el formato de índice físico no coincide con el atributo de compresión, es necesario reorganizar el índice para convertirlo al formato definido. Si la tabla o índice tiene un error en el momento en el que se ejecuta esta función, este valor será NULL.</p> |
| PCT_PAGES_SAVED | SMALLINT | Si el índice no está físicamente comprimido (INDEX_COMPRESSED es “N”), este valor representa el porcentaje estimado de páginas hoja guardadas como si el índice estuviera en realidad comprimido. Si el índice está físicamente comprimido (INDEX_COMPRESSED es “Y”), este valor notifica el valor PCTPAGESSAVED de la vista de catálogo del sistema (SYSCAT.INDEXES o bien SYSCAT.INDEXPARTITIONS). Nota: Este valor es el mismo para cada entrada de un índice o partición de índice para cada partición de base de datos en un entorno DPF. Si la tabla o índice tiene un error en el momento en el que se ejecuta esta función, este valor será NULL. |
| NUM_LEAF_PAGES_SAVED | BIGINT | Si el índice no está físicamente comprimido (INDEX_COMPRESSED es “N”), este valor representa el número estimado de páginas hoja guardadas como si el índice estuviera en realidad comprimido. Si el índice está físicamente comprimido (INDEX_COMPRESSED es “Y”), este valor notifica el número calculado de páginas hoja guardadas en función de los valores PCTPAGESSAVED y NLEAF de la vista de catálogo del sistema (SYSCAT.INDEXES o bien SYSCAT.INDEXPARTITIONS). Si ni PCTPAGESSAVED ni NLEAF son valores inválidos (-1), este valor también se establece en -1. Nota: Este valor es el mismo para cada entrada de un índice o partición de índice para cada partición de base de datos en un entorno DPF. Si la tabla o índice tiene un error en el momento en el que se ejecuta esta función, este valor será NULL. |

Función de tabla ADMIN_GET_INDEX_INFO - Devolver información de índice

La función de tabla ADMIN_GET_INDEX_INFO devuelve información de índice que no está disponible en las vistas de catálogo, por ejemplo información de compresión y los tamaños lógicos y físicos del índice.

Sintaxis

►►—ADMIN_GET_INDEX_INFO—(—*tipoobjeto*—,—*esquemaobjeto*—,—*nombreobjeto*—)——►►

El esquema es SYSPROC.

Parámetros de la función de tabla

tipoobjeto

Argumento de entrada de tipo VARCHAR(1) que indica el tipo de objeto. El valor debe ser uno de los siguientes valores sensibles a mayúsculas y minúsculas:

- 'T', NULL o la serie vacía (") para indicar una tabla
- 'I' para un índice

esquemaobjeto

Parámetro de entrada de tipo VARCHAR(128) sensible a mayúsculas y minúsculas que especifica el esquema de objeto.

Si *tipoobjeto* es 'T', NULL o la serie vacía ("), *esquemaobjeto* indica el esquema de tabla.

- Si *esquemaobjeto* está especificado y *nombreobjeto* es NULL o la serie vacía ("), se devuelve información para todos los índices de todas las tablas en el esquema especificado.
- Si se especifican tanto *esquemaobjeto* como *nombreobjeto*, se devuelve información para todos los índices de la tabla especificada.

Si *tipoobjeto* es 'I', *esquemaobjeto* indica el esquema de índice.

- Si *esquemaobjeto* está especificado y *nombreobjeto* es NULL o la serie vacía ("), se devuelve información para todos los índices en el esquema especificado.
- Si se especifican tanto *esquemaobjeto* como *nombreobjeto*, se devuelve información para el índice especificado.
- Si no se especifica ni *esquemaobjeto* ni *nombreobjeto*, se devuelve información para todos los índices de todos los esquemas.

Si se especifica *nombreobjeto* y no se especifica *esquemaobjeto*, la función devuelve un error de SQL. Se dice que un valor de parámetro no está especificado cuando tiene un valor NULL o el valor de la serie vacía (").

nombreobjeto

Parámetro de entrada de tipo VARCHAR(128) sensible a mayúsculas y minúsculas que especifica el nombre de objeto. Consulte la descripción del parámetro *esquemaobjeto*.

Autorización

Privilegio EXECUTE sobre la función de tabla ADMIN_GET_INDEX_INFO.

Ejemplo

Después de habilitar la compresión de índice para varios índices de una tabla, se desea determinar qué índices están comprimidos y qué índices requieren una reconstrucción para poder comprimirse. En este ejemplo, S es el nombre de esquema y T1 es el nombre de tabla.

```
db2 SELECT iid, compress_attr, index_compressed
      FROM TABLE(sysproc.admin_get_index_info('', 'S', 'T1')) AS t
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| IID | COMPRESS_ATTR | INDEX_COMPRESSED |
|-----|---------------|------------------|
| 1 | Y | Y |
| 2 | Y | Y |
| 3 | Y | N |
| 4 | N | N |

Además, se desea consultar otra información de índice para todos los índices en el esquema S2. En este ejemplo:

- T2 = una tabla particionada con dos particiones de datos
- T3 = una tabla no particionada
- IND_1 = un índice no particionado en T2
- IND_2 = un índice particionado en T2
- IND_3 = un índice particionado en T2
- IND_4 = un índice en T3
- IND_5 = un índice en T3

```
db2 SELECT tabname, indname, iid, index_partitioning, datapartitionid,
      index_object_l_size, index_object_p_size, index_requires_rebuild,
      large_rids, FROM TABLE(sysproc.admin_get_index_info('I', 'S2', '')) AS t
```

El ejemplo siguiente muestra la salida de la consulta.

| TABNAME | INDNAME | IID | INDEX_PARTITIONING | DATAPARTITIONID |
|---------|---------|-----|--------------------|-----------------|
| T2 | IND_1 | 1 | N | 0 |
| T2 | IND_2 | 2 | P | 1 |
| T2 | IND_2 | 2 | P | 2 |
| T2 | IND_3 | 3 | P | 1 |
| T2 | IND_3 | 3 | P | 2 |
| T3 | IND_4 | 4 | | 0 |
| T3 | IND_5 | 5 | | 0 |

Datos de salida de este procedimiento (continuación):

| INDEX_OBJECT_L_SIZE | INDEX_OBJECT_P_SIZE | INDEX_REQUIRES_REBUILD | LARGE_RIDS |
|---------------------|---------------------|------------------------|------------|
| 50 | 51 | N | Y |
| 40 | 40 | N | Y |
| 45 | 45 | N | Y |
| 40 | 40 | N | Y |
| 45 | 45 | N | Y |
| 20 | 20 | N | Y |
| 20 | 20 | N | Y |

Metadatos de la función de tabla ADMIN_GET_INDEX_INFO

Tabla 69. Metadatos de la función de tabla ADMIN_GET_INDEX_INFO

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| INDSCHEMA | VARCHAR(128) | Nombre del esquema en el que está definido el índice. |

Tabla 69. Metadatos de la función de tabla ADMIN_GET_INDEX_INFO (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|---------------------|---------------|---|
| INDNAME | VARCHAR(128) | Nombre del índice. |
| TABSCHEMA | VARCHAR(128) | Nombre del esquema en el que está definida la tabla. |
| TABNAME | VARCHAR(128) | Nombre de tabla. |
| DBPARTITIONNUM | SMALLINT | Número de partición de base de datos. |
| IID | SMALLINT | Identificador del índice. |
| DATAPARTITIONID | INTEGER | ID de la partición de datos. |
| COMPRESS_ATTR | CHAR(1) | Estado del atributo COMPRESSION en el índice. <ul style="list-style-type: none"> • “Y” = La compresión de índice está habilitada • “N” = La compresión de índice no está habilitada |
| INDEX_COMPRESSED | CHAR(1) | Formato físico del índice. <ul style="list-style-type: none"> • “Y” = El índice está en formato comprimido • “N” = El índice está en formato no comprimido <p>Si el formato físico del índice no coincide con el atributo de compresión, es necesario reorganizar el índice para convertirlo al formato definido. Si la tabla o índice tiene un error cuando se ejecuta esta función, este valor será NULL.</p> |
| INDEX_PARTITIONING | CHAR(1) | Identifica la característica de particionamiento del índice. <ul style="list-style-type: none"> • “N” = Índice no particionado • “P” = Índice particionado • Blanco = el índice no está en una tabla particionada |
| INDEX_OBJECT_L_SIZE | BIGINT | Tamaño lógico del objeto de índice. Para tablas no particionadas, se trata de la cantidad de espacio de disco que se ha asignado lógicamente para todos los índices definidos en la tabla. Para un índice no particionado en una tabla no particionada, es la cantidad de espacio de disco que se ha asignado lógicamente para el índice. Para un índice particionado en una tabla particionada, es la cantidad de espacio de disco que se ha asignado lógicamente para todas las particiones de índice definidas en la partición de datos. Todos los tamaños se indican en kilobytes (KB). <p>El tamaño lógico es la cantidad de espacio que la tabla o partición de datos reconoce. Puede ser inferior a la cantidad de espacio que se ha asignado físicamente para contener los datos de índice para la tabla o partición de datos (por ejemplo, en el caso de un truncamiento de tabla lógica). El tamaño devuelto tiene en cuenta las extensiones completas que se han asignado lógicamente para los índices y, para los índices que se han creado en espacios de tablas DMS, incluye una estimación de las extensiones EMP. Si la tabla o índice tiene un error cuando se ejecuta esta función, este valor será NULL.</p> |

Tabla 69. Metadatos de la función de tabla ADMIN_GET_INDEX_INFO (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|------------------------|---------------|---|
| INDEX_OBJECT_P_SIZE | BIGINT | <p>Tamaño físico del objeto de índice. Para tablas no particionadas, se trata de la cantidad de espacio de disco que se ha asignado físicamente para todos los índices definidos en la tabla. Para un índice no particionado en una tabla no particionada, es la cantidad de espacio de disco que se ha asignado físicamente para el índice. Para un índice particionado en una tabla particionada, es la cantidad de espacio de disco que se ha asignado físicamente para todas las particiones de índice definidas en la partición de datos. Todos los tamaños se indican en kilobytes (KB).</p> <p>El tamaño devuelto tiene en cuenta las extensiones completas que se han asignado para los índices e incluye las extensiones EMP para los índices que se han creado en los espacios de tablas DMS. Si la tabla o índice tiene un error cuando se ejecuta esta función, este valor será NULL.</p> |
| INDEX_REQUIRES_REBUILD | CHAR(1) | <p>Estado de recreación para el índice.</p> <ul style="list-style-type: none"> • “Y” si el índice definido en la tabla o partición de datos requiere volver a crearse • “N” en caso contrario <p>Si la tabla tiene un error cuando se ejecuta esta función, este valor será NULL.</p> |
| LARGE_RIDS | CHAR(1) | <p>Indica si el índice utiliza o no los ID de fila grandes (RID) (número de página de 4 bytes, número de ranura de 2 bytes).</p> <ul style="list-style-type: none"> • “Y” indica que el índice utiliza RID grandes • “N” indica que el índice no utiliza RID grandes • “P” (pendiente) indica que la tabla en la que está definido el índice soporta RID grandes (es decir, la tabla se encuentra en un espacio de tablas grande), pero el índice de la tabla o partición de datos todavía no se ha reorganizado o recreado. Por consiguiente, la tabla continúa utilizando los RID de 4 bytes, y deben emprenderse las acciones necesarias para convertir la tabla o el índice a RID grandes. <p>Si la tabla tiene un error cuando se ejecuta esta función, este valor será NULL.</p> |

Función de tabla ADMIN_GET_MSGS - Recuperar mensajes generados por un programa de utilidad de movimiento de datos que utiliza el procedimiento ADMIN_CMD

La función de tabla ADMIN_GET_MSGS se utiliza para recuperar los mensajes generados por una sola ejecución de un mandato de un programa de utilidad de movimiento de datos mediante el procedimiento ADMIN_CMD. El parámetro de entrada *id_operación* identifica la operación.

Sintaxis

►►—ADMIN_GET_MSGS—(—*id_operación*—)—————►►

El esquema es SYSPROC.

Parámetro de la función de tabla

id_operación

Argumento de entrada de tipo VARCHAR(139) que especifica el ID de operación del archivo o archivos de mensajes generados por un programa de utilidad de movimiento de datos que se ha ejecutado mediante el procedimiento ADMIN_CMD. El ID de operación lo genera el procedimiento ADMIN_CMD.

Autorización

Privilegio EXECUTE sobre la función de tabla ADMIN_GET_MSGS. El ID de usuario delimitado debe tener acceso de lectura sobre los archivos que hay bajo el directorio indicado por la variable de registro DB2_UTIL_MSGPATH. Si la variable de registro no está establecida, el ID de usuario delimitado debe tener acceso de lectura sobre los archivos del subdirectorío tmp del directorio de la instancia.

Ejemplo

Comprobar todos los mensajes que devuelve el programa de utilidad EXPORT que se han ejecutado mediante el procedimiento ADMIN_CMD, con el ID de operación '24523_THERESAX'

```
SELECT * FROM TABLE(SYSPROC.ADMIN_GET_MSGS('24523_THERESAX')) AS MSG
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
DBPARTITIONNUM AGENTTYPE SQLCODE  MSG
-----
-                -          SQL3104N El programa de utilidad de exportación
                    está iniciando la exportación de los
                    datos en el archivo
                    "/home/theresax/rtest/data/ac_load03.de1".
-                -          SQL3105N El programa de utilidad de exportación
                    ha finalizado la exportación de "8"
                    filas.
```

2 registro(s) seleccionado(s).

Notas de uso

La sentencia de la consulta que invoca esta función de tabla con el *id_operación* adecuado se encuentra en la columna MSG_RETRIEVAL del primer conjunto de resultados que devuelve el procedimiento ADMIN_CMD.

Información devuelta

Tabla 70. Información devuelta por la función de tabla ADMIN_GET_MSGS

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| DBPARTITIONNUM | INTEGER | Número de partición de base de datos. Este valor sólo se devuelve para una carga distribuida e indica para qué partición de base de datos es el mensaje correspondiente. |

Tabla 70. Información devuelta por la función de tabla ADMIN_GET_MSGS (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| AGENTTYPE | CHAR(4) | Tipo de agente. Este valor sólo se devuelve para una carga distribuida. Los valores posibles son: <ul style="list-style-type: none"> • 'LOAD': para agente de carga • 'PART': para agente de particionamiento • 'PREP': para agente de preparticionamiento • Nulo: no hay información de tipo de agente disponible |
| SQLCODE | VARCHAR(9) | SQLCODE del mensaje que se devuelve. |
| MSG | VARCHAR(1024) | Mensaje de error abreviado correspondiente al SQLCODE. |

Función ADMIN_IS_INLINED - Determinar si los datos están en línea

La función ADMIN_IS_INLINED recupera la información de estado de los datos en línea correspondiente a una columna XML, BLOB, CLOB o DBCLOB.

Sintaxis

►—ADMIN_IS_INLINED—(—nombre-columna—)—————►

El esquema es SYSIBM.

Valor de retorno

Esta función devuelve uno de los valores siguientes de tipo SMALLINT o el valor NULL:

- 1 Indica que los datos están en línea.
- 0 Indica que los datos no están en línea.
- NULL Indica que las entradas son NULL.

Parámetros de la función

nombre-columna

Identifica una columna de la tabla base con el tipo de datos XML, BLOB, CLOB o DBCLOB (SQLSTATE 42884). La columna debe hacer referencia, de forma directa o indirecta, a la columna de una tabla base que no se ha generado en función de una expresión (SQLSTATE 42815).

Ejemplo

Ejemplo 1: En el ejemplo siguiente se indica si los tres documentos XML de la columna XML xml_doc1 de la tabla TAB1 están en línea:

```
db2 => SELECT PK, ADMIN_IS_INLINED(xml_doc1) as IS_INLINED
       from TAB1
```

Esta consulta genera la salida siguiente:

| PK | IS_INLINED |
|----|------------|
| 1 | 1 |
| 2 | 0 |
| 3 | 0 |

3 registro(s) seleccionado(s).

Ejemplo 2: En el ejemplo siguiente se indica si uno de los documentos XML de la columna XML xml_doc1 de la tabla TAB1 está en línea:

```
db2 => SELECT PK, ADMIN_IS_INLINED(xml_doc1) as IS_INLINED
       from TAB1 where PK=1
```

Esta consulta genera la salida siguiente:

| PK | IS_INLINED |
|----|------------|
| 1 | 1 |

1 registro(s) seleccionado(s).

Ejemplo 3: En el ejemplo siguiente se indica si los tres datos CLOB contenidos en la columna CLOB clob_1 de la tabla TAB1 están en línea:

```
db2 => SELECT PK, ADMIN_IS_INLINED(clob_1) as IS_INLINED
       from TAB1
```

Esta consulta genera la salida siguiente:

| PK | IS_INLINED |
|----|------------|
| 1 | 0 |
| 2 | 0 |
| 3 | 1 |

3 registro(s) seleccionado(s).

Procedimiento ADMIN_REMOVE_MSGS - Limpiar los mensaje que genera un programa de utilidad de movimiento de datos que utiliza el procedimiento ADMIN_CMD

El procedimiento ADMIN_REMOVE_MSGS se utiliza para limpiar los mensajes generados por una sola ejecución de un mandato de un programa de utilidad de movimiento de datos mediante el procedimiento ADMIN_CMD. El parámetro de entrada *id_operación* identifica la operación.

Sintaxis

►► ADMIN_REMOVE_MSGS (—*id_operación*—) ◀◀

El esquema es SYSPROC.

Parámetro del procedimiento

id_operación

Argumento de entrada de tipo VARCHAR(139) que especifica el ID de operación del archivo o archivos de mensajes generados por un programa de utilidad de movimiento de datos que se ha ejecutado mediante el procedimiento ADMIN_CMD. El ID de operación lo genera el procedimiento ADMIN_CMD.

Autorización

Privilegio EXECUTE sobre el procedimiento ADMIN_REMOVE_MSGS. El ID de usuario delimitado debe ser capaz de eliminar los archivos que hay bajo el directorio indicado por la variable de registro DB2_UTIL_MSGPATH. Si la variable de registro no está establecida, el ID de usuario delimitado debe poder suprimir los archivos del subdirectorio tmp del directorio de la instancia.

Ejemplo

Limpiar los mensajes con el ID de operación '24523_THERESAX'.

```
CALL SYSPROC.ADMIN_REMOVE_MSGS('24523_THERESAX')
```

Notas de uso

La sentencia CALL que invoca este procedimiento con el *id_operación* adecuado se encuentra en la columna MSG_REMOVAL del primer conjunto de resultados que devuelve el procedimiento ADMIN_CMD.

Procedimiento ADMIN_REVALIDATE_DB_OBJECTS - Revalidar objetos de base de datos no válidos

El procedimiento ADMIN_REVALIDATE_DB_OBJECTS revalida los objetos de base de datos no válidos.

Este procedimiento toma tres parámetros de entrada, *tipo_objeto*, *esquema_objeto* y *nombre_objeto*, que controlan el nivel de revalidación que se va a ejecutar:

- Para revalidar todos los objetos no válidos de la base de datos, especifique NULL en todos los parámetros o llame al procedimiento sin los parámetros.
- Para revalidar todos los objetos de base de datos no válidos en un esquema específico, especifique un valor para *esquema_objeto* y especifique NULL para *nombre_objeto* y *tipo_objeto*.
- Para revalidar un objeto de base de datos no válido concreto, especifique valores válidos en todos los parámetros.

Sintaxis

```
►►—ADMIN_REVALIDATE_DB_OBJECTS—(—tipo_objeto—,—esquema_objeto—,——————►  
►—nombre_objeto—)—————►◄
```

El esquema es SYSPROC.

Parámetros del procedimiento

tipo_objeto

Parámetro de entrada de tipo VARCHAR(30) que identifica el tipo del objeto de base de datos. Son válidos los siguientes:

- FUNCTION
- GLOBAL VARIABLE
- METHOD
- MODULE
- PROCEDURE
- SPECIFIC
- TABLE
- TRIGGER
- TYPE
- VIEW

Este valor no es sensible a mayúsculas y minúsculas. Este valor puede ser NULL.

Si se especifica alguno de estos tipos, el procedimiento revalida todos los objetos no válidos de dicho tipo, salvo aquellos que pertenecen a un MODULE. Si desea revalidar todos los objetos que se encuentran dentro de un módulo, utilice el tipo MODULE con el nombre de un módulo específico, y se revalidarán todos los objetos situados dentro de ese módulo.

Si existe una rutina que tiene más de una signatura de parámetro y desea revalidar una de ellas solamente, utilice el tipo SPECIFIC con el nombre de la rutina que desea revalidar.

Si emplea el tipo TABLE, las tablas especificadas se reorganizarán y se recopilarán sus estadísticas. El procedimiento invoca al programa de utilidad reorg, seguido del programa de utilidad runstats, contra cualquier tabla de consultas materializadas o regulares que se encuentra en estado Pendiente de reorganización. El procedimiento intentará utilizar un perfil de usuario para runstats, si existe. En caso contrario, se invoca una operación runstats por omisión.

esquema_objeto

Argumento de entrada de tipo VARCHAR(128) que identifica el nombre de esquema utilizado para calificar referencias de objeto de base de datos. El nombre es sensible a mayúsculas y minúsculas. Este valor puede ser NULL.

nombre_objeto

Argumento de entrada de tipo VARCHAR(128) que identifica un objeto de base de datos. El nombre es sensible a mayúsculas y minúsculas. Este valor no puede ser el valor de una función de fila o tabla con tipo, ya que el procedimiento no da soporte a estos tipos de objeto; si se especifica el nombre de un objeto así, se devuelve un error. Este valor puede ser NULL.

Autorización

Privilegio EXECUTE sobre el procedimiento ADMIN_REVALIDATE_DB_OBJECTS.

Ejemplos

Ejemplo 1: Revalidar todo lo contenido en la base de datos actual.

```
CALL SYSPROC.ADMIN_REVALIDATE_DB_OBJECTS(NULL, NULL, NULL)
```


O, alternativamente, llamar al procedimiento sin parámetros.

```
CALL SYSPROC.ADMIN_REVALIDATE_DB_OBJECTS()
```

Ejemplo 2: Revalidar todos los objetos calificados por el esquema MY_SCHEMA.

```
CALL SYSPROC.ADMIN_REVALIDATE_DB_OBJECTS(NULL, 'MY_SCHEMA', NULL)
```

Ejemplo 3: Revalidar todos los objetos de activador de la base de datos.

```
CALL SYSPROC.ADMIN_REVALIDATE_DB_OBJECTS('trigger', NULL, NULL)
```

Ejemplo 4: Revalidar un objeto de vista específico.

```
CALL SYSPROC.ADMIN_REVALIDATE_DB_OBJECTS('view', 'MY_SCHEMA', 'MY_VIEW')
```

Ejemplo 5: Revalidar todos los procedimientos de MY_SCHEMA. En este ejemplo, existen tres procedimientos (proc1, proc2 y proc3) bajo este esquema. El objeto referenciado utilizado por proc1 no existe. La llamada siguiente revalida proc2 y proc3, pero proc1 continúa siendo no válido. En esta situación, la llamada devuelve un aviso.

```
CALL SYSPROC.ADMIN_REVALIDATE_DB_OBJECTS('procedure', 'MY_SCHEMA', NULL)
```

Ejemplo 6: Revalidar un objeto que no existe. Este ejemplo devuelve un error.

```
CALL SYSPROC.ADMIN_REVALIDATE_DB_OBJECTS('procedure', 'MY_SCHEMA', 'MY_VIEW')
```

Ejemplo 7: Revalidar todos los procedimientos de MY_SCHEMA con la notación de Oracle.

```
CALL SYSPROC.ADMIN_REVALIDATE_DB_OBJECTS(  
  object_type=>'PROCEDURE',object_schema=>'MY_SCHEMA')
```

Notas de uso

Se deben indicar todos los valores de parámetro que no son nulos que se proporcionan al procedimiento ADMIN_REVALIDATE_DB_OBJECTS, o el procedimiento no puede identificar los objetos que se deben revalidar. Por ejemplo, si especifica un nombre de vista pero le especifica un tipo de activador, el procedimiento no revalidará la vista ya que el tipo no coincide.

Este procedimiento revalidará únicamente los objetos no válidos y las tablas de consultas materializadas o regulares con estado Pendiente de reorganización. Todos los objetos no válidos pueden encontrarse en SYSCAT.INVALIDOBJECTS. Para determinar qué tablas se encuentran en estado Pendiente de reorganización, utilice la función de tabla ADMIN_GET_TAB_INFO.

Si se especifica un objeto válido como entrada, el procedimiento no realizará ninguna operación y devolverá un código satisfactorio. Si se produce una anomalía durante la revalidación de las tablas, el procedimiento falla. Si se produce una anomalía durante la revalidación de otros objetos, el procedimiento omite el error y continúa revalidando el resto de los objetos. Si se produce al menos un error, el procedimiento devuelve un aviso (SQLSTATE 0168B). Si falla la revalidación de todos los objetos, el procedimiento devuelve un error (SQLSTATE 429C4). Todos los detalles de los errores de revalidación de objetos, excepto tablas, se encuentran en SYSCAT.INVALIDOBJECTS.

Cuando se revalida una variable global, también se instancia para la sesión actual.

Para supervisar el progreso de una revalidación de tabla, puede supervisar el progreso de la operación de reorganización de tabla asociada. Para el resto de

objetos, emita una consulta a la vista de catálogo SYSCAT.INVALIDOBJECTS; los objetos se suprimen de esta vista cuando su revalidación se completa con éxito y las entradas se actualizan si la revalidación falla.

Vista administrativa ADMINTABCOMPRESSINFO y función de tabla ADMIN_GET_TAB_COMPRESS_INFO_V97 - Devolver información de compresión

La vista administrativa ADMINTABCOMPRESSINFO y la función de tabla ADMIN_GET_TAB_COMPRESS_INFO_V97 devuelven información de compresión para tablas, tablas de consulta materializada (MQT) y tablas de jerarquía.

Vista administrativa ADMINTABCOMPRESSINFO

La vista administrativa ADMINTABCOMPRESSINFO devuelve información compresión para tablas, tablas de consultas materializadas (MQT) y tablas de jerarquía. En la vista de catálogo SYSCAT.TABLES, estos tipos de tablas se informan con T para tabla, S para tablas de consultas materializadas y H para tablas de jerarquía. Para una tabla, la información se devuelve en el nivel de partición de datos y en el nivel de partición de base de datos.

El esquema es SYSIBMADM.

En la tabla Vista administrativa ADMINTABCOMPRESSINFO y metadatos de la función de tabla ADMIN_GET_TAB_COMPRESS_INFO_V97 encontrará una lista completa de la información que puede devolverse.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa ADMINTABCOMPRESSINFO
- Privilegio CONTROL en la vista administrativa ADMINTABCOMPRESSINFO
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función ADMIN_GET_TAB_COMPRESS_INFOV97
- Autorización DATAACCESS

Ejemplos

Ejemplo 1: Recuperar toda la información de compresión para todas las tablas

```
SELECT * FROM SYSIBMADM.ADMINTABCOMPRESSINFO
```

El ejemplo siguiente muestra los datos de salida de la consulta:

| TABSCHEMA | TABNAME | DBPARTITIONNUM | DATA_PARTITION_ID | COMPRESS_ATTR | DICT_BUILDER | ... |
|-----------|-----------|----------------|-------------------|---------------|--------------|-----|
| SYSIBM | SYSTABLES | 0 | 0 | N | NOT BUILT | ... |
| SYSIBM | SYSTABLES | 0 | 0 | N | NOT BUILT | ... |
| ... | | | | | | |
| SIMAP2 | STAFF | 0 | 4 | Y | REORG | ... |
| SIMAP2 | STAFF | 0 | 4 | Y | REORG | ... |
| ... | | | | | | |

156 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación):

| DICT_BUILD_TIMESTAMP | COMPRESS_DICT_SIZE | EXPAND_DICT_SIZE | ROWS_SAMPLED | ... |
|----------------------------|--------------------|------------------|--------------|-----|
| - | 0 | 0 | 0 | ... |
| - | 0 | 0 | 0 | ... |
| ... | | | | |
| 2009-03-31-11.08.18.000000 | 3968 | 3000 | 6 | ... |
| 2009-03-31-11.08.18.000000 | 13312 | 10944 | 6 | ... |
| ... | | | | |

Datos de salida de esta consulta (continuación):

| PAGES_SAVED_PERCENT | BYTES_SAVED_PERCENT | AVG_COMPRESS_REC_LENGTH | OBJECT_TYPE |
|---------------------|---------------------|-------------------------|-------------|
| 0 | 0 | 0 | DATA |
| 0 | 0 | 0 | XML |
| ... | | | |
| 70 | 70 | 31 | DATA |
| 66 | 66 | 235 | XML |
| ... | | | |

Ejemplo 2: Determinar el objeto en el que se creó el diccionario, la acción de creación del diccionario y la hora de creación del diccionario para todas las tablas.

```
SELECT TABSCHEMA, TABNAME, DBPARTITIONNUM, DATA_PARTITION_ID,
       OBJECT_TYPE, DICT_BUILDER, DICT_BUILD_TIMESTAMP
FROM SYSIBMADM.ADMINTABCOMPRESSINFO
```

El ejemplo siguiente muestra los datos de salida de la consulta:

| TABSCHEMA | TABNAME | DBPARTITIONNUM | DATA_PARTITION_ID | ... |
|-----------|-----------------|----------------|-------------------|-----|
| SYSIBM | SYSTABLES | 0 | 0 | ... |
| SYSIBM | SYSTABLES | 0 | 0 | ... |
| ... | | | | |
| SIMAP2 | STAFF | 0 | 4 | ... |
| SIMAP2 | STAFF | 0 | 4 | ... |
| SYSTOOLS | HMON_COLLECTION | 0 | 0 | ... |
| SYSTOOLS | HMON_COLLECTION | 0 | 0 | ... |

156 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación):

| OBJECT_TYPE | DICT_BUILDER | DICT_BUILD_TIMESTAMP |
|-------------|--------------|----------------------------|
| DATA | NOT BUILT | - |
| XML | NOT BUILT | - |
| ... | | |
| DATA | REORG | 2009-03-31-11.08.18.000000 |
| XML | REORG | 2009-03-31-11.08.18.000000 |
| DATA | REDISTRIBUTE | 2009-03-29-06.44.32.000000 |
| XML | REDISTRIBUTE | 2009-03-29-06.44.32.000000 |

Función de tabla ADMIN_GET_TAB_COMPRESS_INFO_V97

La función de tabla ADMIN_GET_TAB_COMPRESS_INFO_V97 devuelve la misma información que la vista administrativa ADMINTABCOMPRESSINFO, pero le permite especificar un esquema, un nombre de tabla y una modalidad de ejecución.

En la tabla Vista administrativa ADMINTABCOMPRESSINFO y metadatos de la función de tabla ADMIN_GET_TAB_COMPRESS_INFO_V97 encontrará una lista completa de la información que puede devolverse.

Esta función devuelve dos filas para cada tabla. Una fila tiene un valor de DATA en la columna OBJECT_TYPE, y la otra fila tiene un valor de XML para esa columna. La fila marcada como DATA es equivalente al valor de retorno de la función de tabla "Vista ADMIN_TABCOMPRESSINFO y ADMIN_GET_TAB_COMPRESS_INFO" en la página 1001 en desuso. La fila marcada como XML describe el diccionario de compresión XML.

Sintaxis

```
►►—ADMIN_GET_TAB_COMPRESS_INFO_V97—(—esquematabla—,—nombretabla—,—modejec—)————►◄
```

El esquema es SYSPROC.

Parámetros de la función de tabla

esquematabla

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de esquema.

nombretabla

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de tabla, un nombre de tabla de consultas materializadas o un nombre de tabla de jerarquía.

modejec

Argumento de entrada de tipo VARCHAR(30) que especifica la modalidad de ejecución. La modalidad de ejecución puede ser una de las siguientes:

- 'REPORT' -- Notifica la información de compresión de última generación. Es el valor por omisión.
- 'ESTIMATE' -- Genera la nueva información de compresión a partir de la tabla actual.

Autorización

Privilegio EXECUTE sobre la función ADMIN_GET_TAB_COMPRESS_INFO_V97.

Ejemplos

Ejemplo 1: Recuperar la información de compresión existente para la tabla SIMAP2.STAFF

```
SELECT *
FROM TABLE(
    SYSPROC.ADMIN_GET_TAB_COMPRESS_INFO_V97('SIMAP2', 'STAFF', 'REPORT'))
AS T
```

El siguiente ejemplo muestra los datos de salida de esta consulta:

| TABSHEMA | TABNAME | DBPARTITIONNUM | DATA_PARTITION_ID | COMPRESS_ATTR | DICT_BUILDER | ... |
|----------|---------|----------------|-------------------|---------------|--------------|-----|
| SIMAP2 | STAFF | 0 | 4 | Y | REORG | ... |
| SIMAP2 | STAFF | 0 | 4 | Y | NOT BUILT | ... |

2 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación):

| DICT_BUILD_TIMESTAMP | COMPRESS_DICT_SIZE | EXPAND_DICT_SIZE | ROWS_SAMPLED | ... |
|----------------------------|--------------------|------------------|--------------|-----|
| 2009-03-31-12.19.30.000000 | 13312 | 5296 | 35 | ... |
| - | 0 | 0 | 0 | ... |

Datos de salida de esta consulta (continuación):

| PAGES_SAVED_PERCENT | BYTES_SAVED_PERCENT | AVG_COMPRESS_REC_LENGTH | OBJECT_TYPE |
|---------------------|---------------------|-------------------------|-------------|
| 38 | 38 | 27 | DATA |
| 0 | 0 | 0 | XML |

Ejemplo 2: Recuperar la información de compresión estimada para la tabla SIMAP2.STAFF desde ya.

```
SELECT *
FROM TABLE(
  SYSPROC.ADMIN_GET_TAB_COMPRESS_INFO_V97('SIMAP2', 'STAFF', 'ESTIMATE'))
AS T
```

El siguiente ejemplo muestra los datos de salida de esta consulta:

| TABSCHEMA | TABNAME | DBPARTITIONNUM | DATA_PARTITION_ID | COMPRESS_ATTR | DICT_BUILDER | ... |
|-----------|---------|----------------|-------------------|---------------|----------------|-----|
| SIMAP2 | STAFF | 0 | 4 | Y | TABLE FUNCTION | ... |
| SIMAP2 | STAFF | 0 | 4 | Y | TABLE FUNCTION | ... |

2 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación):

| DICT_BUILD_TIMESTAMP | COMPRESS_DICT_SIZE | EXPAND_DICT_SIZE | ROWS_SAMPLED | ... |
|----------------------------|--------------------|------------------|--------------|-----|
| 2009-03-31-12.27.06.000000 | 13312 | 5296 | 35 | ... |
| 2009-03-31-12.27.06.000000 | 13312 | 9544 | 8 | ... |

Datos de salida de esta consulta (continuación):

| PAGES_SAVED_PERCENT | BYTES_SAVED_PERCENT | AVG_COMPRESS_REC_LENGTH | OBJECT_TYPE |
|---------------------|---------------------|-------------------------|-------------|
| 38 | 38 | 27 | DATA |
| 75 | 75 | 95 | XML |

Ejemplo 3: Determinar el tamaño total de diccionario para todos los objetos del esquema SIMAP2

```
SELECT TABSCHEMA, TABNAME, OBJECT_TYPE, DICT_BUILDER, (
  COMPRESS_DICT_SIZE+EXPAND_DICT_SIZE)
AS TOTAL_DICT_SIZE, DBPARTITIONNUM, DATA_PARTITION_ID
FROM TABLE(
  SYSPROC.ADMIN_GET_TAB_COMPRESS_INFO_V97('SIMAP2', '', 'REPORT'))
AS T
```

Datos de salida de esta consulta:

| TABSCHEMA | TABNAME | OBJECT_TYPE | DICT_BUILDER | ... |
|-----------|------------|-------------|--------------|-----|
| SIMAP2 | ACT | DATA | NOT BUILT | ... |
| SIMAP2 | ACT | XML | NOT BUILT | ... |
| SIMAP2 | ADEFUSR | DATA | INSPECT | ... |
| SIMAP2 | ADEFUSR | XML | NOT BUILT | ... |
| ... | | | | |
| SIMAP2 | CUSTOMER | DATA | REORG | ... |
| SIMAP2 | CUSTOMER | XML | REORG | ... |
| SIMAP2 | DEPARTMENT | DATA | NOT BUILT | ... |
| SIMAP2 | DEPARTMENT | XML | NOT BUILT | ... |
| ... | | | | |
| SIMAP2 | STAFF | DATA | REORG | ... |

```

SIMAP2      STAFF      XML      NOT BUILT ...
SIMAP2      SUPPLIERS  DATA    TABLE GROWTH ...
SIMAP2      SUPPLIERS  XML      NOT BUILT ...

```

44 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación):

```

TOTAL_DICT_SIZE DBPARTITIONNUM DATA_PARTITION_ID
-----
          0          0          0
          0          0          0
        1890          0          0
          0          0          0
...
        6968          0          1
       24256          0          1
          0          1          0
          0          1          0
...
       18608          0          4
          0          0          4
        6960          0          2
          0          0          2

```

Ejemplo 4: Ver un informe de la información de diccionario de las tablas en el esquema SIMAP2.

```

SELECT * FROM TABLE (
  SYSPROC.ADMIN_GET_TAB_COMPRESS_INFO_V97('SIMAP2', '', 'REPORT'))
AS T

```

Datos de salida de esta consulta:

```

TABSCHEMA TABNAME  DBPARTITIONNUM DATA_PARTITION_ID COMPRESS_ATTR DICT_BUILDER ...
-----
SIMAP2     ACT             0                0 N             NOT BUILT ...
SIMAP2     ACT             0                0 N             NOT BUILT ...
SIMAP2     ADEFUSR        0                0 N             INSPECT ...
SIMAP2     ADEFUSR        0                0 N             NOT BUILT ...
...
SIMAP2     CUSTOMER       0                1 Y             REORG ...
SIMAP2     CUSTOMER       0                1 Y             REORG ...
...
SIMAP2     STAFF          0                4 Y             REORG ...
SIMAP2     STAFF          0                4 Y             NOT BUILT ...
SIMAP2     SUPPLIERS     0                2 N             NOT BUILT ...
SIMAP2     SUPPLIERS     0                2 N             NOT BUILT ...

```

44 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación):

```

DICT_BUILD_TIMESTAMP      COMPRESS_DICT_SIZE EXPAND_DICT_SIZE ROWS_SAMPLED ...
-----
-                          0                0                0 ...
-                          0                0                0 ...
2009-03-31-12.11.02.000000  290              1890             22 ...
-                          0                0                0 ...
...
2009-03-31-11.08.18.000000  3968             3000             6 ...
2009-03-31-11.08.18.000000  13312            10944            6 ...
...
2009-03-31-12.19.30.000000  13312            5296             35 ...
-                          0                0                0 ...
-                          0                0                0 ...
-                          0                0                0 ...

```

Datos de salida de esta consulta (continuación):

| PAGES_SAVED_PERCENT | BYTES_SAVED_PERCENT | AVG_COMPRESS_REC_LENGTH | OBJECT_TYPE |
|---------------------|---------------------|-------------------------|-------------|
| 0 | 0 | 0 | DATA |
| 0 | 0 | 0 | XML |
| 20 | 25 | 21 | DATA |
| 0 | 0 | 0 | XML |
| ... | | | |
| 70 | 70 | 31 | DATA |
| 66 | 66 | 235 | XML |
| ... | | | |
| 38 | 38 | 27 | DATA |
| 0 | 0 | 0 | XML |
| 0 | 0 | 0 | DATA |
| 0 | 0 | 0 | XML |

Ejemplo 5: Ver un informe de la información de diccionario de objetos DATA de las tablas en el esquema SIMAP2.

```
SELECT * FROM TABLE (
    SYSPROC.ADMIN_GET_TAB_COMPRESS_INFO_V97('SIMAP2',' ','REPORT')
WHERE OBJECT_TYPE='DATA')
```

Datos de salida de esta consulta:

| TABSCHEMA | TABNAME | DBPARTITIONNUM | DATA_PARTITION_ID | COMPRESS_ATTR | DICT_BUILDER | ... |
|-----------|------------|----------------|-------------------|---------------|--------------|-----|
| SIMAP2 | ACT | 0 | 0 | N | NOT BUILT | ... |
| SIMAP2 | ADEFUSR | 0 | 0 | N | INSPECT | ... |
| ... | | | | | | |
| SIMAP2 | CUSTOMER | 0 | 1 | Y | REORG | ... |
| SIMAP2 | DEPARTMENT | 1 | 0 | N | NOT BUILT | ... |
| ... | | | | | | |
| SIMAP2 | STAFF | 0 | 4 | Y | REORG | ... |
| SIMAP2 | SUPPLIERS | 0 | 2 | N | NOT BUILT | ... |

22 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación):

| DICT_BUILD_TIMESTAMP | COMPRESS_DICT_SIZE | EXPAND_DICT_SIZE | ROWS_SAMPLED... |
|----------------------------|--------------------|------------------|-----------------|
| - | 0 | 0 | 0 ... |
| 2009-03-31-12.11.02.000000 | 290 | 1890 | 22 ... |
| ... | | | |
| 2009-03-31-11.08.18.000000 | 3968 | 3000 | 6 ... |
| - | 0 | 0 | 0 ... |
| ... | | | |
| 2009-03-31-12.19.30.000000 | 13312 | 5296 | 35 ... |
| - | 0 | 0 | 0 ... |

Datos de salida de esta consulta (continuación):

| PAGES_SAVED_PERCENT | BYTES_SAVED_PERCENT | AVG_COMPRESS_REC_LENGTH | OBJECT_TYPE |
|---------------------|---------------------|-------------------------|-------------|
| 0 | 0 | 0 | DATA |
| 20 | 25 | 21 | DATA |
| 70 | 70 | 31 | DATA |
| 0 | 0 | 0 | DATA |
| 38 | 38 | 27 | DATA |
| 0 | 0 | 0 | DATA |

Ejemplo 6: Ver un informe de la información de diccionario de objetos XML de la tabla CUSTOMER en el esquema SIMAP2.

```
SELECT * FROM TABLE (
    SYSPROC.ADMIN_GET_TAB_COMPRESS_INFO_V97('SIMAP2', 'CUSTOMER', 'REPORT'))
WHERE OBJECT_TYPE='XML'
```

Datos de salida de esta consulta:

| TABSCHEMA | TABNAME | DBPARTITIONNUM | DATA_PARTITION_ID | COMPRESS_ATTR | DICTIONARY_BUILDER | ... |
|-----------|----------|----------------|-------------------|---------------|--------------------|-----|
| SIMAP2 | CUSTOMER | 0 | 1 | Y | REORG | ... |

Datos de salida de esta consulta (continuación):

| DICTIONARY_BUILD_TIMESTAMP | COMPRESS_DICT_SIZE | EXPAND_DICT_SIZE | ROWS_SAMPLED | ... |
|----------------------------|--------------------|------------------|--------------|-----|
| 2009-03-31-11.08.18.000000 | 13312 | 10944 | 6 | ... |

Datos de salida de esta consulta (continuación):

| PAGES_SAVED_PERCENT | BYTES_SAVED_PERCENT | AVG_COMPRESS_REC_LENGTH | OBJECT_TYPE |
|---------------------|---------------------|-------------------------|-------------|
| 66 | 66 | 235 | XML |

Notas de uso

- Si se especifican *esquematabla* y *nombretabla*, sólo se devuelve información para esa tabla específica.
- Si se especifica el *esquematabla* pero *nombretabla* está vacío (") o es NULL, se devolverá información para todas las tablas del esquema determinado.
- Si *esquematabla* está vacío (") o es NULL y se especifica *nombretabla*, se devolverá un error. Para recuperar información para una tabla determinada, tanto el esquema como el nombre de la tabla deberán identificar a la tabla.
- Si *esquematabla* y *nombretabla* están vacíos (") o son NULL, se devolverá información para todas las tablas.
- Si no existe *esquematabla* o *nombretabla*, o si *nombretabla* no corresponde a un nombre de tabla (tipo T), un nombre de tabla de consultas materializadas (tipo S) o un nombre de tabla de jerarquía (tipo H), se devuelve un conjunto de resultados vacío.
- Cuando la función de tabla ADMIN_GET_TAB_COMPRESS_INFO_V97 esté recuperando datos para una tabla determinada, necesitará un bloqueo compartido en la fila correspondiente de SYSTABLES para garantizar la coherencia de los datos que se devuelven (por ejemplo, para garantizar que la tabla no se descarte mientras se recupera información para ella). El bloqueo sólo se mantendrá mientras tiene lugar la recuperación de la información de compresión para la tabla, no mientras está activa la llamada de función de tabla.
- Si la tabla a la que se emite la consulta es una tabla no XML, se devolverá una fila para el objeto de almacenamiento XML (XDA).

Vista administrativa ADMINTABCOMPRESSINFO y metadatos de la función de tabla ADMIN_GET_TAB_COMPRESS_INFO_V97

Tabla 71. Vista administrativa ADMINTABCOMPRESSINFO y metadatos de la función de tabla ADMIN_GET_TAB_COMPRESS_INFO_V97

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--------------------------------------|
| TABSCHEMA | VARCHAR(128) | Nombre de esquema |
| TABNAME | VARCHAR(128) | Nombre de tabla |
| DBPARTITIONNUM | SMALLINT | Número de partición de base de datos |
| DATA_PARTITION_ID | INTEGER | Número de partición de datos |

Tabla 71. Vista administrativa ADMINTABCOMPRESSINFO y metadatos de la función de tabla ADMIN_GET_TAB_COMPRESS_INFO_V97 (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|----------------------------|---------------|---|
| COMPRESS_ATTR | CHAR(1) | El estado del valor COMPRESS en la tabla, que puede ser uno de los siguientes: <ul style="list-style-type: none"> • 'Y' = Compresión de filas establecida en sí • 'N' = Compresión de filas establecida en no |
| DICTIONARY_BUILDER | VARCHAR(30) | Vía de acceso del código tomado para crear el diccionario, que puede ser una de las siguientes: <ul style="list-style-type: none"> • 'INSPECT' = INSPECT ROWCOMPESTIMATE • 'LOAD' = LOAD INSERT/REPLACE • 'NOT BUILT' = ningún diccionario disponible • 'REDISTRIBUTE' = REDISTRIBUTE • 'REORG' = REORG RESETDICTIONARY • 'TABLE GROWTH' = INSERT |
| DICTIONARY_BUILD_TIMESTAMP | TIMESTAMP | Fecha y hora en que se creó el diccionario. La granularidad horaria es de segundos. Si no hay ningún diccionario disponible, entonces la fecha y hora será NULL. |
| COMPRESS_DICT_SIZE | BIGINT | Tamaño del diccionario de compresión medido en bytes. |
| EXPAND_DICT_SIZE | BIGINT | Tamaño del diccionario de expansión medido en bytes. Si existe un diccionario histórico, este valor es la suma de los tamaños de los diccionarios actual e histórico. |
| ROWS_SAMPLED | INTEGER | Número de registros que han contribuido a crear el diccionario. Las tablas migradas con diccionarios de compresión devolverán NULL en esta columna. |
| PAGES_SAVED_PERCENT | SMALLINT | Porcentaje de páginas salvadas de la compresión. Esta información se basa en los datos de registro del almacenamiento intermedio de ejemplo únicamente. Las tablas migradas con diccionarios de compresión devolverán NULL en esta columna. |
| BYTES_SAVED_PERCENT | SMALLINT | Porcentaje de bytes salvadas de la compresión. Esta información se basa en los datos de registro del almacenamiento intermedio de ejemplo únicamente. Las tablas migradas con diccionarios de compresión devolverán NULL en esta columna. |
| AVG_COMPRESS_REC_LENGTH | SMALLINT | La longitud media de registro comprimida de los registros que permiten crear el diccionario. Las tablas migradas con diccionarios de compresión devolverán NULL en esta columna. |
| OBJECT_TYPE | VARCHAR(4) | El tipo del objeto. Dependiendo del tipo, esta fila contiene valores que pertenecen al objeto especificado. Los valores pueden ser uno de los siguientes: <ul style="list-style-type: none"> • 'XML' • 'DATA' |

Vista administrativa ADMINTABINFO y función de tabla ADMIN_GET_TAB_INFO_V97 – Recuperar información de tamaño y estado

La función de tabla ADMINTABINFO y la función de tabla ADMIN_GET_TAB_INFO_V97 proporcionan métodos para recuperar la información de tamaño y estado de tabla que actualmente no está disponible en las vistas de catálogo.

ADMINTABINFO, vista administrativa

La vista administrativa ADMINTABINFO devuelve información de tamaño y estado sólo para las tablas, las tablas de consultas materializadas (MQT) y las tablas de jerarquía. En la vista de catálogo SYSCAT.TABLES, estos tipos de tablas se informan con T para tabla, S para tablas de consultas materializadas y H para tablas de jerarquía. Para una tabla, la información se devuelve en el nivel de partición de datos y en el nivel de partición de base de datos.

El esquema es SYSIBMADM.

En la tabla Vista administrativa ADMINTABINFO y metadatos de la función de tabla ADMIN_GET_TAB_INFO_V97 encontrará una lista completa de la información que puede devolverse.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa ADMINTABINFO
- Privilegio CONTROL en la vista administrativa ADMINTABINFO
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE en la función de tabla ADMIN_GET_TAB_INFO_V97
- Autorización DATAACCESS

Ejemplos

Ejemplo 1: Recuperar información de tamaño y de estado para todas las tablas

```
SELECT * FROM SYSIBMADM.ADMINTABINFO
```

Ejemplo 2: Determinar la cantidad de espacio físico que utiliza gran número de tablas que se han llenado escasamente.

```
SELECT TABSCHEMA, TABNAME, SUM(DATA_OBJECT_P_SIZE),  
      SUM(INDEX_OBJECT_P_SIZE), SUM(LONG_OBJECT_P_SIZE),  
      SUM(LOB_OBJECT_P_SIZE), SUM(XML_OBJECT_P_SIZE)  
FROM SYSIBMADM.ADMINTABINFO GROUP BY TABSCHEMA, TABNAME
```

Ejemplo 3: Identificar tablas que son idóneas para utilizar RID grandes pero que, actualmente, no están habilitadas para utilizar RID grandes.

```
SELECT TABSCHEMA, TABNAME FROM SYSIBMADM.ADMINTABINFO  
WHERE LARGE_RIDS = 'P'
```

Ejemplo 4: Identificar qué tablas utilizan índices de tipo 1 y que necesitan una reorganización para la conversión en índices de tipo 2.

```
SELECT TABSCHEMA, TABNAME FROM SYSIBMADM.ADMINTABINFO
WHERE INDEX_TYPE = 1
```

Ejemplo 5: Identificar qué tablas tienen datos XML en formato tipo 1 y solicitar un movimiento de tabla en línea para convertir al formato tipo 2.

```
SELECT TABSCHEMA, TABNAME FROM SYSIBMADM.ADMINTABINFO
WHERE XML_RECORD_TYPE=1
```

Ejemplo 4: Comprobar el tipo actual de información estadística reunida para la tabla T1

```
SELECT SUBSTR(TABSCHEMA, 1, 10) AS TBSCHM, SUBSTR(TABNAME, 1, 10)
AS TBNAME, STATSTYPE FROM SYSIBMADM.ADMINTABINFO WHERE TABNAME = 'T1';
```

| TBSCHM | TBNAME | STATSTYPE |
|----------|--------|-----------|
| DB2USER1 | T1 | U |

1 registro(s) seleccionado(s).

ADMIN_GET_TAB_INFO_V97, función de tabla

La función de tabla ADMIN_GET_TAB_INFO_V97 devuelve la misma información que la vista administrativa ADMINTABINFO, pero le permite especificar un esquema y un nombre de tabla.

En la tabla Vista administrativa ADMINTABINFO y metadatos de la función de tabla ADMIN_GET_TAB_INFO_V97 encontrará una lista completa de la información que puede devolverse.

Sintaxis

►►—ADMIN_GET_TAB_INFO_V97—(—*esquematabla*—,—*nombretabla*—)—————►►

El esquema es SYSPROC.

Parámetros de la función de tabla

esquematabla

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de esquema.

nombretabla

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de tabla, un nombre de tabla de consultas materializadas o un nombre de tabla de jerarquía.

Autorización

Privilegio EXECUTE en la función de tabla ADMIN_GET_TAB_INFO_V97.

Ejemplos

Ejemplo 1: Recuperar información de tamaño y estado para la tabla DBUSER1.EMPLOYEE.

```
SELECT * FROM TABLE (SYSPROC.ADMIN_GET_TAB_INFO_V97('DBUSER1', 'EMPLOYEE'))
AS T
```

Ejemplo 2: Imaginemos que existe una tabla no particionada (DBUSER1.EMPLOYEE) cuyos objetos asociados (por ejemplo, los índices y los LOB) se almacenan en un único espacio de tablas. Calcule cuánto espacio físico del espacio de tablas utiliza la tabla:

```
SELECT (data_object_p_size + index_object_p_size + long_object_p_size +
lob_object_p_size + xml_object_p_size) as total_p_size
FROM TABLE( SYSPROC.ADMIN_GET_TAB_INFO_V97( 'DBUSER1', 'EMPLOYEE' )) AS T
```

Calcular cuánto espacio se necesitaría si la tabla se moviera a otro espacio de tablas, donde el nuevo espacio de tablas tendría el mismo tamaño de página y tamaño de extensión que el espacio de tablas original:

```
SELECT (data_object_l_size + index_object_l_size + long_object_l_size +
lob_object_l_size + xml_object_l_size) as total_l_size
FROM TABLE( SYSPROC.ADMIN_GET_TAB_INFO_V97( 'DBUSER1', 'EMPLOYEE' )) AS T
```

Ejemplo 3: Determinar el tamaño total para los diccionarios de compresión para la tabla DBUSER1.EMPLOYEE.

```
SELECT SUBSTR(TABSHEMA,1,10) AS TBSHEMA, SUBSTR(TABNAME,1,10) AS TBNAME,
DICTIONARY_SIZE + XML_DICTIONARY_SIZE AS TOTAL_DICTIONARY_SIZE
FROM TABLE(SYSPROC.ADMIN_GET_TAB_INFO_V97('DBUSER1','EMPLOYEE'))
```

Ejemplo 4: Determinar la cantidad de espacio reclamable a una tabla de clústeres multidimensionales SAMPLE.STAFF:

```
SELECT RECLAIMABLE_SPACE
FROM TABLE(SYSPROC.ADMIN_GET_TAB_INFO_V97('SAMPLE','STAFF'))
```

Notas de uso

- Si se especifican *esquematabla* y *nombretabla*, sólo se devuelve información para esa tabla específica.
- Si se especifica *esquematabla* pero *nombretabla* es NULL o la serie vacía ("), se devuelve información para todas las tablas del esquema determinado.
- Si *esquematabla* es NULL o la serie vacía (") y se especifica *nombretabla*, se devolverá un error. Para recuperar información para una tabla determinada, tanto el esquema como el nombre de la tabla deberán identificar a la tabla.
- Si tanto *esquematabla* como *nombretabla* son NULL o la serie vacía ("), se devolverá información para todas las tablas.
- Si no existe *esquematabla* o *nombretabla*, o si *nombretabla* no corresponde a un nombre de tabla (tipo T), un nombre de tabla de consultas materializadas (tipo S) o un nombre de tabla de jerarquía (tipo H), se devuelve un conjunto de resultados vacío.
- Cuando la función de tabla ADMIN_GET_TAB_INFO_V97 recupere datos para una tabla determinada, necesitará un bloqueo compartido en la fila correspondiente de SYSTABLES para garantizar la coherencia de los datos que se devuelven (por ejemplo, para garantizar que la tabla no se descarte mientras se recupera información para ésta). El bloqueo sólo se mantendrá mientras tiene lugar la recuperación de la información de tamaño y de estado para la tabla, no mientras está activa la llamada de función de tabla.
- El tamaño físico que se informa para las tablas de los espacios de tablas SMS es igual al tamaño lógico.
- Cuando exista una reorganización in situ activa en una tabla, no se calculará el tamaño físico del objeto de datos (DATA_OBJECT_P_SIZE). Sólo se devolverá el

tamaño lógico. Para determinar si una reorganización in situ está activa en la tabla, consulte la columna de salida INPLACE_REORG_STATUS.

- El tamaño lógico que se informa para los objetos LOB creados antes de la Versión 8 de DB2 UDB puede que sea superior al tamaño físico si los objetos todavía no se han reorganizado.

REDISTRIBUTING_PENDING

1. ninguna redistribución ejecutada para la tabla determinada N
2. la redistribución ha empezado a ejecutarse en el grupo de partición de bases de datos pero no en la tabla N
3. la redistribución ha fallado en la fase previa al movimiento de datos N
4. la redistribución ha fallado en la fase de movimiento de datos Y
5. la redistribución se ha realizado satisfactoriamente y se ha confirmado para la tabla N

Vista administrativa ADMINTABINFO y metadatos de función de tabla ADMIN_GET_TAB_INFO_V97

Tabla 72. Vista administrativa ADMINTABINFO y metadatos de función de tabla ADMIN_GET_TAB_INFO_V97

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| TABSCHEMA | VARCHAR(128) | Nombre de esquema. |
| TABNAME | VARCHAR(128) | Nombre de tabla. |
| TABTYPE | CHAR(1) | Tipo de tabla: <ul style="list-style-type: none"> • 'H' = tabla de jerarquía • 'S' = tabla de consultas materializadas • 'T' = tabla |
| DBPARTITIONNUM | SMALLINT | Número de partición de base de datos. |
| DATA_PARTITION_ID | INTEGER | Número de partición de datos. |
| AVAILABLE | CHAR(1) | Estado de la tabla: <ul style="list-style-type: none"> • 'N' = la tabla no está disponible. Si la tabla no está disponible, todas las demás columnas de salida relacionadas con el tamaño y el estado serán NULL. • 'Y' = la tabla está disponible. <p>Nota: La realización de una recuperación en avance en una carga no recuperable establecerá una tabla en estado de no disponibilidad.</p> |

Tabla 72. Vista administrativa ADMINTABINFO y metadatos de función de tabla ADMIN_GET_TAB_INFO_V97 (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|---------------------|---------------|---|
| DATA_OBJECT_L_SIZE | BIGINT | <p>Tamaño lógico del objeto de datos. Cantidad de espacio de disco que se ha asignado lógicamente para la tabla y que se informa en kilobytes. El tamaño lógico es la cantidad de espacio que la tabla reconoce. Puede que sea inferior a la cantidad de espacio que se ha asignado físicamente para la tabla (por ejemplo, en el caso de un truncamiento de tabla lógica). Para las tablas de clúster multidimensional (MDC), este tamaño incluye el tamaño lógico del objeto de correlación de bloques. El tamaño devuelto tiene en cuenta las extensiones completas que se han asignado lógicamente para la tabla y, para los objetos que se han creado en espacios de tablas DMS, incluye una estimación de las extensiones EMP (Extent Map Page). Este tamaño representa el tamaño lógico sólo de la tabla base. En otras columnas se informa del espacio que utilizan los datos LOB, los datos largos, los índices y los objetos XML.</p> |
| DATA_OBJECT_P_SIZE | BIGINT | <p>Tamaño físico del objeto de datos. Cantidad de espacio de disco que se ha asignado físicamente para la tabla y que se informa en kilobytes. Para las tablas MDC, este tamaño incluye el tamaño del objeto de correlación de bloques. El tamaño devuelto tiene en cuenta las extensiones completas que se han asignado para la tabla e incluye las extensiones EMP para los objetos que se han creado en los espacios de tablas DMS. Este tamaño representa el tamaño físico sólo de la tabla base. En otras columnas se informa del espacio que utilizan los datos LOB, los datos largos, los índices y los objetos XML.</p> |
| INDEX_OBJECT_L_SIZE | BIGINT | <p>Tamaño lógico del objeto de índice. Cantidad de espacio de disco que se ha asignado lógicamente para los índices definidos en la tabla y que se informa en kilobytes. El tamaño lógico es la cantidad de espacio que la tabla reconoce. Puede que sea inferior a la cantidad de espacio que se ha asignado físicamente para contener los datos de índice para la tabla (por ejemplo, en el caso de un truncamiento de tabla lógica). El tamaño devuelto tiene en cuenta las extensiones completas que se han asignado lógicamente para los índices y, para los índices que se han creado en espacios de tablas DMS, incluye una estimación de las extensiones EMP.</p> <p>Para los índices particionados de tablas particionadas, es el tamaño lógico del objeto de índice que contiene particiones de índice para la partición de datos identificada por DATA_PARTITION_ID. Este valor no toma en consideración los índices no particionados de tablas particionadas. Para obtener más información sobre los índices particionados y no particionados, utilice la función ADMIN_GET_INDEX_INFO.</p> |

Tabla 72. Vista administrativa ADMINTABINFO y metadatos de función de tabla
ADMIN_GET_TAB_INFO_V97 (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|---------------------|---------------|--|
| INDEX_OBJECT_P_SIZE | BIGINT | <p>Tamaño físico del objeto de índice. Cantidad de espacio de disco que se ha asignado físicamente para los índices definidos en la tabla y que se informa en kilobytes. El tamaño devuelto tiene en cuenta las extensiones completas que se han asignado para los índices e incluye las extensiones EMP para los índices que se han creado en los espacios de tablas DMS.</p> <p>Para los índices particionados de tablas particionadas, es el tamaño físico del objeto de índice que contiene particiones de índice para la partición de datos identificada por DATA_PARTITION_ID. Este valor no toma en consideración los índices no particionados de tablas particionadas. Para obtener más información sobre los índices particionados y no particionados, utilice la función ADMIN_GET_INDEX_INFO.</p> |
| LONG_OBJECT_L_SIZE | BIGINT | <p>Tamaño lógico del objeto largo. Cantidad de espacio de disco que se ha asignado lógicamente para los datos de campo largo en una tabla y que se informa en kilobytes. El tamaño lógico es la cantidad de espacio que la tabla reconoce. Puede que sea inferior a la cantidad de espacio que se ha asignado físicamente para contener los datos de campo largo para la tabla (por ejemplo, en el caso de un truncamiento de tabla lógica). El tamaño devuelto tiene en cuenta las extensiones completas que se han asignado lógicamente para los datos de campo largo y, para los datos de campo largo que se han creado en espacios de tablas DMS, incluye una estimación de las extensiones EMP.</p> |
| LONG_OBJECT_P_SIZE | BIGINT | <p>Tamaño físico del objeto largo. Cantidad de espacio de disco que se ha asignado físicamente para los datos de campo largo en una tabla y que se informa en kilobytes. El tamaño devuelto tiene en cuenta las extensiones completas que se han asignado para los datos de campo largo e incluye las extensiones EMP para los datos de campo largo que se han creado en los espacios de tablas DMS.</p> |
| LOB_OBJECT_L_SIZE | BIGINT | <p>Tamaño lógico del objeto LOB. Cantidad de espacio de disco que se ha asignado lógicamente para los datos LOB en una tabla y que se informa en kilobytes. El tamaño lógico es la cantidad de espacio que la tabla reconoce. Puede que sea inferior a la cantidad de espacio que se ha asignado físicamente para contener los datos LOB para la tabla (por ejemplo, en el caso de un truncamiento de tabla lógica). El tamaño incluye el espacio que se ha asignado lógicamente para el objeto de asignación de LOB. El tamaño devuelto tiene en cuenta las extensiones completas que se han asignado lógicamente para los datos LOB y, para los datos LOB que se han creado en espacios de tablas DMS, incluye una estimación de las extensiones EMP.</p> |

Tabla 72. Vista administrativa ADMINTABINFO y metadatos de función de tabla ADMIN_GET_TAB_INFO_V97 (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|----------------------|---------------|---|
| LOB_OBJECT_P_SIZE | BIGINT | Tamaño físico del objeto LOB. Cantidad de espacio de disco que se ha asignado físicamente para los datos LOB en una tabla y que se informa en kilobytes. El tamaño incluye el espacio que se ha asignado para el objeto de asignación de LOB. El tamaño devuelto tiene en cuenta las extensiones completas que se han asignado para los datos LOB e incluye las extensiones EMP para los datos LOB que se han creado en los espacios de tablas DMS. |
| XML_OBJECT_L_SIZE | BIGINT | Tamaño lógico del objeto XML. Cantidad de espacio de disco que se ha asignado lógicamente para los datos XML en una tabla y que se informa en kilobytes. El tamaño lógico es la cantidad de espacio que la tabla reconoce. Puede que sea inferior a la cantidad de espacio que se ha asignado físicamente para contener los datos XML para la tabla (por ejemplo, en el caso de un truncamiento de tabla lógica). El tamaño devuelto tiene en cuenta las extensiones completas que se han asignado lógicamente para los datos XML y, para los datos XML que se han creado en espacios de tablas DMS, incluye una estimación de las extensiones EMP. |
| XML_OBJECT_P_SIZE | BIGINT | El tamaño físico del objeto XML. Cantidad de espacio de disco que se ha asignado físicamente para los datos XML en una tabla y que se informa en kilobytes. El tamaño devuelto tiene en cuenta las extensiones completas que se han asignado para los datos XML e incluye las extensiones EMP para los datos XML que se han creado en los espacios de tablas DMS. |
| INDEX_TYPE | SMALLINT | Indica el tipo de índices que actualmente está utilizándose para la tabla. Devuelve: <ul style="list-style-type: none"> • 1 si están utilizándose índices de tipo 1. • 2 si están utilizándose índices de tipo 2. |
| REORG_PENDING | CHAR(1) | El valor 'Y' indica que se ha aplicado una modificación de reorganización recomendada a la tabla y que se necesita una reorganización clásica (fuera de línea). De lo contrario, se devuelve 'N'. |
| INPLACE_REORG_STATUS | VARCHAR(10) | Estado actual de una reorganización de tabla in situ en la tabla. El valor de estado puede ser uno de los siguientes: <ul style="list-style-type: none"> • ABORTED (en estado PAUSED, pero incapaz de RESUME; se necesita STOP) • EXECUTING • NULL (si no se ha realizado ninguna reorganización in situ en la tabla) • PAUSED |
| LOAD_STATUS | VARCHAR(12) | Estado actual de una operación de carga para la tabla. El valor de estado puede ser uno de los siguientes: <ul style="list-style-type: none"> • IN_PROGRESS • NULL (si no existe ninguna carga en progreso para la tabla y si la tabla no está en estado de carga pendiente) • PENDING |

Tabla 72. Vista administrativa ADMINTABINFO y metadatos de función de tabla
ADMIN_GET_TAB_INFO_V97 (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------------|---------------|---|
| READ_ACCESS_ONLY | CHAR(1) | 'Y' si la tabla se encuentra en estado de Acceso de sólo lectura; de lo contrario, 'N'. El valor 'N' no significa que la tabla ofrezca acceso completo. Si existe una carga en proceso o pendiente, el valor 'Y' significa que los datos de tabla están disponibles para el acceso de lectura y el valor 'N' significa que no puede accederse a la tabla. De forma similar, si el estado de la tabla corresponde a un establecimiento de integridad pendiente (vea la columna SYSCAT.TABLES STATUS), el valor 'N' significa que no puede accederse a la tabla. |
| NO_LOAD_RESTART | CHAR(1) | El valor 'Y' indica que la tabla se encuentra en un estado de carga parcial y que no será posible realizar un reinicio de carga. De lo contrario, se devuelve el valor 'N'. |
| NUM_REORG_REC_ALTERS | SMALLINT | Número de operaciones de modificación de reorganización recomendada (por ejemplo, operaciones de modificación después de las cuales se necesita una reorganización) que se han realizado para esta tabla desde la última reorganización. |
| INDEXES_REQUIRE_REBUILD | CHAR(1) | Para las tablas no particionadas, 'Y' si debe volver a crearse cualquiera de los índices definidos en la tabla; de lo contrario, 'N'. Para las tablas particionadas, 'Y' si se debe volver a crear alguno de las particiones de índices para la partición de datos que identifica DATA_PARTITION_ID; 'N' en caso contrario. |
| LARGE_RIDS | CHAR(1) | Indica si la tabla utiliza o no los ID de fila grandes (RID) (número de página de 4 bytes, número de ranura de 2 bytes). El valor 'Y' indica que la tabla utiliza RID grandes y el valor 'N' indica que no utiliza RID grandes. Se devolverá el valor 'P' (pendiente) si la tabla da soporte a los RID grandes (es decir, la tabla se encuentra en un espacio de tablas grande) pero, como mínimo, uno de los índices para la tabla no se ha reorganizado o no ha vuelto a crearse todavía, por lo que la tabla todavía utiliza RID de 4 bytes (lo que significa que debe realizarse una acción para convertir la tabla o los índices). |
| LARGE_SLOTS | CHAR(1) | Indica si la tabla utiliza o no ranuras grandes (lo que admite más de 255 filas por página). El valor 'Y' indica que la tabla utiliza ranuras grandes y el valor 'N' indica que no utiliza ranuras grandes. Se devolverá el valor 'P' (pendiente) si la tabla da soporte a las ranuras grandes (es decir, la tabla se encuentra en un espacio de tablas grande), pero todavía no se ha realizado en la tabla ninguna operación de truncamiento de tabla u operación de reorganización de tabla fuera de línea, por lo que todavía utiliza un máximo de 255 filas por página. |
| DICTIONARY_SIZE | BIGINT | Tamaño del diccionario de tabla, en bytes, utilizado para la compresión de filas, si existe un diccionario de compresión de filas para la tabla. Si existe un diccionario histórico, este valor es la suma de los tamaños de los diccionarios actual e histórico. |
| BLOCKS_PENDING_CLEANUP | BIGINT | Para tablas MDC, el número de bloqueos pendientes de limpieza. Para tablas no MDC este valor siempre será cero. |

Tabla 72. Vista administrativa ADMINTABINFO y metadatos de función de tabla ADMIN_GET_TAB_INFO_V97 (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|---------------------|---------------|---|
| STATSTYPE | CHAR(1) | <ul style="list-style-type: none"> • 'F' = Estadísticas elaboradas por el sistema sin exploración de tabla o índice. Estas estadísticas se almacenan en la memoria y difieren de lo que se almacena en los catálogos del sistema. Es un estado temporal y es posible que DB2 otorgue estadísticas completas y las almacene en catálogos de sistema. • 'A' = Estadísticas otorgadas de forma asincrónica por el sistema. Las estadísticas han sido reunidas automáticamente por DB2 mediante un proceso de fondo y se han almacenado en los catálogos de sistema. • 'S' = Estadísticas otorgadas de forma sincrónica por el sistema. Las estadísticas han sido reunidas automáticamente por DB2 durante la compilación de sentencias de SQL. Estas estadísticas se almacenan en la memoria y difieren de lo que se almacena en los catálogos del sistema. Es un estado temporal y es posible que DB2 almacene las estadísticas en los catálogos de sistema. • 'U' = Estadísticas otorgadas por el usuario. La elaboración de estadísticas la ha iniciado el usuario mediante un programa de utilidad del tipo RUNSTATS, CREATE INDEX, LOAD, REDISTRIBUTE o mediante estadísticas de catálogo de sistema actualizadas manualmente. • NULL = tipo desconocido |
| XML_RECORD_TYPE | SMALLINT | <p>Indica el tipo de registro XML que actualmente está utilizándose para la tabla.</p> <ul style="list-style-type: none"> • 1 si se está utilizando el formato de registro XML de tipo 1 (un solo nodo). • 2 si se está utilizando el formato de registro XML de tipo 2 (varios nodos). • Nulo si la tabla no tiene columnas XML. |
| RECLAIMABLE_SPACE | BIGINT | <p>En una tabla MDC de un espacio de tablas DMS, este valor indica la cantidad de espacio de disco que se puede reclamar mediante la ejecución del mandato REORG con la opción RECLAIM. El espacio de disco se indica en kilobytes. En el resto de las tablas, el valor es cero.</p> |
| XML_DICTIONARY_SIZE | BIGINT | <p>Tamaño del diccionario XML, en bytes, utilizado para la compresión de datos si existe un diccionario de compresión de datos para el objeto de almacenamiento XML. Si la tabla no contiene ninguna columna XML o si el diccionario de compresión no se ha creado, el valor es 0.</p> |

Vista administrativa ADMINTEPCOLUMNS y función de tabla ADMIN_GET_TEMP_COLUMNS - Recuperar información de columna para tablas temporales

La vista administrativa ADMINTEPCOLUMNS y la función de tabla ADMIN_GET_TEMP_COLUMNS proporcionan métodos para recuperar información de atributos de columna para las tablas temporales creadas y tablas temporales declaradas.

Aunque las vistas de catálogo contienen información de atributos de columna para instancias de tablas temporales creadas, no contienen esta información para tablas temporales declaradas.

ADMINTEMPCOLUMNS, vista administrativa

La vista administrativa ADMINTEMPCOLUMNS devuelve información de atributos de columna para las instancias de tablas temporales creadas y tablas temporales declaradas.

El esquema es SYSIBMADM.

En la tabla Vista administrativa ADMINTEMPCOLUMNS y metadatos de la función de tabla ADMIN_GET_TEMP_COLUMNS encontrará una lista completa de la información que puede devolverse.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa ADMINTEMPCOLUMNS
- Privilegio CONTROL en la vista administrativa ADMINTEMPCOLUMNS
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE en la función de tabla ADMIN_GET_TEMP_COLUMNS
- Autorización DATAACCESS

Ejemplos

Ejemplo 1: Recuperar información de atributos de columna para todas las instancias de las tablas temporales creadas y las tablas temporales declaradas que existen actualmente en la base de datos.

```
SELECT * FROM SYSIBMADM.ADMINTEMPCOLUMNS
```

Ejemplo 2: Determinar qué tablas temporales activas en la base de datos están utilizando el tipo de datos definido por el usuario USMONEY.

```
SELECT APPLICATION_HANDLE, TABSCHEMA, TABNAME  
FROM SYSIBMADM.ADMINTEMPCOLUMNS  
WHERE TYPENAME = 'USMONEY'
```

Ejemplo 3: Recuperar el esquema de tabla, nombre de tabla y nombres de columna para todas las tablas temporales declaradas que ha declarado SYSTEM_USER.

```
SELECT T.TABSCHEMA, T.TABNAME, C.COLNAME  
FROM SYSIBMADM.ADMINTEMPCOLUMNS C, SYSIBMADM.ADMINTEMPTABLES T  
WHERE T.TEMPABTYPE = 'D'  
AND T.INSTANTIATOR = SYSTEM_USER  
AND T.TABSCHEMA = C.TABSCHEMA  
AND T.TABNAME = C.TABNAME
```

ADMIN_GET_TEMP_COLUMNS, función de tabla

La función de tabla ADMIN_GET_TEMP_TABLES devuelve la misma información que la vista administrativa ADMINTEMPCOLUMNS, pero le permite especificar un nombre de esquema y un nombre de tabla.

En la tabla Vista administrativa ADMINTEMPCOLUMNS y metadatos de la función de tabla ADMIN_GET_TEMP_COLUMNS encontrará una lista completa de la información que puede devolverse.

Sintaxis

►—ADMIN_GET_TEMP_COLUMNS—(—*descriptorcontexto_aplicación*—,—*esquematabla*—,—*nombretabla*—)————►

El esquema es SYSPROC.

Parámetros de la función de tabla

descriptorcontexto_aplicación

Argumento de entrada de tipo BIGINT que especifica un descriptor de contexto de aplicación. Si se especifica *descriptorcontexto_aplicación*, se devuelven datos solamente para la conexión especificada; si *descriptorcontexto_aplicación* es NULL, se devuelven datos para todas las conexiones.

esquematabla

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de esquema.

nombretabla

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de tabla temporal creada o un nombre de tabla temporal declarada.

Autorización

Privilegio EXECUTE para la función de tabla ADMIN_GET_TEMP_COLUMNS.

Ejemplos

Ejemplo 1: Recuperar información de columna para la tabla temporal declarada TEMPEMPLYEE para la conexión actual.

```
SELECT *
FROM TABLE(
  SYSPROC.ADMIN_GET_TEMP_COLUMNS(
    APPLICATION_ID(), 'SESSION', 'TEMPEMPLYEE')
AS T
```

Notas de uso

- Si se especifican *esquematabla* y *nombretabla*, sólo se devuelve información para esa tabla temporal específica.
- Si se especifica *esquematabla* pero *nombretabla* es NULL o la serie vacía ("), se devolverá información para todas las tablas del esquema determinado.
- Si *esquematabla* es NULL o la serie vacía (") y se especifica *nombretabla*, se devolverá un error. Para recuperar información para una tabla temporal determinada, tanto el esquema como el nombre de la tabla deberán identificar a la tabla.
- Si *esquematabla* y *nombretabla* son NULL o la serie vacía ("), se devolverá información para todas las tablas temporales para la conexión o todas las conexiones, en función del valor de *descriptorcontexto_aplicación*.
- Si *esquematabla* o *nombretabla* no existen, si *nombretabla* no corresponde a un nombre de tabla temporal o si las instancias de la tabla temporal identificada no existen en la base de datos, se devolverá un conjunto de resultados vacío.

Vista administrativa ADMINTEMPCOLUMNS y metadatos de la función de tabla ADMIN_GET_TEMP_COLUMNS

Tabla 73. Vista administrativa ADMINTEMPCOLUMNS y metadatos de la función de tabla ADMIN_GET_TEMP_COLUMNS

| Nombre de columna | Tipo de datos | Descripción |
|--------------------|---------------|---|
| APPLICATION_HANDLE | BIGINT | ID exclusivo a nivel del sistema para la aplicación. En una base de datos de una sola partición, este identificador consiste en un recuento de 16 bits. En una base de datos de varias particiones, este identificador consiste en el número de partición de coordinación concatenado con un recuento de 16 bits. Además, este identificador es el mismo en cada partición en la que la aplicación puede realizar una conexión secundaria. |
| APPLICATION_NAME | VARCHAR(256) | Nombre de la aplicación. |
| TABSHEMA | VARCHAR(128) | Nombre de esquema de la tabla temporal que contiene la columna. |
| TABNAME | VARCHAR(128) | Nombre de tabla de la tabla temporal que contiene la columna. |
| COLNAME | VARCHAR(128) | Nombre de la columna. |
| COLNO | SMALLINT | Número de esta columna en la tabla (empezando por 0). |
| TYPESHEMA | VARCHAR(128) | Nombre de esquema del tipo de datos de la columna. |
| TYPENAME | VARCHAR(128) | Nombre no calificado del tipo de datos de la columna. |
| LENGTH | INTEGER | Longitud máxima de los datos. El valor es 0 para tipos diferenciados. La columna LENGTH indica la precisión para campos DECIMAL e indica el número de bytes de almacenamiento necesarios para las columnas de coma flotante decimal; es decir, 8 y 16 para DECFLOAT(16) y DECFLOAT(34), respectivamente. |
| SCALE | SMALLINT | Escala si el tipo de columna es DECIMAL o el número de dígitos de segundos fraccionarios si el tipo de columna es TIMESTAMP; 0 en caso contrario. |
| DEFAULT | VARCHAR(254) | Valor por omisión de la columna de una tabla expresada como constante, registro especial o función de conversión apropiada para el tipo de datos de la columna. También puede ser la palabra clave NULL. Los valores se pueden convertir a partir de lo que se ha especificado como valor por omisión. Por ejemplo, las constantes de fecha y hora se muestran en formato ISO, los nombres de funciones de conversión se califican con nombres de esquema y los identificadores se delimitan. El valor nulo si no se ha especificado una cláusula DEFAULT o la columna es una columna de vista. |
| NULLS | CHAR(1) | Atributo de capacidad de nulos para la columna. <ul style="list-style-type: none"> • "Y" = La columna puede contener nulos • "N" = La columna no puede contener nulos <p>El valor puede ser "N" para una columna de vista derivada de una expresión o función. No obstante, estas columnas permiten valores nulos cuando la sentencia que utiliza la vista se procesa con avisos para errores aritméticos.</p> |

Tabla 73. Vista administrativa ADMINTEMPCOLUMNS y metadatos de la función de tabla ADMIN_GET_TEMP_COLUMNS (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| CODEPAGE | SMALLINT | Página de códigos utilizada para los datos de esta columna; 0 si la columna está definida como FOR BIT DATA o no es de tipo serie. |
| LOGGED | CHAR(1) | Sólo se aplica a las columnas cuyo tipo es LOB o diferenciado en base a LOB; blanco en caso contrario. <ul style="list-style-type: none"> • "Y" = La columna se anota cronológicamente • "N" = La columna no se anota cronológicamente |
| COMPACT | CHAR(1) | Sólo se aplica a las columnas cuyo tipo es LOB o diferenciado en base a LOB; blanco en caso contrario. <ul style="list-style-type: none"> • "Y" = La columna se compacta en el almacenamiento • "N" = La columna no se compacta |
| INLINE_LENGTH | INTEGER | Tamaño máximo en bytes de la representación interna de una instancia de un documento XML de un tipo estructurado que puede almacenarse en la tabla base; 0 cuando no es de aplicación. |
| IDENTITY | CHAR(1) | <ul style="list-style-type: none"> • "Y" = Columna de identidad • "N" = No es una columna de identidad |
| GENERATED | CHAR(1) | Tipo de columna generada. <ul style="list-style-type: none"> • "A" = Siempre se genera un valor de columna • "D" = Por omisión se genera un valor de columna • Blanco = No se genera ninguna columna |

Vista administrativa ADMINTEMPTABLES y función de tabla ADMIN_GET_TEMP_TABLES - Recuperar información para tablas temporales

La vista administrativa ADMINTEMPTABLES y la función de tabla ADMIN_GET_TEMP_TABLES proporcionan métodos para recuperar información de atributos de tabla y de tiempo de creación de instancias para las instancias de tablas temporales creadas y tablas temporales declaradas.

Aunque las vistas de catálogo contienen información de atributos de tabla para tablas temporales creadas, no contienen esta información para tablas temporales declaradas. Asimismo, las vistas de catálogo no contienen información de tiempo de creación de instancias de tablas para las tablas temporales creadas o las tablas temporales declaradas.

ADMINTEMPTABLES, vista administrativa

La vista administrativa ADMINTEMPTABLES devuelve información de atributos de tabla y de tiempo de creación de instancias para las instancias de tablas temporales creadas y tablas temporales declaradas.

El esquema es SYSIBMADM.

En la tabla Vista administrativa ADMINTEMPTABLES y metadatos de la función de tabla ADMIN_GET_TEMP_TABLES encontrará una lista completa de la información que puede devolverse.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa ADMINTEMPTABLES
- Privilegio CONTROL en la vista administrativa ADMINTEMPTABLES
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE en la función de tabla ADMIN_GET_TEMP_TABLES
- Autorización DATAACCESS

Ejemplos

Ejemplo 1: Recuperar información de atributos de tabla y de tiempo de creación de instancias para todas las instancias de las tablas temporales creadas y las tablas temporales declaradas que existen actualmente en la base de datos.

```
SELECT * FROM SYSIBMADM.ADMINTEMPTABLES
```

Ejemplo 2: Determinar las conexiones que tienen una instancia de una tabla temporal creada.

```
SELECT APPLICATION_HANDLE, TABSCHEMA, TABNAME  
FROM SYSIBMADM.ADMINTEMPTABLES  
WHERE TEMPTABTYPE = 'C'
```

Ejemplo 3: Recuperar información de atributos de tabla y de tiempo de creación de instancias para todas las tablas temporales declaradas para todas las tablas inicializadas por el usuario que se conectó a la base de datos.

```
SELECT TABSCHEMA, TABNAME, ONCOMMIT, ONROLLBACK,  
INSTANTIATION_TIME FROM SYSIBMADM.ADMINTEMPTABLES  
WHERE TEMPTABTYPE = 'D' AND INSTANTIATOR = SYSTEM_USER
```

ADMIN_GET_TEMP_TABLES, función de tabla

La función de tabla ADMIN_GET_TEMP_TABLES devuelve la misma información que la vista administrativa ADMINTABINFO, pero le permite especificar un nombre de esquema y un nombre de tabla.

En la tabla Vista administrativa ADMINTABINFO y metadatos de la función de tabla ADMIN_GET_TEMP_TABLES encontrará una lista completa de la información que puede devolverse.

Sintaxis

```
►►—ADMIN_GET_TEMP_TABLES—(—descriptorcontexto_aplicación—,—esquematabla—,—nombretabla—)————►◄
```

El esquema es SYSPROC.

Parámetros de la función de tabla

descriptorcontexto_aplicación

Argumento de entrada de tipo BIGINT que especifica un descriptor de contexto de aplicación. Si se especifica *descriptorcontexto_aplicación*, se devuelven datos solamente para la conexión especificada; si *descriptorcontexto_aplicación* es NULL, se devuelven datos para todas las conexiones.

esquematabla

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de esquema.

nombretabla

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de tabla temporal creada o un nombre de tabla temporal declarada.

Autorización

Privilegio EXECUTE para la función de tabla ADMIN_GET_TEMP_TABLES.

Ejemplos

Ejemplo 1: Recuperar información de atributos de tabla y de tiempo de creación de instancias para todas las instancias de la tabla temporal creada DBUSER1.EMPLOYEE para todas las conexiones.

```
SELECT TABSCHEMA, TABNAME, ONCOMMIT, ONROLLBACK, INSTANTIATION_TIME
FROM TABLE (SYSPROC.ADMIN_GET_TEMP_TABLES(NULL, 'DBUSER1', 'EMPLOYEE'))
AS T
```

Ejemplo 2: Recuperar el tiempo de creación de instancias y el ID de espacio de tablas para todas las instancias de tablas temporales de usuario para la conexión actual.

```
SELECT TABSCHEMA, TABNAME, INSTANTIATION_TIME, TBSP_ID
FROM TABLE (SYSPROC.ADMIN_GET_TEMP_TABLES(APPLICATION_ID(), '', ''))
AS T
```

Notas de uso

-
- Si se especifican *esquematabla* y *nombretabla*, sólo se devuelve información para esa tabla temporal específica.
- Si se especifica *esquematabla* pero *nombretabla* es NULL o la serie vacía ("), se devolverá información para todas las tablas del esquema determinado.
- Si *esquematabla* es NULL o la serie vacía (") y se especifica *nombretabla*, se devolverá un error. Para recuperar información para una tabla temporal determinada, tanto el esquema como el nombre de la tabla deberán identificar a la tabla.
- Si *esquematabla* y *nombretabla* son NULL o la serie vacía ("), se devolverá información para todas las tablas temporales para la conexión o todas las conexiones, en función del valor de *descriptorcontexto_aplicación*.
- Si *esquematabla* o *nombretabla* no existen, si *nombretabla* no corresponde a un nombre de tabla temporal o si las instancias de la tabla temporal identificada no existen en la base de datos, se devolverá un conjunto de resultados vacío.

Vista administrativa ADMINTEMPTABLES y metadatos de función de tabla ADMIN_GET_TEMP_TABLES

Tabla 74. Vista administrativa ADMINTEMPTABLES y metadatos de función de tabla ADMIN_GET_TEMP_TABLES

| Nombre de columna | Tipo de datos | Descripción |
|--------------------|---------------|--|
| APPLICATION_HANDLE | BIGINT | ID exclusivo a nivel del sistema para la aplicación. En una base de datos de una sola partición, este identificador consiste en un recuento de 16 bits. En una base de datos de varias particiones, este identificador consiste en el número de partición de coordinación concatenado con un recuento de 16 bits. Además, este identificador es el mismo en cada partición en la que la aplicación puede realizar una conexión secundaria. |
| APPLICATION_NAME | VARCHAR(256) | Nombre de la aplicación. |
| TABSCHEMA | VARCHAR(128) | Nombre de esquema. |
| TABNAME | VARCHAR(128) | Nombre de tabla. |
| INSTANTIATOR | VARCHAR(128) | ID de autorización con el que se ha inicializado la tabla temporal creada o declarado la tabla temporal declarada. |
| INSTANTIATOR_TYPE | CHAR(1) | <ul style="list-style-type: none"> • "U" = El instanciador es un usuario individual |
| TEMPTABTYPE | CHAR(1) | Tipo de tabla temporal: <ul style="list-style-type: none"> • "C" = Tabla temporal creada • "D" = Tabla temporal declarada |
| INSTANTIATION_TIME | TIMESTAMP | Momento en el que se ha inicializado la instancia de tabla temporal creada o declarado la tabla temporal declarada. |
| COLCOUNT | SMALLINT | Número de columnas, incluidas las columnas heredadas (si las hay). |
| TAB_FILE_ID | BIGINT | table_file_id - El ID de archivo (FID) para la tabla. |
| TBSP_ID | BIGINT | tablespace_id - Número entero que representa de forma exclusiva un espacio de tablas utilizado por la base de datos actual. |
| PMAP_ID | SMALLINT | Identificador de la correlación de distribución que utiliza actualmente esta tabla. |
| PARTITION_MODE | CHAR(1) | Indica la forma en que los datos se distribuyen entre particiones de base de datos en un sistema de bases de datos particionadas. <ul style="list-style-type: none"> • "H" = Generación aleatoria • Blanco = No hay particionamiento de base de datos |
| CODEPAGE | SMALLINT | Página de códigos del objeto. Es la página de códigos por omisión utilizada para todas las columnas de caracteres y columnas generadas por expresión. |
| ONCOMMIT | CHAR(1) | Especifica la acción que se realiza sobre esta tabla cuando se ejecuta una operación COMMIT. <ul style="list-style-type: none"> • "D" = Suprimir filas • "P" = Conservar filas |
| ONROLLBACK | CHAR(1) | Especifica la acción que se realiza sobre esta tabla cuando se ejecuta una operación ROLLBACK. <ul style="list-style-type: none"> • "D" = Suprimir filas • "P" = Conservar filas |

Tabla 74. Vista administrativa ADMINTEMPABLES y metadatos de función de tabla ADMIN_GET_TEMP_TABLES (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| LOGGED | CHAR(1) | Especifica si esta tabla está anotada. <ul style="list-style-type: none">• "N" = No anotada• "Y" = Anotada |

Capítulo 5. Vistas y rutinas del planificador de tareas administrativas

Procedimiento ADMIN_TASK_ADD - Planificar una nueva tarea

El procedimiento ADMIN_TASK_ADD planifica una tarea administrativa, que es un trabajo que se puede encapsular dentro de un procedimiento.

Sintaxis

```
►►—ADMIN_TASK_ADD—(—nombre—,—indicación-fecha-hora_inicial—,——————►  
►—indicación-fecha-hora_final—,—máx_invocaciones—,—planificación—,——————►  
►—esquema_procedimiento—,—nombre_procedimiento—,—entrada_procedimiento—,—————►  
►—opciones—,—comentarios—)——————►◄◄
```

El esquema es SYSPROC.

Parámetros del procedimiento

nombre

Un argumento de entrada de tipo VARCHAR(128) que especifica el nombre de la tarea. Este argumento no puede ser NULL.

indicación-fecha-hora_inicial

Argumento de entrada del tipo TIMESTAMP que especifica la hora más temprana que una tarea puede empezar. El valor de este argumento no puede estar en el pasado y no puede ser posterior a *indicación-fecha-hora_final*.

El inicio de la ejecución de una tarea dependerá de cómo se defina este argumento y el argumento *planificación*:

- Si el argumento *indicación-fecha-hora_inicial* no es NULL:
 - Si el argumento *planificación* es NULL, la ejecución de la tarea empieza en *indicación-fecha-hora_inicial*.
 - Si el argumento *planificación* no es NULL, la ejecución de la tarea empieza a la hora planificada siguiente o después de *indicación-fecha-hora_inicial*.
- Si el argumento *indicación-fecha-hora_inicial* es NULL:
 - Si el argumento *planificación* es NULL, la ejecución de la tarea empieza inmediatamente.
 - Si el argumento *planificación* no es NULL, la ejecución de la tarea empieza a la hora planificada siguiente.

indicación-fecha-hora_final

Argumento de entrada de tipo TIMESTAMP que especifica la última vez que una tarea puede empezar su ejecución. El valor de este argumento no puede estar en el pasado y no puede ser anterior a *indicación-fecha-hora_inicial*. Si el argumento es NULL, la tarea puede seguir ejecutándose como planificada de forma indefinida.

Una tarea en ejecución no se interrumpirá en su *indicación-fecha-hora_final*.

máx_invocaciones

Argumento de entrada de tipo INTEGER que especifica el número máximo de ejecuciones permitido para la tarea. Si el argumento es NULL, no hay límite en el número de veces que se puede ejecutar una tarea. Si el argumento es 0, la tarea no se ejecutará.

Este valor se aplica a la planificación si *planificación* no es NULL.

Si se especifica *indicación-fecha-hora_final* y *máx_invocaciones*, tiene prioridad *indicación-fecha-hora_final*. Es decir, si se alcanza la indicación de fecha y hora *indicación-fecha-hora_final*, aunque el número de ejecuciones de tareas realizadas hasta el momento no haya alcanzado el valor de *máx_invocaciones*, la tarea no se ejecutará de nuevo.

planificación

Argumento de entrada de tipo VARCHAR(1024) que especifica una planificación de ejecución de tarea en momentos concretos. Si el argumento es NULL, la tarea no se planifica en momentos concretos.

La serie *planificación* se debe especificar utilizando el formato cron de UNIX.

No se da soporte a varias planificaciones.

esquema_procedimiento

Argumento de entrada de tipo VARCHAR(128) que especifica el esquema del procedimiento que esta tarea ejecutará. Este argumento no puede ser NULL.

nombre_procedimiento

Argumento de entrada de tipo VARCHAR(128) que especifica el nombre del procedimiento que esta tarea ejecutará. Este argumento no puede ser NULL.

entrada_procedimiento

Argumento de entrada de tipo CLOB(2M) que especifica los argumentos de entrada del procedimiento que esta tarea ejecutará. Este argumento debe contener una sentencia SQL que devuelva una fila de datos. Los valores devueltos se pasarán como argumentos al procedimiento. Si este argumento es NULL, no se pasa ningún argumento al procedimiento.

El número de columnas devuelto por la sentencia SQL debe coincidir con el número (y tipo) total de argumentos para el procedimiento y debe contener una sola fila. Para los argumentos de salida, el valor se ignora, pero debe ser del mismo tipo de datos SQL que el procedimiento necesita.

Esta sentencia SQL se ejecuta cada vez que se ejecuta la tarea. Si la sentencia SQL falla, el estado de la tarea se establecerá en NOTRUN y se registrará información específica de SQLCODE. Si la sentencia no devuelve un conjunto de resultados, no devuelve una fila, devuelve varias filas o conjuntos de resultados, la tarea no se ejecutará. El estado de la tarea se establecerá en NOTRUN y se establecerá SQLCODE SQL1465N para indicar que este argumento no es válido.

Si el resultado de la sentencia contiene parámetros XML serializados, el tamaño total de todos los parámetros XML combinados está limitado a 256 kilobytes. Si el resultado excede este umbral, el estado de la tarea se establecerá en NOTRUN. Se establecerán SQLCODE -302 y SQLSTATE 22001 para indicar que se ha producido un truncamiento de datos.

Para ver el estado de la tarea, utilice la vista
SYSTOOL.ADMIN_TASK_STATUS

opciones

Argumento de entrada de tipo VARCHAR(512). Este argumento debe ser NULL.

comentarios

Argumento de entrada de tipo VARCHAR(254) que especifica una descripción de la tarea. Este argumento es opcional y puede ser NULL.

Autorización

Privilegio EXECUTE sobre el procedimiento ADMIN_TASK_ADD. Salvo que la base de datos se haya creado con la opción **RESTRICTIVE**, por omisión el privilegio EXECUTE se otorga a PUBLIC.

Notas de uso

El espacio de tablas SYSTOOLSPACE debe existir antes de llamar al procedimiento ADMIN_TASK_ADD. Si no existe, el procedimiento devolverá un mensaje de error SQL0204N.

Cuando se planifica una tarea, se anota el ID de autorización del usuario de sesión actual. El planificador conmuta a este ID de autorización de sesión cuando se ejecuta la tarea.

El planificador de la tarea administrativa no da soporte a la ejecución de procedimientos que realizan una conexión de base de datos sin un ID de usuario y contraseña especificados. Por ejemplo, el procedimiento ADMIN_CMD se puede utilizar para realizar una operación LOAD desde una base de datos. Una conexión a la base de datos de origen se establece mediante el ID de usuario y la contraseña que se proporcionan para la base de datos conectada actualmente. Este tipo de operación LOAD no la puede ejecutar el planificador de tareas.

Si se pasan argumentos no válidos al procedimiento, se devolverá SQL0171N. Los símbolos del mensaje indicarán qué argumento no es válido y el nombre del procedimiento.

La tarea no se puede planificar para su ejecución hasta que se confirme la unidad de trabajo y el planificador haya captado la definición de tarea.

El planificador comprueba cada 5 minutos si hay tareas nuevas o actualizadas. Para asegurarse de que la tarea se ejecute de la forma prevista, la hora de inicio más temprana, tal como se define en los argumentos *indicación-fecha-hora_inicial*, *indicación-fecha-hora_final* y *planificación*, debe ser como mínimo 5 minutos después de que se confirme la unidad de trabajo.

La base de datos debe ser activa en todas las particiones de base de datos para asegurarse de que el planificador pueda ejecutar la tarea.

En un entorno de base de datos particionadas, el procedimiento ADMIN_TASK_ADD se puede llamar desde cualquier partición de base de datos. Sin embargo, el planificador ejecutará todas las tareas desde la partición de base de datos de catálogo.

Los argumentos *indicación-fecha-hora_inicial*, *indicación-fecha-hora_final* y *planificación* se basan en el huso horario del servidor. Es necesario prestar una especial atención cuando se planifica una tarea durante el periodo de transición del horario de verano. Si la tarea está planificada para ejecutarse a las 2:01 horas y es la época del año en que se adelanta una hora, la tarea no se ejecutará ya que el reloj saltará de las 2:00 horas a las 3:00 horas. Por otra parte, cuando se retrasa una hora, las tareas que se planificaron originalmente entre las 2:00 horas y las 3:00 hora se ejecutarán

dos veces. El usuario es responsable de realizar ajustes para el horario de verano para asegurarse de obtener el comportamiento que se desea.

El planificador siempre se confirmará después de llamar al procedimiento especificado por *esquema_procedimiento* y *nombre_procedimiento*. Si es necesario realizar una retrotracción de transacción, la retrotracción debe producirse dentro del procedimiento.

Si el nombre de tarea no es exclusivo, el procedimiento fallará e emitirá un SQL0601N.

Ejemplo

Ejemplo 1: Crear una tarea que lleve a cabo diariamente una copia de seguridad de TSM en línea a las 00:00 horas, a partir del 4 de febrero de 2008:

```
CALL SYSPROC.ADMIN_TASK_ADD
( 'DAILY TSM BACKUP',
  '2007-02-04-00.00.00.000000',
  NULL,
  NULL,
  '0 0 * * *',
  'SYSPROC',
  'ADMIN_CMD',
  'VALUES(''BACKUP DATABASE SALES ONLINE USE TSM WITHOUT PROMPTING'')',
  NULL,
  NULL )
```

Ejemplo 2: Planificar una tarea para desechar un supervisor de sucesos cada hora:

1. Cree un procedimiento SQL, en el esquema PROD, que deseche un supervisor de sucesos llamado "em":

```
CREATE PROCEDURE FLUSH_EVENT_MONITOR()
SPECIFIC FLUSH_EVENT_MONITOR
LANGUAGE SQL
BEGIN
DECLARE stmt VARCHAR(100) ;
SET stmt = 'FLUSH EVENT MONITOR em' ;
EXECUTE IMMEDIATE stmt ;
END
```

Nota: La sentencia FLUSH EVENT MONITOR SQL no se puede llamar directamente en el procedimiento. No obstante, se puede utilizar EXECUTE IMMEDIATE.

2. Llame a ADMIN_TASK_ADD para planificar la tarea:

```
CALL SYSPROC.ADMIN_TASK_ADD
( 'FLUSH_EVENT_MONITOR EVERY HOUR',
  NULL,
  NULL,
  NULL,
  '0 0-23 * * *',
  'PROD',
  'FLUSH_EVENT_MONITOR',
  NULL,
  NULL,
  NULL )
```

Formato cron de UNIX

El formato cron de UNIX se utiliza para especificar la hora en el parámetro *planificación* de los procedimientos ADMIN_TASK_ADD y ADMIN_TASK_UPDATE.

El formato cron tiene cinco fechas de hora y fecha separados como mínimo por un espacio en blanco. No puede haber un espacio en blanco dentro de un valor de campo. Las tareas planificadas se ejecutan cuando los campos *minuto*, *hora* y *mes del año* coincidan con la fecha y hora actual, y como mínimo uno de los dos campos de día (*mes del año*, o *día de la semana*) coincidan con la fecha actual.

En la tabla 1 se listan los campos de hora y fecha y sus valores permitidos en formato cron.

Tabla 75. Nombres de campo y valores para el formato cron de UNIX

| Nombre de campo | Valores permitidos |
|-------------------------|--|
| <i>minuto</i> | 0-59 |
| <i>hora</i> | 0-23 |
| <i>día del mes</i> | 1-31 |
| <i>mes</i> | <ul style="list-style-type: none"> • 1-12, donde 1 es enero, 2 es febrero y así sucesivamente. • Series de tres caracteres en mayúsculas, minúsculas y una combinación de mayúsculas y minúsculas, basadas en el nombre del mes en inglés. Por ejemplo: jan, feb, mar, apr, may, jun, jul, aug, sep, oct, nov o dec. |
| <i>día de la semana</i> | <ul style="list-style-type: none"> • 0-7, donde 0 o 7 es domingo, 1 es lunes, etc. • Series de tres caracteres en mayúsculas, minúsculas y una combinación de mayúsculas y minúsculas, basadas en el nombre del día en inglés: mon, tue, wed, thu, fri, sat o sun. |

Rangos y listas

Se admiten rangos de números. Los rangos son dos números separados por un guión. El rango especificado es inclusivo. Por ejemplo, el rango 8-11 para una entrada de hora especifica la ejecución a las horas 8, 9, 10 y 11.

Se admiten las listas. Una lista es un conjunto de números o rangos separados por comas. Por ejemplo:

1,2,5,9
0-4,8-12

Rango no restringido

Un campo puede contener un asterisco (*), que representa todos los valores posibles en el campo.

El día de la ejecución de un mandato puede especificarse mediante dos campos: *día del mes* y *día de la semana*. Si los dos campos están restringidos por el uso de un valor distinto al asterisco, el mandato se ejecutará cuando cualquiera de los dos campos coincida con la hora actual. Por ejemplo, el valor 30 4 1,15 * 5 hace que un mandato se ejecute a las 4:30 horas del día 1 y 15 de cada mes, además de cada viernes.

Valores de paso

Los valores de paso se pueden utilizar junto a los rangos. La sintaxis *rango/paso* define el rango y un intervalo de ejecución.

Si especifica *primero-último/paso*, la ejecución tiene lugar en *primero* y en todos los valores sucesivos que distan de *primero* el valor *paso*, hasta llegar a *último*.

Por ejemplo, para especificar la ejecución del mandato cada dos horas, utilice 0-23/2. Esta expresión es equivalente al valor 0,2,4,6,8,10,12,14,16,18,20,22.

Si especifica **/paso*, la ejecución tiene lugar en cada intervalo de *paso* por el rango sin límites. Por ejemplo, como alternativa a 0-23/2 para la ejecución de cada dos horas, utilice */2.

Ejemplo

La tabla 2 lista los valores que puede utilizar para el argumento *planificación* en los procedimientos ADMIN_TASK_ADD o ADMIN_TASK_UPDATE para diversos casos de ejemplo de planificación.

Tabla 76. Planificaciones de tareas de ejemplo y los valores apropiados del argumento *planificación*

| Planificación de tarea que desee | Valor de <i>planificación</i> |
|---|-------------------------------|
| 14:10 horas cada lunes | 10 14 * * 1 |
| Cada día a medianoche | 0 0 * * * |
| Cada día laborable a medianoche | 0 0 * * 1-5 |
| Medianoche entre los días 1 y 15 del mes | 0 0 1,15 * * |
| 18:32 los días 17, 21 y 29 de noviembre, más cada lunes y miércoles de noviembre cada año | 32 18 17,21,29 11 mon,wed |

Vista administrativa ADMIN_TASK_LIST - Recuperar información sobre las tareas en el planificador

La vista administrativa ADMIN_TASK_LIST recupera información sobre cada tarea definida en el planificador de tareas administrativas.

El esquema es SYSTOOLS.

Esta vista se crea la primera vez que se llama al procedimiento ADMIN_TASK_ADD.

Autorización

Privilegio SELECT o CONTROL en la vista administrativa ADMIN_TASK_LIST. Salvo que la base de datos se haya creado con la opción **RESTRICTIVE**, por omisión el privilegio SELECT se otorga a PUBLIC.

Al realizar una consulta en la vista ADMIN_TASK_LIST, sólo devolverá las tareas que se crearon utilizando el ID de autorización de sesión. Si tiene autorización SYSADM, SYSCTRL, SYSMAINT o DBADM, se devuelven todas las tareas.

Ejemplo

Solicitar la lista de tareas del planificador:

```
SELECT * from SYSTOOLS.ADMIN_TASK_LIST
```

Información devuelta

Tabla 77. Información devuelta por la vista administrativa ADMIN_TASK_LIST

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| NAME | VARCHAR(128) | Nombre de la tarea. |
| TASKID | INTEGER | El identificador de la tarea. |
| OWNER | VARCHAR(128) | ID de autorización de sesión del usuario que ha creado la tarea. |
| OWNERTYPE | VARCHAR(1) | El tipo de ID de autorización. Los valores válidos son: <ul style="list-style-type: none">• U - Usuario |
| BEGIN_TIME | TIMESTAMP | La indicación de fecha y hora en que la tarea puede ejecutarse por primera vez. ¹ |
| END_TIME | TIMESTAMP | La indicación de fecha y hora en que la tarea puede ejecutarse por última vez. ¹ Si esta columna es NULL, la tarea puede ejecutarse de forma indefinida, salvo que se especifique MAX_INVOCATIONS. |
| MAX_INVOCATIONS | INTEGER | El número máximo de ejecuciones permitidas para la tarea. Si esta columna es NULL, la tarea puede ejecutarse de forma indefinida, salvo que se especifique END_TIME. |
| SCHEDULE | VARCHAR(1024) | La planificación de la tarea, en formato cron de UNIX. |
| PROCEDURE_SCHEMA | VARCHAR(128) | El esquema del procedimiento que esta tarea ejecutará. |
| PROCEDURE_NAME | VARCHAR(128) | El nombre del procedimiento que esta tarea ejecutará. |
| PROCEDURE_INPUT | CLOB(2M) | Los parámetros de entrada del procedimiento que esta tarea ejecutará. Si esta columna es NULL, no hay parámetros de entrada. |
| OPTIONS | VARCHAR(512) | Opciones que afectan el comportamiento de la tarea. |
| UPDATE_TIME | TIMESTAMP | La indicación de fecha y hora de la última actualización de la tarea. |
| REMARKS | VARCHAR(254) | Una descripción de la tarea. |

Nota:

- ¹ BEGIN_TIME y END_TIME se basan en el huso horario del servidor de base de datos. El usuario es responsable de realizar ajustes para el horario de verano.

Procedimiento ADMIN_TASK_REMOVE - Eliminar registros de tareas planificadas o de estados de tareas

El procedimiento ADMIN_TASK_REMOVE elimina tareas administrativas planificadas que son trabajos que se pueden encapsular dentro de un procedimiento. También elimina registros de estados de tareas.

Sintaxis

```
►►—ADMIN_TASK_REMOVE—(—nombre—,—indicación-fecha-hora_inicial—)—————►►
```

El esquema es SYSPROC.

Parámetros del procedimiento

nombre

Un argumento de entrada de tipo VARCHAR(128) que especifica el nombre de la tarea.

indicación-fecha-hora_final

Un argumento de salida de tipo TIMESTAMP que especifica la indicación de fecha y hora *indicación-fecha-hora_final* del registro de estado.

Autorización

Privilegio EXECUTE en el procedimiento ADMIN_TASK_REMOVE. Salvo que la base de datos se haya creado con la opción **RESTRICTIVE**, por omisión el privilegio EXECUTE se otorga a PUBLIC.

Aunque el ID de autorización de sentencia puede permitir que se ejecute el procedimiento, la eliminación satisfactoria de registros de tareas y estado dependerá del valor del ID de autorización de sesión actual. El ID de autorización de sesión actual debe coincidir con el ID de autorización de sesión que se ha registrado al crear la tarea. Los usuarios con autorización SYSADM, SYSCTRL, SYSMAINT o DBADM pueden eliminar cualquier registro de tarea o estado. Si un usuario no autorizado intenta eliminar un registro de tarea o estado, se devuelve un SQL0551N.

Notas de uso

La tarea no se elimina hasta que no se confirma la unidad de trabajo.

El comportamiento de la eliminación de la tarea dependerá de cómo se definan los argumentos *name* y *indicación-fecha-hora_final*:

- Si el argumento *indicación-fecha-hora_final* es NULL:
 - Si el argumento *name* es NULL, se eliminarán todos los registros de tareas y estados. Si actualmente hay una o más tareas en ejecución, no se eliminarán los registros de tarea y estado asociado. En este caso, se devuelve SQL1464W.
 - Si el argumento *name* no es NULL, se elimina el registro de tarea que coincide con *name*. Si la tarea especificada se está ejecutando actualmente, la tarea no se elimina y se devuelve SQL20453N. Si se elimina la tarea especificada, se eliminarán todos los registros de estado asociados.
- Si el argumento *indicación-fecha-hora_final* no es NULL:

- Si el argumento *name* es NULL, se eliminarán todos los registros de estado con indicaciones de fecha y hora *indicación-fecha-hora_final* anteriores o iguales a *indicación-fecha-hora_final*. No se eliminará ningún registro de tarea. El procedimiento no eliminará ningún registro de estado que tenga el valor de estado RUNNING.
- Si el argumento *name* no es NULL, los registros de estado de la tarea que coincide con *name* se eliminarán si su indicación de fecha y hora *indicación-fecha-hora_final* es anterior o igual a *indicación-fecha-hora_final*. No se eliminará ningún registro de tarea. El procedimiento no eliminará ningún registro de estado que tenga el valor de estado RUNNING.

Si un usuario intenta eliminar una tarea que no existe, se devuelve un SQL0204N.

Ejemplo

Eliminar una tarea de copia de seguridad denominada 'DAILY TSM BACKUP':

```
CALL SYSPROC.ADMIN_TASK_REMOVE('DAILY TSM BACKUP', NULL)
```

Vista administrativa ADMIN_TASK_STATUS - Recuperar información de estado de tarea

La vista administrativa ADMIN_TASK_STATUS recupera información sobre el estado de la ejecución de tareas en el planificador de tareas administrativas.

El esquema es SYSTOOLS.

Esta vista se crea la primera vez que se llama al procedimiento ADMIN_TASK_ADD.

Autorización

Privilegio SELECT o CONTROL en la vista administrativa ADMIN_TASK_STATUS. Salvo que la base de datos se haya creado con la opción **RESTRICTIVE**, por omisión el privilegio SELECT se otorga a PUBLIC.

Al realizar una consulta en la vista ADMIN_TASK_STATUS, sólo devolverá los registros de estado de tarea creados por su ID de autorización de sesión.

Ejemplo

Ejemplo 1: Solicitar el estado de tareas en el planificador:

```
SELECT * from SYSTOOLS.ADMIN_TASK_STATUS
```

Ejemplo 2: Formatear los datos de la columna SQLERRMC utilizando la función SQLERRM:

```
SELECT TASKID, STATUS, SQLCODE, SQLSTATE, RC,
  VARCHAR( SQLERRM( 'SQL' || CHAR( ABS(SQLCODE) ),
    SQLERRMC, x'FF', 'en_US', 1 ), 256) AS MSG_TXT
FROM SYSTOOLS.ADMIN_TASK_STATUS
```

Información devuelta

Tabla 78. Información devuelta por la vista administrativa ADMIN_TASK_STATUS

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| NAME | VARCHAR(128) | Nombre de la tarea. |
| TASKID | INTEGER | El identificador de la tarea. |
| STATUS | VARCHAR(10) | El estado de la tarea. Los valores válidos son: <ul style="list-style-type: none"> • RUNNING - La tarea está actualmente en ejecución. • COMPLETED - La tarea ha finalizado su ejecución. • NOTRUN - Un error ha impedido que el planificador llamara al procedimiento de la tarea. • UNKNOWN - La tarea ha empezado a ejecutarse pero una condición inesperada ha impedido al planificador que registrara el resultado de la tarea. Esto puede ocurrir si el sistema finaliza de forma anormal o se produce una anomalía en la alimentación mientras la tarea está ejecutándose. |
| INVOCATION | INTEGER | La cuenta de invocaciones actual. |
| BEGIN_TIME | TIMESTAMP | La hora en que se ha iniciado la tarea. ¹ Si STATUS es RUNNING, COMPLETED, o UNKNOWN, este valor indica la hora en que la tarea ha empezado a ejecutarse. Si STATUS es NOTRUN, indica la hora en que la tarea debía haberse iniciado. |
| END_TIME | TIMESTAMP | La hora en la que la tarea ha finalizado su ejecución. ¹ Este valor será NULL si STATUS es RUNNING. Si STATUS es UNKNOWN, este valor indica la hora en que el planificador de tareas ha detectado que la tarea ya no se estaba ejecutando y ha actualizado la tabla de estado. |
| AGENT_ID | BIGINT | El ID de agente de la aplicación que ejecuta la tarea. El ID de agente es anónimo con el manejador de aplicaciones. Este valor sólo es válido mientras la tarea está en ejecución. |
| SQLCODE | INTEGER | Si STATUS es COMPLETED, este valor indica el SQLCODE devuelto por CALL al procedimiento. Si STATUS es NOTRUN, este valor indica el SQLCODE del error que ha impedido que se ejecutara la tarea. Si el estado es RUNNING o UNKNOWN, este valor será NULL. |
| SQLSTATE | CHAR(5) | Si STATUS es COMPLETED, este valor indica el SQLSTATE devuelto por CALL al procedimiento. Si STATUS es NOTRUN, este valor indica el SQLSTATE del error que ha impedido que se ejecutara la tarea. Si el estado es RUNNING o UNKNOWN, este valor será NULL. |

Tabla 78. Información devuelta por la vista administrativa ADMIN_TASK_STATUS (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|--------------------------|--|
| SQLERRMC | VARCHAR(70) FOR BIT DATA | <p>Contiene uno o más símbolos, separados por X'FF', tal como aparecen en el campo SQLERRMC del SQLCA. Estos símbolos se sustituyen por variables en las descripciones de condiciones de error</p> <p>Si STATUS es COMPLETED, este valor indica el SQLERRMC devuelto por CALL al procedimiento.</p> <p>Si STATUS es NOTRUN, este valor indica el SQLERRMC del error que ha impedido que se ejecutara la tarea.</p> <p>Si el estado es RUNNING o UNKNOWN, este valor será NULL.</p> |
| RC | INTEGER | Si STATUS es COMPLETED, éste contiene el código de retorno de CALL para el procedimiento si éste tenía un código de retorno. De lo contrario, será NULL. |

- ¹ BEGIN_TIME y END_TIME se basan en el huso horario del servidor de base de datos. El usuario es responsable de realizar ajustes para el horario de verano.

Procedimiento ADMIN_TASK_UPDATE - Actualizar una tarea existente

El procedimiento ADMIN_TASK_UPDATE actualiza una tarea administrativa, que es un trabajo que se puede encapsular dentro de un procedimiento.

Sintaxis

```
►►—ADMIN_TASK_UPDATE—(—nombre—,—indicación-fecha-hora_inicial—,—
►—indicación-fecha-hora_final—,—máx_invocaciones—,—planificación—,—
►—opciones—,—comentarios—)►►
```

El esquema es SYSPROC.

Parámetros del procedimiento

nombre

Un argumento de entrada de tipo VARCHAR(128) que especifica el nombre de una tarea existente. Este argumento no puede ser NULL.

indicación-fecha-hora-inicial

Argumento de entrada del tipo TIMESTAMP que especifica la hora más temprana que una tarea puede empezar. El valor de este argumento no puede estar en el pasado y no puede ser posterior a *indicación-fecha-hora_final*.

El inicio de la ejecución de una tarea dependerá de cómo se defina este parámetro y el parámetro *planificación*:

- Si el argumento *indicación-fecha-hora_inicial* no es NULL:
 - Si el argumento *planificación* es NULL, la ejecución de la tarea empieza en *indicación-fecha-hora_inicial*.
 - Si el argumento *planificación* no es NULL, la ejecución de la tarea empieza a la hora planificada siguiente o después de *indicación-fecha-hora_inicial*.
- Si el argumento *indicación-fecha-hora_inicial* es NULL:

- Si el argumento *planificación* es NULL, la ejecución de la tarea empieza inmediatamente.
- Si el argumento *planificación* no es NULL, la ejecución de la tarea empieza a la hora planificada siguiente.

indicación-fecha-hora_final

Argumento de entrada de tipo `TIMESTAMP` que especifica la última vez que una tarea puede empezar su ejecución. El valor de este argumento no puede estar en el pasado y no puede ser anterior a *indicación-fecha-hora_inicial*. Si el argumento es NULL, la tarea puede seguir ejecutándose como planificada de forma indefinida.

Una tarea en ejecución no se interrumpirá en su *indicación-fecha-hora_final*.

máx_invocaciones

Argumento de entrada de tipo `INTEGER` que especifica el número máximo de ejecuciones permitido para la tarea. Si el argumento es NULL, no hay límite en el número de veces que se puede ejecutar una tarea. Si el argumento es 0, la tarea no se ejecutará.

Este valor se aplica a la planificación si *planificación* no es NULL.

Si se especifica *indicación-fecha-hora_final* y *máx_invocaciones*, tiene prioridad *indicación-fecha-hora_final*. Es decir, si se alcanza la indicación de fecha y hora *indicación-fecha-hora_final*, aunque el número de ejecuciones de tareas realizadas hasta el momento no haya alcanzado el valor de *máx_invocaciones*, la tarea no se ejecutará de nuevo.

planificación

Argumento de entrada de tipo `VARCHAR(1024)` que especifica una planificación de ejecución de tarea en momentos concretos. Si el argumento es NULL, la tarea no se planifica en momentos concretos.

La serie *planificación* se debe especificar utilizando el formato cron de UNIX.

No se da soporte a varias planificaciones.

opciones

Argumento de entrada de tipo `VARCHAR(512)`. Este argumento debe ser NULL.

comentarios

Argumento de entrada de tipo `VARCHAR(254)` que especifica una descripción de la tarea. Es un argumento opcional que se puede establecer en NULL.

Autorización

Privilegio `EXECUTE` en el procedimiento `ADMIN_TASK_UPDATE`. Salvo que la base de datos se haya creado con la opción **RESTRICTIVE**, por omisión el privilegio `EXECUTE` se otorga a `PUBLIC`.

Aunque el ID de autorización de sentencia puede permitir que se ejecute el procedimiento, una tarea no puede actualizarse a menos que el ID de autorización de sesión actual coincida con el ID de autorización de sesión que se registró al crear la tarea. Los usuarios con autorización `SYSADM`, `SYSCTRL`, `SYSMAINT` o `DBADM` pueden actualizar cualquier tarea existente. Si se intenta actualizar una tara que ha añadido un usuario distinto se devolverá `SQL0551N`.

Notas de uso

Si se pasan argumentos no válidos al procedimiento, se devolverá SQL0171N. Los símbolos del mensaje indicarán qué argumento no es válido y el nombre del procedimiento.

Los cambios realizados en la tarea no entran en vigor hasta que se confirme la unidad de trabajo y el planificador haya captado la definición de tarea actualizada. Si deja la unidad de trabajo sin confirmar, se podría impedir o retrasar la ejecución de la tarea existente.

El planificador comprueba cada 5 minutos si hay tareas actualizadas. Para asegurarse de que la tarea se ejecute de la forma prevista, la hora de inicio más temprana, tal como se define en los parámetros *indicación-fecha-hora_inicial*, *indicación-fecha-hora_final* y *planificación*, debe ser como mínimo 5 minutos después de que se confirme la unidad de trabajo.

La base de datos debe ser activa en todas las particiones de base de datos para asegurarse de que el planificador pueda ejecutar la tarea.

Los argumentos *indicación-fecha-hora_inicial*, *indicación-fecha-hora_final* y *planificación* se basan en el huso horario del servidor de base de datos. Es necesario prestar una especial atención cuando se planifica una tarea durante el periodo de transición del horario de verano. Si la tarea está planificada para ejecutarse a las 2:01 horas y es la época del año en que se adelanta una hora, la tarea no se ejecutará ya que el reloj saltará de las 2:00 horas a las 3:00 horas. Por otra parte, cuando se retrasa una hora, las tareas que se planificaron originalmente entre las 2:00 horas y las 3:00 hora se ejecutarán dos veces. El usuario es responsable de realizar ajustes para el horario de verano para asegurarse de obtener el comportamiento que se desea.

Cuando se actualiza una tarea, se restablece el contador de invocaciones interno de la tarea. Como ejemplo, suponga una tarea que se repite con *máx_invocaciones* con un valor 10. Si la tarea se ejecuta 3 veces, habrá 3 registros de estado correspondientes en la salida de ADMIN_TASK_STATUS. Las entradas tienen los valores INVOCATION 1, 2 y 3, respectivamente. Ahora, suponga que el creador de la tarea la actualiza. Esta actualización restablecerá el contador de invocaciones interno. Los registros de estado originales permanecen intactos. Con el tiempo, se crearán nuevos registros de estado con los valores de INVOCATION 1, 2, 3, etc. Se puede utilizar BEGIN_TIME para distinguir entre la ejecución de tarea original y actualizada.

Capítulo 6. Rutinas y procedimientos de auditoría

Procedimiento y función de tabla AUDIT_ARCHIVE - Archivar archivo de registro de comprobación

El procedimiento y la función de tabla AUDIT_ARCHIVE archivan el archivo de registro de comprobación en la base de datos conectada.

Sintaxis

►►—AUDIT_ARCHIVE—(—*directorio*—,—*númparticiónbd*—)—►►

El esquema es SYSPROC.

La sintaxis es la misma para el procedimiento y la función de tabla.

Parámetros del procedimiento y la función de tabla

directorio

Argumento de entrada de tipo VARCHAR (1024) que especifica el directorio en que se grabará(n) el/los archivo(s) de anotaciones cronológicas de auditoría. El directorio debe existir en el servidor y el propietario de la instancia deberá estar disponible para crear archivos en dicho directorio. Si el argumento es nulo o una serie vacía, se utilizará el directorio por omisión.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o NULL ó -2 para un agregado de todas las particiones de base de datos.

Autorización

Ejecutar el privilegio en el procedimiento o tabla de función AUDIT_ARCHIVE.

Ejemplos

Ejemplo 1: Archive el/los registro(s) de comprobación de todas las particiones de base de datos en el directorio por omisión utilizando el procedimiento.

```
CALL SYSPROC.AUDIT_ARCHIVE(NULL, NULL)
```

Ejemplo 2: Archive el/los registro(s) de comprobación de todas las particiones de base de datos en el directorio por omisión utilizando la función de tabla.

```
SELECT * FROM TABLE(SYSPROC.AUDIT_ARCHIVE(' ', -2)) AS T1
```

Información devuelta

Tabla 79. Información devuelta por la función de tabla y el procedimiento AUDIT_ARCHIVE

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| DBPARTITIONNUM | INTEGER | Número de partición del archivo registrado. |

Tabla 79. Información devuelta por la función de tabla y el procedimiento AUDIT_ARCHIVE (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|-----------------------------|---|
| PATH | VARCHAR(1024) | Ubicación del directorio del archivo registrado. |
| FILE | VARCHAR(1024) | Nombre del archivo registrado. |
| SQLCODE | INTEGER | SQLCODE recibido al intentar archivar el archivo. |
| SQLSTATE | VARCHAR(5) | SQLSTATE recibido al intentar archivar el archivo. Si SQLSTATE es NULL, el valor será cero. |
| SQLERRMC | VARCHAR(70) FOR BIT DATA | sqlerrmc recibido al intentar archivar el archivo. Si SQLSTATE es NULL, el valor será cero. |

AUDIT_DELIM_EXTRACT - Realizar extracción en archivo delimitado

El procedimiento almacenado AUDIT_DELIM_EXTRACT realiza una extracción a un archivo delimitado en archivos de comprobación archivados de la base de datos conectada. En especial, a aquellos archivos de comprobación archivados con nombres de archivo que coinciden con el patrón de máscara especificado.

Sintaxis

```
►►—AUDIT_DELIM_EXTRACT—(—delimitador—,—directorio_destino—,—directorio_origen—,——————►
►—máscara_archivo—,—opciones_evento—)—————►►
```

El esquema es SYSPROC.

Parámetros del procedimiento

delimitador

Argumento de entrada de tipo VARCHAR(1) opcional que especifica el delimitador de caracteres que se debe utilizar en archivos delimitados. Si el argumento es nulo o una serie vacía, se utilizarán comillas dobles como delimitador.

directorio_destino

Argumento de entrada de tipo VARCHAR(1024) opcional que especifica el directorio donde se almacenarán los archivos delimitados. Si el argumento es nulo o una serie vacía, se utilizará el mismo directorio que el *directorio_origen*.

directorio_origen

Argumento de entrada de tipo VARCHAR(1024) opcional que especifica el directorio donde se almacenan los archivos de anotaciones cronológicas de auditoría archivados. Si el argumento es nulo o una serie vacía, se utilizará la comprobación por omisión.

máscara_archivo

Un argumento de entrada de tipo VARCHAR(1024) opcional es una máscara de la que los archivos pueden extraer. Si el argumento es nulo o la serie está vacía, extraerá de todos los archivos de anotaciones cronológicas de auditoría del directorio de origen.

opciones_evento

Un argumento de entrada de tipo VARCHAR(1024) opcional que especifica la serie que define los eventos que deben extraerse. Esto concuerda con la misma serie del programa de utilidad db2audit. Si el argumento es nulo o una serie vacía, extraerá todos los eventos.

Autorización

Privilegios de ejecución en las funciones AUDIT_DELIM_EXTRACT y AUDIT_LIST_LOGS.

Ejemplos

Nota: Los archivos de anotaciones cronológicas de auditoría contienen una indicación de fecha y hora como parte de su convenio de denominación.

Ejemplo 1: Realiza una extracción delimitada en todos los archivos de anotaciones cronológicas de auditoría archivados el 18 de junio de 2007 en el directorio de archivo por omisión. Este ejemplo sólo extrae eventos de ejecución, utilizando el delimitador de caracteres comillas dobles (") y creando o añadiendo los archivos de extracción resultantes (< categoría >.del) en el directorio \$HOME/audit_delim_extract.

```
CALL SYSPROC.AUDIT_DELIM_EXTRACT(NULL, '$HOME/AUDIT_DELIM_EXTRACT', NULL, '%20070618%', 'CATEGORIES EXECUTE STATUS BOTH')
```

Función de tabla AUDIT_LIST_LOGS - Enumera los archivos de anotaciones cronológicas de auditoría archivados

La función de tabla AUDIT_LIST_LOGS enumera los archivos de registro de comprobación archivados de una base de datos presentes en un directorio específico.

Sintaxis

►►—AUDIT_LIST_LOGS—(—*directorio*—)——►►

El esquema es SYSPROC.

Parámetros del procedimiento

directorio

Argumento de entrada de tipo VARCHAR(1024) opcional que especifica el directorio en que se grabará(n) el/los archivo(s) de comprobación archivados. El directorio debe existir en el servidor y el propietario de la instancia deberá estar disponible para crear archivos en dicho directorio. Si el argumento es nulo o una serie vacía, se utilizará el directorio de búsqueda por omisión.

Autorización

Privilegio EXECUTE en la función de tabla AUDIT_LIST_LOGS.

Ejemplos

Ejemplo 1: Enumera todos los registros de comprobación archivados en el directorio de archivos de comprobación por omisión:

```
SELECT * FROM TABLE(SYSPROC.AUDIT_LIST_LOGS('')) AS T1
```

Nota: Aquí solo se enumeran los registros del directorio para la base de datos en que se ejecuta la consulta. Los archivos archivados tienen el formato db2audit.db.<nombrebd>.log.<indicación horaria>

Información devuelta

Tabla 80. La información devuelta para AUDIT_LIST_LOGS

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| PATH | VARCHAR(1024) | Ubicación de la vía de acceso del archivo registrado. |
| FILE | VARCHAR(1024) | Nombre del archivo registrado. |
| SIZE | BIGINT | Tamaño del archivo registrado. |

Capítulo 7. Rutinas de mantenimiento automático

Procedimiento AUTOMAINT_GET_POLICY - recuperar la política de mantenimiento automático

El procedimiento almacenado del sistema AUTOMAINT_GET_POLICY recupera la configuración de mantenimiento automático para la base de datos. Este procedimiento tiene dos parámetros: el tipo de mantenimiento automático con el que reunir información y un puntero en un BLOB en el que devolver la información de configuración. La información de configuración se devuelve en formato XML.

Sintaxis

►►—AUTOMAINT_GET_POLICY—(—*tipo_política*—,—*política*—)—————►►

El esquema es SYSPROC.

Parámetros del procedimiento

tipo_política

Argumento de entrada de tipo VARCHAR(128) que especifica el tipo de política de mantenimiento automático que debe recuperarse. El valor puede ser uno de los siguientes:

AUTO_BACKUP

copia de seguridad automática

AUTO_REORG

reorganización automática de tablas e índices

AUTO_RUNSTATS

operaciones runstats de tabla automáticas

MAINTENANCE_WINDOW

ventana de mantenimiento

política

Un argumento de salida de tipo BLOB(2M) que especifica los valores de mantenimiento automático para el tipo de política dado, en formato XML.

Autorización

Privilegio EXECUTE en el procedimiento AUTOMAINT_GET_POLICY.

Ejemplo

Aquí presentamos un ejemplo de una llamada del procedimiento AUTOMAINT_GET_POLICY desde el código fuente SQL C incorporado.

- Se declara una variable BLOB para el parámetro de salida del procedimiento.
- El procedimiento se llama especificando la copia de seguridad automática como el tipo de política de mantenimiento automático y la variable BLOB como el parámetro de salida en que el procedimiento devolverá la política de copia de seguridad para la base de datos conectada actualmente.

```
EXEC SQL BEGIN DECLARE SECTION;
SQL TYPE IS BLOB(2M) backupPolicy;
EXEC SQL END DECLARE SECTION;

EXEC SQL CALL AUTOMAINT_GET_POLICY( 'AUTO_BACKUP', :backupPolicy );
```

Procedimiento AUTOMAINT_GET_POLICYFILE - recuperar la política de mantenimiento automático

El procedimiento almacenado del sistema AUTOMAINT_GET_POLICYFILE recupera la configuración de mantenimiento automático para la base de datos. Este procedimiento tiene dos parámetros: el tipo de mantenimiento automático con el que reunir información y el nombre de un archivo en el que devolver la información de configuración. La información de configuración se devuelve en formato XML.

Sintaxis

```
►►—AUTOMAINT_GET_POLICYFILE—(—tipo_política—,—política—)—————►►
```

El esquema es SYSPROC.

Parámetros del procedimiento

tipo_política

Argumento de entrada de tipo VARCHAR(128) que especifica el tipo de política de mantenimiento automático que debe recuperarse. El valor puede ser uno de los siguientes:

AUTO_BACKUP

copia de seguridad automática

AUTO_REORG

reorganización automática de tablas e índices

AUTO_RUNSTATS

operaciones runstats de tabla automáticas

MAINTENANCE_WINDOW

ventana de mantenimiento

nombre_archivo_política

Argumento de entrada de tipo VARCHAR(2048) que especifica el nombre del archivo creado en el subdirectorio tmp del directorio de instancias de DB2.

Nota: Es posible que el nombre del archivo tenga un prefijo con una vía de acceso relacionada con tmp. En ese caso, el directorio debe existir, debe tener permiso para crear/sobregabar el archivo y debe utilizarse el separador de vía de acceso correcto para el DB2 Server.

Por ejemplo:

En UNIX, si el directorio de instancias se ha definido como \$HOME/sqllib. Para un archivo de política denominado 'policy.xml', el nombre de archivo será '\$HOME/sqllib/tmp/policy.xml'

En Windows, el nombre del directorio de instancias puede determinarse a partir de los valores de la variable de registro **DB2INSTPROF** y la variable de entorno **DB2INSTANCE**. Para un archivo de políticas denominado

'policy.xml', si db2set da DB2INSTPROF=C:\DB2PROF y %DB2INSTANCE%=db2, el nombre de archivo será C:\DB2PROF\db2\tmp\policy.xml

Autorización

Privilegio EXECUTE en el procedimiento AUTOMAINT_GET_POLICYFILE.

Ejemplo

Para obtener los valores actuales del mantenimiento automático para operaciones de copia de seguridad:

```
llame sysproc.automaint_get_policyfile( 'AUTO_BACKUP', 'AutoBackup.xml' )
```

De esta manera, se creará un archivo XML denominado AutoBackup.xml en el subdirectorio tmp bajo el directorio de instancias de DB2.

Procedimiento AUTOMAINT_SET_POLICY - Configurar política de mantenimiento automático

Puede utilizar el procedimiento almacenado del sistema AUTOMAINT_SET_POLICY para configurar el mantenimiento automático para la base de datos. Este procedimiento tiene dos parámetros: el tipo de mantenimiento automático para configurar y un BLOB con XML que especifica la configuración.

Para habilitar la reclamación de extensiones durante las operaciones de reorganización automática en tablas de clúster multidimensional (MDC), es necesario especificar el atributo "reclaimExtentsSizeForMDCTables" en el elemento ReorgOptions, con un valor de umbral en los archivos de entrada XML. Este valor de umbral es el tamaño mínimo en kilobytes de espacio libre dentro de la tabla que puede reclamarse. Este valor debe ser 0 o superior. Por ejemplo, si se especifica un valor de 1024 KB para el umbral, sólo las tablas con 1 MB o más de espacio libre se tendrán en cuenta para la reorganización automática de reclamación de extensiones.

Sintaxis

►►—AUTOMAINT_SET_POLICY—(—*tipo_política*—,—*política*—)—————►

El esquema es SYSPROC.

Parámetros de la función de tabla

tipo_política

Argumento de entrada de tipo VARCHAR(128) que especifica el tipo de política de mantenimiento automático que debe configurarse. El valor puede ser uno de los siguientes:

AUTO_BACKUP

copia de seguridad automática

AUTO_REORG

reorganización automática de tablas e índices

AUTO_RUNSTATS

operaciones runstats de tabla automáticas

MAINTENANCE_WINDOW ventana de mantenimiento

política

Un argumento de entrada de tipo BLOB(2M) que especifica la política de mantenimiento automático en formato XML.

Autorización

Privilegio EXECUTE en el procedimiento SYSPROC.AUTOMAINT_SET_POLICY.

Ejemplos

Ejemplo 1: Para establecer los valores actuales del mantenimiento automático para operaciones runstats:

```
CALL SYSPROC.AUTOMAINT_SET_POLICY
( 'AUTO_RUNSTATS',
  BLOB(' <?xml version="1.0" encoding="UTF-8"?>
    <DB2AutoRunstatsPolicy
      xmlns="http://www.ibm.com/xmlns/prod/db2/autonomic/config">
      <RunstatsTableScope><FilterCondition/></RunstatsTableScope>
    </DB2AutoRunstatsPolicy>')
)
```

De esta manera se sustituirá la configuración actual de colección de estadísticas automática por la nueva configuración contenida en el documento XML que se ha pasado como segundo parámetro al procedimiento."

Ejemplo 2: La función de reorganización automática de DB2 puede usar la nueva opción "RECLAIM EXTENTS ONLY" para reorganizar las tablas de clúster multidimensional (MDC). Para habilitar esta función, establezca el valor de "reclaimExtentsSizeForMDCTables" en la política AUTO_REORG:

```
CALL SYSPROC.AUTOMAINT_SET_POLICY
( 'AUTO_REORG',
  BLOB(' <?xml version="1.0" encoding="UTF-8"?>
    <DB2AutoReorgPolicy
      xmlns="http://www.ibm.com/xmlns/prod/db2/autonomic/config">
      <ReorgOptions dictionaryOption="Keep" indexReorgMode="Online"
        useSystemTempTableSpace="false" reclaimExtentsSizeForMDCTables ="1024" >
      <ReorgTableScope>
        <FilterClause>TABSCHEMA NOT LIKE 'EMP% '</FilterClause>
      </ReorgTableScope>
    </DB2AutoReorgPolicy>')
)
```

Existen archivos de entrada XML de ejemplo en el directorio SQLLIB/samples/automainctcg que puede modificar para cumplir sus requisitos y pasar el contenido XML mediante la función escalar BLOB(), tal como se muestra en el ejemplo.

Procedimiento AUTOMAINT_SET_POLICYFILE - configurar la política de mantenimiento automático

Puede utilizar el procedimiento almacenado del sistema AUTOMAINT_SET_POLICYFILE para configurar el mantenimiento automático para la base de datos. Este procedimiento tiene dos parámetros: el tipo de mantenimiento automático para configurar y el nombre de un documento XML que especifica la configuración.

Este procedimiento devuelve el éxito de SQL o el código de error de SQL.

Sintaxis

►—AUTOMAINT_SET_POLICYFILE—(—*tipo_política*—,—*política*—)—————►

El esquema es SYSPROC.

Parámetros de la función de tabla

tipo_política

Argumento de entrada de tipo VARCHAR(128) que especifica el tipo de política de mantenimiento automático que debe configurarse. El valor puede ser uno de los siguientes:

AUTO_BACKUP

copia de seguridad automática

AUTO_REORG

reorganización automática de tablas e índices

AUTO_RUNSTATS

operaciones runstats de tabla automáticas

MAINTENANCE_WINDOW

ventana de mantenimiento

nombre_archivo_política

Argumento de entrada de tipo VARCHAR(2048) que especifica el nombre del archivo disponible en el subdirectorío tmp del directorío de instancias de DB2.

Nota: Cuando el nombre de archivo se especifica con una vía de acceso relativa, deberá utilizarse el separador de la vía de acceso para el DB2 Server y el directorío y el archivo deberían tener permiso de lectura.

Por ejemplo:

En UNIX, si el directorío de instancias se ha definido como \$HOME/sqllib. Para un archivo de política denominado 'automaint/policy.xml', el nombre de archivo será '\$HOME/sqllib/tmp/automaint/policy.xml'

En Windows, el nombre del directorío de instancias puede determinarse a partir de los valores de la variable de registro **DB2INSTPROF** y la variable de entorno **DB2INSTANCE**. Para un archivo de políticas denominado 'automaint\policy.xml', si db2set da DB2INSTPROF=C:\DB2PROF y %DB2INSTANCE%=db2, el nombre de archivo será C:\DB2PROF\db2\tmp\automaint\policy.xml

Autorización

Privilegio EXECUTE en el procedimiento SYSPROC.AUTOMAINT_SET_POLICYFILE.

Ejemplo

Para modificar los valores actuales del mantenimiento automático para la copia de seguridad automática:

```
llame sysproc.automaint_set_policyfile( 'AUTO_BACKUP', 'AutoBackup.xml' )
```

De esta manera se sustituirán los valores actuales de configuración de copia de seguridad automática por la nueva configuración del archivo `AutoBackup.xml` ubicado en el directorio `tmp` bajo el directorio de instancias de DB2.

Existen archivos de entrada XML de ejemplo en el directorio `SQLLIB/samples/automaintcfg` que pueden utilizarse como referencia para crear un archivo xml de política.

Capítulo 8. Procedimientos de la API de SQL común

La API de SQL común proporciona una recopilación de procedimientos almacenados de firma estable y de firma común que pueden llevarse a distintos servidores de datos de IBM®. Puede utilizar estos procedimientos almacenados para crear aplicaciones que realicen diversas funciones administrativas comunes, como obtener y fijar los parámetros de configuración u obtener información sobre el sistema.

Los procedimientos almacenados proporcionan parámetros XML con sintáctica idéntica y manejo de errores en todos los servidores de datos para garantizar la independencia de la versión del servidor de datos. La estandarización y la estabilidad de las firmas se alcanzan utilizando documentos XML sencillos (con una DTD común) como parámetros. Las diferencias de versión, plataforma y tecnología se expresan mediante distintos pares de valor de clave en listas jerárquicas de propiedad.

Parámetros de entrada y salida comunes

Los procedimientos almacenados de la API de SQL común comparten un conjunto de parámetros de entrada y salida.

La tabla siguiente ofrece una descripción breve de estos parámetros. Para obtener información más detallada, refiérase a los temas de consulta sobre los procedimientos almacenados de la API de SQL común.

Tabla 81. Parámetros de entrada y salida compartidos de la API de SQL común

| Parámetro | Descripción |
|---------------------------|--|
| <i>versión-principal</i> | Indica la versión principal del tipo de documento a la que da soporte el emisor de la llamada para los documentos XML pasados como parámetros en el procedimiento. |
| <i>versión-secundaria</i> | <p>Indica la versión secundaria del tipo de documento a la que da soporte el emisor de la llamada para los documentos XML pasados como parámetros en el procedimiento.</p> <p>Los parámetros <i>versión-principal</i> y <i>versión-secundaria</i> se utilizan juntos para garantizar que el emisor de la llamada no emplea un documento de entrada XML con versión incorrecta. El procedimiento procesa todos los documentos XML en la <i>versión-principal</i> y la <i>versión-secundaria</i> especificadas y devuelve un error si una de las versiones no es válida. Este diseño posibilita la extensión en releases futuros porque pueden agregarse versiones de tipo de documento nuevas sin influir en las aplicaciones existentes.</p> |

Tabla 81. Parámetros de entrada y salida compartidos de la API de SQL común (continuación)

| Parámetro | Descripción |
|---------------------------------|---|
| <i>entorno-local-solicitado</i> | Especifica el entorno local que se utilizará para devolver el contenido traducido de los documentos XML devueltos en los parámetros <i>salida-xml</i> y <i>mensaje-xml</i> . Sólo se traducen los valores, no los nombres de las claves. |
| <i>entrada-xml</i> | Especifica un documento de entrada XML que contiene valores de entrada para el procedimiento. |
| <i>filtro-xml</i> | Especifica una serie de consulta XPath válida que se utiliza para recuperar un único valor de un documento de parámetros de salida. |
| <i>salida-xml</i> | Devuelve un documento de salida XML completo codificado en UTF-8. Según el procedimiento al que se esté llamando, este documento puede contener parámetros de configuración y sus valores, información del sistema o textos de mensaje. Cuando el procedimiento funciona en la <i>modalidad completa</i> , este parámetro devuelve un documento XML que se puede modificar y pasar de vuelta al procedimiento como parámetro <i>entrada-xml</i> . Este método ofrece un enfoque programático para la creación de documentos de entrada XML válidos. |
| <i>mensaje-xml</i> | Devuelve un documento de salida XML completo del tipo de mensaje de servidor de datos en codificación UTF-8 que ofrece información detallada sobre una condición de aviso de SQL. |

Creación de versiones de documentos XML

Para permitir la extensión en releases futuros, los procedimientos almacenados de la API de SQL común devuelven documentos de salida XML que incluyen información sobre la versión.

Siempre que cambia la estructura de un documento de salida XML (por ejemplo, cuando se añade o suprime un elemento), los niveles de la versión se incrementan. Por consiguiente, un procedimiento puede dar soporte a distintas versiones del documento de salida XML.

La información de versión del documento XML se expresa como pares de valor-clave para la versión principal y la versión secundaria del tipo de documento. Por ejemplo, un documento de salida XML puede definir las claves y los valores siguientes en un elemento de diccionario:

```
<key>Document Type Name</key><string>Data Server Configuration Output</string>
<key>Document Type Major Version</key><integer>2</integer>
<key>Document Type Minor Version</key><integer>0</integer>
```

Cuando se llama al procedimiento, se especifica la versión principal y la versión secundaria del documento XML que se desea devolver. El contenido del documento de salida XML variará en función de los valores especificados.

Por ejemplo, el procedimiento GET_CONFIG recupera la base de datos y los parámetros de configuración del gestor de la base de datos que se han determinado para una instancia concreta. Cuando se llama a este procedimiento con *versión-principal* 2 y *versión-secundaria* 0, devuelve un documento XML que contiene los parámetros de configuración agrupados en categorías. Sin embargo, cuando se llama al mismo procedimiento con *versión-principal* 1 y *versión-secundaria* 0, devuelve un documento XML que contiene los parámetros de configuración, pero no están agrupados en categorías.

Del mismo modo, el procedimiento GET_MESSAGE recupera el texto de mensaje y SQLSTATE para un SQLCODE especificado. Cuando se llama a este procedimiento con *versión-principal* 2 y *versión-secundaria* 0, devuelve un documento XML que contiene el mensaje de texto corto, el mensaje de texto largo y el SQLSTATE para el SQLCODE correspondiente. No obstante, cuando se llama al mismo procedimiento con *versión-principal* 1 y *versión-secundaria* 0, devuelve un documento XML que contiene únicamente el mensaje de texto corto y el SQLSTATE. El mensaje de texto largo no está disponible en la versión 1 del documento.

Para determinar las versiones de documento más altas que soporta un procedimiento, especifique NULL para *versión-principal*, *versión-secundaria* y el resto de parámetros de entrada. El procedimiento devuelve las versiones de documento más altas soportadas como valores de los parámetros de salida *versión-principal* y *versión-secundaria*, y fija los parámetros de salida *salida-xml* y *mensaje-xml* en NULL.

Si especifica valores no nulos para *versión-principal* y *versión-secundaria*, debe especificar versiones de documento soportadas, o el procedimiento emitirá un error (-20457) para indicar que el procedimiento ha encontrado una versión no soportada.

Los documentos de entrada XML puede incluir opcionalmente valores para la versión principal y la versión secundaria del tipo de documento. Si estos valores están especificados en el documento de entrada XML, los valores pasados para *versión-principal* y *versión-secundaria* en la llamada del procedimiento deben coincidir con los valores especificados en el documento XML, o el procedimiento emitirá un error (+20458). Este comportamiento garantiza que el emisor de la llamada no especifique una versión no soportada del documento de entrada XML.

Documentos de entrada XML

Los documentos XML que se pasan como entrada a los procedimientos almacenados de la API de SQL común comparten un sencillo formato basado en una DTD común.

El documento de entrada XML está formado por un conjunto de entradas comunes a todos los procedimientos almacenados y por un conjunto de entradas específicas de cada procedimiento almacenado. El documento de entrada XML cuenta con la estructura general siguiente:

```
<?xml version="1.0" encoding="UTF-8"?>
<plist version="1.0">
<dict>
  <key>Document Type Name</key><string>Data Server Message Input</string>
  <key>Document Type Major Version</key><integer>1</integer>
  <key>Document Type Minor Version</key><integer>0</integer>
```

```

    <key>Document Locale</key><string>en_US</string>
    <key>Complete</key><false/>
    <dict>
      <!-- Aquí se muestran los datos específicos del tipo de documento. -->
    </dict>
  </dict>
</plist>

```

Importante: Los documentos de entrada XML deben tener codificación UTF-8 y contener caracteres del inglés únicamente.

Modalidad completa para la devolución de documentos de entrada XML válidos

Puede utilizar la *modalidad completa* para crear un documento XML válido para cualquier procedimiento almacenado de la API de SQL común que acepte entradas. A continuación, puede personalizar el documento y pasarlo de vuelta al procedimiento.

Para ejecutar un procedimiento en la modalidad completa, especifique "true" en la clave Complete del documento XML de entrada y traslade el contenido mínimo siguiente:

```

<?xml version="1.0" encoding="UTF-8"?>
<plist version="1.0">
<dict>
  <key>Complete</key><true/>
</dict>
</plist>

```

Se ignorarán todos los elementos XML que no sean obligatorios y no se devolverán en el documento de salida.

Al ejecutar el procedimiento, se devuelve un documento de entrada XML completo en el parámetro *salida-xml* del procedimiento almacenado. El documento XML devuelto incluye un tipo de documento y una sección para todos los posibles parámetros obligatorios y opcionales. El documento XML devuelto incluye también otras entradas (como nombres de visualización, sugerencias y el entorno local del documento) que no son obligatorios, pero que suelen necesitarse para representar el documento en una aplicación cliente.

Tras representar el documento XML y modificarlo de forma independiente con respecto a la plataforma, puede ejecutar el mismo procedimiento almacenado y trasladarlo al documento XML modificado como entrada.

Documentos de salida XML

Los documentos XML que se devuelven como salida desde los procedimientos almacenados de la API de SQL común comparten un mismo conjunto de entradas.

Como mínimo, los documentos XML devueltos en el parámetro *salida-xml* incluyen los pares de valor de clave obligatorios siguientes:

```

<?xml version="1.0" encoding="UTF-8"?>
<plist version="1.0">
<dict>
  <key>Document Type Name</key>
  <string>Salida de la configuración del servidor de datos</string>
  <key>Document Type Major Version</key><integer>1</integer>
  <key>Document Type Minor Version</key><integer>0</integer>
  <key>Data Server Product Name</key><string>DSN</string>

```

```

<key>Data Server Product Version</key><string>8.1.0.356</string>
<key>Data Server Major Version</key><integer>8</integer>
<key>Data Server Minor Version</key><integer>1</integer>
<key>Data Server Platform</key><string>z/OS</string>
<key>Document Locale</key><string>en_US</string>

<!-- Aquí se muestran los datos específicos del tipo de documento. -->
</dict>
</plist>

```

Las entradas del documento de salida XML podrían agruparse mediante diccionarios anidados. Cada entrada del documento de salida XML describe un único componente de información. La entrada está formada por el valor, el nombre de visualización y una sugerencia. Opcionalmente, puede proporcionarse una unidad de visualización. El nombre de visualización, la sugerencia y la unidad de visualización son sensibles al idioma y se traducirán al idioma especificado en el valor del parámetro *entorno-local-solicitado* (o al idioma por omisión si todavía no se ofrece soporte para el entorno local solicitado). En general, la estructura de las entradas es similar a la siguiente:

```

<key>Real Storage Size</key>
<dict>
  <key>Display Name</key><string>Tamaño de almacenamiento real</string>
  <key>Value</key><integer>2048</integer>
  <key>Display Unit</key><string>MB</string>
  <key>Hint</key><string>Tamaño del almacenamiento real en línea</string>
</dict>

```

Los servidores de datos de IBM disponen de un documento de parámetros comunes que incluye algunas palabras clave aplicables a todos los servidores de datos y algunas palabras clave específicas del servidor de datos. Siempre que un servidor de datos agrega o suprime una palabra clave, el número de versión (de todos los servidores de datos) aumenta. En función del cambio producido, el número de la versión principal puede aumentar y el número de la versión secundaria fijarse en 0, o también es posible que sólo aumente el número de la versión secundaria.

Los documentos de salida XML se generan con codificación UTF-8 y contienen caracteres del inglés únicamente.

Expresiones XPath para el filtrado de salidas

Puede utilizar una expresión XPath para filtrar la salida XML devuelta por un procedimiento almacenado de la API de SQL común.

Para filtrar la salida, especifique una serie de consulta XPath válida en el parámetro *filtro-xml* del procedimiento. Las siguientes restricciones se aplican a la expresión XPath especificada:

- La expresión XPath debe hacer referencia a un solo valor.
- La expresión XPath siempre debe ser absoluta a partir del nodo raíz. Por ejemplo, están permitidas las expresiones de vía de acceso siguientes: */*, *nombre-nodo*, *.* y *...*. Las expresiones siguientes no están permitidas: *//* y *@*.
- Los únicos predicados permitidos son [*vía-acceso*='valor'] y [*n*].
- El único eje permitido es *following-sibling*.
- La expresión XPath debe acabar con uno de los elementos siguientes y, en caso necesario, se le debe añadir el predicado [1]: *following-sibling::string*, *following-sibling::data*, *following-sibling::date*, *following-sibling::real* o *following-sibling::integer*.

- A menos que el eje se encuentre al final de la expresión XPath, debe ir seguida de `::dict`, `::string`, `::data`, `::date`, `::real` o `::integer` y, en caso necesario, se le debe añadir el predicado [1].
- El único operador XPath soportado es `=`.
- La expresión XPath no puede contener funciones, espacios de nombres, instrucciones de proceso o comentarios.

Consejo: Si el procedimiento almacenado funciona con *modalidad completa*, no aplique filtros o se emitirá un SQLCODE (+20458).

Para obtener un control más adecuado sobre el proceso del documento XML devuelto en el parámetro *salida-xml*, puede utilizar la función XMLPARSE disponible con DB2 pureXML.

Ejemplo

La expresión XPath siguiente selecciona el valor para la clave de versión de producto del servidor de datos de un documento de salida XML:

```
/plist/dict/key[.='Data Server Product Version']following-sibling::string[1]
```

El procedimiento devuelve la serie 8.1.0.356 en el parámetro *salida-xml*. Por lo tanto, la llamada al procedimiento devuelve un valor único, en lugar de un documento XML.

Documentos de mensaje XML

Cuando un procedimiento almacenado de la API de SQL común encuentra un error de proceso interno o un parámetro no válido, el servidor de datos devuelve al emisor de la llamada un SQLCODE y el mensaje de SQL correspondiente. En estos casos, el procedimiento devuelve un documento de mensaje XML en el parámetro *mensaje-xml* que contiene información más detallada sobre la situación de aviso.

El documento de mensaje XML cuenta con la estructura general siguiente:

```
<?xml version="1.0" encoding="UTF-8"?>
<plist version="1.0">
<dict>
  <key>Document Type Name</key><string>Mensaje del servidor de datos</string>
  <key>Document Type Major Version</key><integer>1</integer>
  <key>Document Type Minor Version</key><integer>0</integer>
  <key>Data Server Product Name</key><string>QDB2/AIX64</string>
  <key>Data Server Product Version</key><string>9.5.0.3</string>
  <key>Data Server Major Version</key><integer>9</integer>
  <key>Data Server Minor Version</key><integer>5</integer>
  <key>Data Server Platform</key><string>AIX 64BIT</string>
  <key>Document Locale</key><string>en_US</string>
  <key>Short Message Text</key>
  <dict>
    <key>Value</key><string>
      <!-- Aquí se muestra la descripción adicional del aviso. --></string>
    <key>Hint</key><string></string>
  </dict>
</dict>
</plist>
```

Los documentos de mensaje XML se generan con codificación UTF-8 y contienen caracteres del inglés únicamente.

Ejemplo

En el ejemplo siguiente, una llamada al procedimiento GET_MESSAGE produce un aviso de SQL:

```
db2 "CALL SYSPROC.GET_MESSAGE(NULL,NULL,'en_US',NULL,NULL,?,?)"
SQL20458W El procedimiento "SYSPROC.GET_MESSAGE" ha encontrado
un error interno de proceso de parámetros en el parámetro "3".
El valor del parámetro "7" contiene más información sobre
el error. SQLSTATE=01H54
```

El documento XML que se devuelve en el parámetro 7 (*mensaje-xml*) incluye el contenido siguiente:

```
<?xml version="1.0" encoding="UTF-8"?>
<plist version="1.0">
<dict>
  <key>Document Type Name</key><string>Mensaje del servidor de datos</string>
  <key>Document Type Major Version</key><integer>1</integer>
  <key>Document Type Minor Version</key><integer>0</integer>
  <key>Data Server Product Name</key><string>QDB2/AIX64</string>
  <key>Data Server Product Version</key><string>9.5.0.3</string>
  <key>Data Server Major Version</key><integer>9</integer>
  <key>Data Server Minor Version</key><integer>5</integer>
  <key>Data Server Platform</key><string>AIX 64BIT</string>
  <key>Document Locale</key><string>en_US</string>
  <key>Short Message Text</key>
  <dict>
    <key>Value</key><string>Si los parámetros 1 y 2 se establecen en NULL, todos
    los demás parámetros de entrada deberán establecerse también en NULL, pero
    el valor del parámetro "3" no es NULL. </string>
    <key>Hint</key><string></string>
  </dict>
</dict>
</plist>
```

El valor de la clave de texto de mensaje corto ofrece información adicional sobre el aviso.

Procedimiento CANCEL_WORK - Cancelar trabajo

El procedimiento almacenado CANCEL_WORK cancela una actividad específica (por ejemplo, una sentencia de SQL) o todas las actividades de una aplicación conectada.

Para cancelar una actividad específica, deberá pasar un descriptor de contexto de aplicación, una ID de unidad de trabajo y un ID de actividad de la actividad que desea cancelar. Para cancelar todas las actividades de una aplicación conectada, deberá pasar el descriptor de contexto de aplicación. Se retrotraerán todos los cambios asociados a la actividad cancelada.

Sintaxis

```
►► CANCEL_WORK (—versión-principal—, —versión-secundaria—, —entorno-local-solicitado—, —►►
► entrada-xml—, —filtro-xml—, —salida-xml—, —mensaje-xml—) ◀◀
```

El esquema es SYSPROC.

Parámetros del procedimiento

versión-principal

Argumento de entrada y salida de tipo INTEGER que indica la versión principal del documento. En la entrada, este argumento indica la versión principal del documento a la que da soporte el emisor de la llamada para los documentos XML pasados como parámetros en el procedimiento (consulte las descripciones de parámetros de *entrada-xml*, *salida-xml*, y *mensaje-xml*). El procedimiento procesa todos los documentos XML en la versión especificada y devuelve un error (+20458) si la versión no es válida. En la salida, este parámetro especifica la versión principal del documento más reciente a la que da soporte el procedimiento. Para determinar la versión de documento más reciente soportada, especifique NULL para este parámetro de entrada y el resto de parámetros requeridos.

Versiones que reciben soporte: 1

versión-secundaria

Argumento de entrada y salida de tipo INTEGER que indica la versión secundaria del documento. En la entrada, este argumento especifica la versión secundaria del documento a la que da soporte el emisor de la llamada para los documentos XML pasados como parámetros para este procedimiento (consulte las descripciones de parámetros de *entrada-xml*, *salida-xml*, y *mensaje-xml*). El procedimiento procesa todos los documentos XML en la versión especificada y devuelve un error si la versión no es válida. En la salida, este parámetro indica la versión secundaria del documento más reciente que se soporta para la versión principal soportada más reciente. Para determinar la versión de documento más reciente soportada, especifique NULL para este parámetro de entrada y el resto de parámetros requeridos.

Versiones que reciben soporte: 0

entorno-local-solicitado

Argumento de entrada de tipo VARCHAR(33) que especifica un entorno local. Si el servidor da soporte al idioma especificado, se devuelve contenido traducido en los parámetros *salida-xml* y *mensaje-xml*. De lo contrario, el contenido se devuelve en el idioma por omisión. Sólo se utiliza el idioma y en ocasiones la información de territorio procedente del entorno local. El entorno local no se utiliza para dar formato a los números o influir en la codificación del documento. Por ejemplo, los valores y los nombres de claves no se traducen. La única parte traducida de los documentos de mensaje XML y salida XML son los textos de las sugerencias, el nombre de visualización y la unidad de visualización de cada entrada. El emisor de la llamada debe comparar siempre el idioma solicitado con el idioma que se está utilizando en el documento XML de salida (consulte la entrada de entorno local de documento en el documento XML de salida).

Actualmente, el único valor soportado para *entorno-local-solicitado* es en_US.

entrada-xml

Un argumento de entrada de tipo BLOB(32MB) que especifica un documento XML de entrada (con codificación UTF-8) que contiene los valores de entrada para el procedimiento.

Para este procedimiento, el documento XML de entrada debe especificar un descriptor de contexto de aplicación. Si desea cancelar una actividad específica, el documento XML de entrada debe especificar también los parámetros

opcionales que identifican un ID de unidad de trabajo y un ID de actividad. El documento XML de entrada completo de este procedimiento almacenado es similar al documento siguiente:

```
<?xml version="1.0" encoding="UTF-8"?>
<plist version="1.0">
<dict>
  <key>Document Type Name</key><string>Data Server Cancel Work Input</string>
  <key>Document Type Major Version</key><integer>1</integer>
  <key>Document Type Minor Version</key><integer>0</integer>
  <key>Required Parameters</key>
  <dict>
    <key>Application Handle</key>
    <dict>
      <key>Display name</key><string>Descriptor de contexto de aplicación</string>
      <key>Value</key><integer>10</integer>
      <key>Hint</key>
      <string>
        Valor numérico equivalente al descriptor de contexto de aplicación
        que se va a cancelar
      </string>
    </dict>
  </dict>
  <key>Optional Parameters</key>
  <dict>
    <key>Unit Of Work Id</key>
    <dict>
      <key>Display Name</key><string>ID de unidad de trabajo</string>
      <key>Value</key><integer>20</integer>
      <key>Hint</key>
      <string>
        Valor numérico que especifica el ID de unidad de trabajo de la actividad
        que se va a cancelar
      </string>
    </dict>
    <key>Activity Id</key>
    <dict>
      <key>Display Name</key><string>ID de actividad</string>
      <key>Value</key><integer>10</integer>
      <key>Hint</key>
      <string>
        Valor numérico equivalente al ID de aplicación que se va a cancelar
      </string>
    </dict>
  </dict>
</dict>
</plist>
```

Si especifica el descriptor de contexto de aplicación de la aplicación en la que se está ejecutando el procedimiento almacenado, el procedimiento devuelve un aviso (SQL20458).

filtro-xml

Argumento de entrada de tipo BLOB(4K) que especifica una serie de consulta XPath válida. Utilice un filtro si desea recuperar un único valor de un documento XML de salida. Para obtener más información, consulte el tema que describe el filtrado XPath.

El ejemplo siguiente selecciona el valor de la versión del producto de servidor de datos del documento XML de salida: `/plist/dict/key[.='Versión del producto de servidor de datos']/following-sibling::string`. Si la clave no va seguida del elemento hermano especificado, se devuelve un error.

salida-xml

Parámetro de salida de tipo BLOB(32MB) que devuelve un documento XML de

salida completo en UTF-8. Si se especifica un filtro, este parámetro devuelve un valor de serie. Si el procedimiento almacenado no puede devolver un documento de salida completo (por ejemplo, si se produce un error de proceso que genera un error o aviso de SQL), este parámetro se fija en NULL.

La salida XML se determina mediante los valores que se han especificado para *versión-principal* y *versión-secundaria*:

| Versión principal | Versión secundaria | Valor de <i>salida-xml</i> |
|-------------------|--------------------|--|
| NULL | NULL | NULL |
| 1 | 0 | Estado de la actividad que el procedimiento ha intentado cancelar. |

Cuando el procedimiento funciona en la *modalidad completa*, este parámetro devuelve un documento XML que se puede modificar y pasar de vuelta al procedimiento como parámetro *entrada-xml*. Este método ofrece un enfoque programático para la creación de documentos de entrada XML válidos. Para obtener más información, consulte el tema sobre modalidad completa.

mensaje-xml

Parámetro de salida XML completo de tipo BLOB(64K) que devuelve un documento XML de salida completo del tipo de mensaje de servidor de datos en codificación UTF-8 que ofrece información detallada sobre una condición de aviso de SQL. Este documento se devuelve cuando la llamada a un procedimiento genera un aviso de SQL, y el mensaje de aviso indica que se devuelve información adicional en el documento de salida del mensaje XML. Si el mensaje de aviso no especifica que se devuelve información adicional, este parámetro se fija en NULL.

Autorización

- Autorización SYSADM o DBADM
- Privilegio EXECUTE para el procedimiento CANCEL_WORK

Ejemplo

Ejemplo 1: Se devuelve la versión soportada más reciente del procedimiento.

```
db2 "call sysproc.cancel_work(null,null,null,null,null,?,?)"
```

El ejemplo siguiente muestra los datos de salida de la consulta:

Valor parámetros de salida

```
-----
Nombre de parámetro : VERSIÓN-PRINCIPAL
Valor de parámetro  : 1
```

```
Nombre de parámetro : VERSIÓN-SECUNDARIA
Valor de parámetro  : 0
```

```
Nombre de parámetro : SALIDA-XML
Valor de parámetro  : -
```

```
Nombre de parámetro : MENSAJE-XML
Valor de parámetro  : -
```

Estado de devolución = 0

Ejemplo 2: Se cancela una actividad específica.

```

db2 "call sysproc.cancel_work(1,0,'en_US',blob(
<?xml version="1.0" encoding="UTF-8"?>
<plist version="1.0">
<dict>
  <key>Document Type Name</key><string>Data Server Cancel Work Input</string>
  <key>Document Type Major Version</key><integer>1</integer>
  <key>Document Type Minor Version</key><integer>0</integer>
  <key>Required Parameters</key>
  <dict>
    <key>Application Handle</key>
    <dict>
      <key>Display name</key><string>Descriptor de contexto de aplicación</string>
      <key>Value</key><integer>1</integer>
      <key>Hint</key>
      <string>
        Valor numérico equivalente al descriptor de contexto de aplicación
        que se va a cancelar
      </string>
    </dict>
  </dict>
  <key>Optional Parameters</key>
  <dict>
    <key>Unit Of Work Id</key>
    <dict>
      <key>Display Name</key><string>ID de unidad de trabajo</string>
      <key>Value</key><integer>2</integer>
      <key>Hint</key>
      <string>
        Valor numérico que especifica el ID de unidad de trabajo de la actividad
        que se va a cancelar
      </string>
    </dict>
    <key>Activity Id</key>
    <dict>
      <key>Display Name</key><string>ID de actividad</string>
      <key>Value</key><integer>3</integer>
      <key>Hint</key>
      <string>
        Valor numérico equivalente al ID de aplicación que se va a cancelar
      </string>
    </dict>
  </dict>
</plist> ) ,null,?,?)"

```

El siguiente ejemplo muestra una salida de esta consulta:

Valor parámetros de salida

```

-----
Nombre de parámetro : VERSIÓN-PRINCIPAL
Valor de parámetro  : 1

```

```

Nombre de parámetro : VERSIÓN-SECUNDARIA
Valor de parámetro  : 0

```

```

Nombre de parámetro : SALIDA-XML
Valor de parámetro  : x'3C3F78...'

```

```

Nombre de parámetro : MENSAJE-XML
Valor de parámetro  : -

```

Estado de devolución = 0

Si el procedimiento CANCEL_WORK logra cancelar la actividad, el contenido del documento XML de salida es el siguiente:

```

<?xml version="1.0" encoding="UTF-8"?>
<plist version="1.0">
<dict><key>Document Type Name</key><string>Data Server Cancel Work Output</string>
  <key>Document Type Major Version</key><integer>1</integer>
  <key>Document Type Minor Version</key><integer>0</integer>
  <key>Data Server Product Name</key><string>QDB2/AIX64</string>
  <key>Data Server Product Version</key><string>9.7.0.0</string>
  <key>Data Server Major Version</key><integer>9</integer>
  <key>Data Server Minor Version</key><integer>7</integer>
  <key>Data Server Platform</key><string>AIX 64BIT</string>
  <key>Document Locale</key><string>en_US</string>
  <key>Successful Cancel Work Message</key>
  <dict>
    <key>Display Name</key><string>Mensaje trabajo cancelación satisfactorio</string>
    <key>Value</key><string>The activity has been cancelled successfully</string>
    <key>Hint</key><string></string>
  </dict>
</dict>
</plist>

```

Ejemplo 2: Se cancela la aplicación.

```

db2 "call sysproc.cancel_work(1,0,'en_US,blob(
<?xml version="1.0" encoding="UTF-8"?>
<plist version="1.0">
<dict>
  <key>Document Type Name</key><string>Data Server Cancel Work Input</string>
  <key>Document Type Major Version</key><integer>1</integer>
  <key>Document Type Minor Version</key><integer>0</integer>
  <key>Required Parameters</key>
  <dict>
    <key>Application Handle</key>
    <dict>
      <key>Display name</key><string>Descriptor de contexto de aplicación</string>
      <key>Value</key><integer>101</integer>
      <key>Hint</key>
      <string>
        Valor numérico equivalente al descriptor de contexto de aplicación
        que se va a cancelar
      </string>
    </dict>
  </dict>
</dict>
</plist> ),null,?,?)"

```

El siguiente ejemplo muestra una salida de esta consulta:

```

Valor parámetros de salida
-----
Nombre de parámetro : VERSIÓN-PRINCIPAL
Valor de parámetro  : 1

Nombre de parámetro : VERSIÓN-SECUNDARIA
Valor de parámetro  : 0

Nombre de parámetro : SALIDA-XML
Valor de parámetro  : x'3C3F78...'

Nombre de parámetro : MENSAJE-XML
Valor de parámetro  : -

Estado de devolución = 0

```

Si el procedimiento CANCEL_WORK logra cancelar la aplicación, el contenido del documento XML de salida es el siguiente:

```

<?xml version="1.0" encoding="UTF-8"?>
<plist version="1.0">
<dict>
  <key>Document Type Name</key><string>Data Server Cancel Work Output</string>
  <key>Document Type Major Version</key><integer>1</integer>
  <key>Document Type Minor Version</key><integer>0</integer>
  <key>Data Server Product Name</key><string>QDB2/AIX64</string>
  <key>Data Server Product Version</key><string>9.7.0.0</string>
  <key>Data Server Major Version</key><integer>9</integer>
  <key>Data Server Minor Version</key><integer>7</integer>
  <key>Data Server Platform</key><string>AIX 64BIT</string>
  <key>Document Locale</key><string>en_US</string>
  <key>Successful Cancel Work Message</key>
  <dict>
    <key>Display Name</key><string>Mensaje trabajo cancelación satisfactorio</string>
    <key>Value</key>
    <string>The application has been cancelled successfully</string>
    <key>Hint</key><string></string>
  </dict>
</dict>
</plist>

```

Ejemplo 3: Se especifica un filtro para devolver el valor de un mensaje de trabajo de cancelación satisfactorio.

```

db2 "call sysproc.cancel_work(1,0,'en_US,blob(
<?xml version="1.0" encoding="UTF-8"?>
<plist version="1.0">
<dict>
  <key>Document Type Name</key><string>Data Server Cancel Work Input</string>
  <key>Document Type Major Version</key><integer>1</integer>
  <key>Document Type Minor Version</key><integer>0</integer>
  <key>Required Parameters</key>
  <dict>
    <key>Application Handle</key>
    <dict>
      <key>Display name</key><string>Descriptor de contexto de aplicación</string>
      <key>Value</key><integer>101</integer>
      <key>Hint</key>
      <string>
        Valor numérico equivalente al descriptor de contexto de aplicación
        que se va a cancelar
      </string>
    </dict>
  </dict>
</dict>
</plist> ),blob('/plist/dict/key[.="Mensaje trabajo cancelación satisfactorio"]
/following-sibling::dict[1]/key[.="Value"]
/following-sibling::string[1]'),?,?)"

```

El ejemplo siguiente muestra los datos de salida de la consulta:

```

Valor parámetros de salida
-----
Nombre de parámetro : VERSIÓN-PRINCIPAL
Valor de parámetro  : 1

Nombre de parámetro : VERSIÓN-SECUNDARIA
Valor de parámetro  : 0

Nombre de parámetro : SALIDA-XML
Valor de parámetro  : x'3C3F78...'

Nombre de parámetro : MENSAJE-XML
Valor de parámetro  : -

Estado de devolución = 0

```

Se devuelve el valor siguiente para *salida-xml*:
"The application has been cancelled successfully"

Procedimiento GET_CONFIG - Obtener datos de configuración

El procedimiento almacenado GET_CONFIG recupera la configuración de la base de datos, la configuración del gestor de bases de datos y las variables de registro que se han determinado para una instancia concreta.

En un entorno de bases de datos particionadas, este procedimiento recupera los valores de configuración de la base de datos y de la variable de registro de todas las particiones.

Sintaxis

```
► GET_CONFIG(—versión-principal—,—versión-secundaria—,—entorno-local-solicitado—,—  
► entrada-xml—,—filtro-xml—,—salida-xml—,—mensaje-xml—)
```

El esquema es SYSPROC.

Parámetros del procedimiento

versión-principal

Argumento de entrada y salida de tipo INTEGER que indica la versión principal del documento. En la entrada, este argumento indica la versión principal del documento a la que da soporte el emisor de la llamada para los documentos XML pasados como parámetros en el procedimiento (consulte las descripciones de parámetros de *entrada-xml*, *salida-xml*, y *mensaje-xml*). El procedimiento procesa todos los documentos XML en la versión especificada y devuelve un error (+20458) si la versión no es válida. En la salida, este parámetro especifica la versión principal del documento más reciente a la que da soporte el procedimiento. Para determinar la versión de documento más reciente soportada, especifique NULL para este parámetro de entrada y el resto de parámetros requeridos.

Versiones que reciben soporte: 1 y 2

versión-secundaria

Argumento de entrada y salida de tipo INTEGER que indica la versión secundaria del documento. En la entrada, este argumento especifica la versión secundaria del documento a la que da soporte el emisor de la llamada para los documentos XML pasados como parámetros para este procedimiento (consulte las descripciones de parámetros de *entrada-xml*, *salida-xml*, y *mensaje-xml*). El procedimiento procesa todos los documentos XML en la versión especificada y devuelve un error si la versión no es válida. En la salida, este parámetro indica la versión secundaria del documento más reciente que se soporta para la versión principal soportada más reciente. Para determinar la versión de documento más reciente soportada, especifique NULL para este parámetro de entrada y el resto de parámetros requeridos.

Versiones que reciben soporte: 0

entorno-local-solicitado

Argumento de entrada de tipo VARCHAR(33) que especifica un entorno local. Si el servidor da soporte al idioma especificado, se devuelve contenido traducido en los parámetros *salida-xml* y *mensaje-xml*. De lo contrario, el

contenido se devuelve en el idioma por omisión. Sólo se utiliza el idioma y en ocasiones la información de territorio procedente del entorno local. El entorno local no se utiliza para dar formato a los números o influir en la codificación del documento. Por ejemplo, los valores y los nombres de claves no se traducen. La única parte traducida de los documentos de mensaje XML y salida XML son los textos de las sugerencias, el nombre de visualización y la unidad de visualización de cada entrada. El emisor de la llamada debe comparar siempre el idioma solicitado con el idioma que se está utilizando en el documento XML de salida (consulte la entrada de entorno local de documento en el documento XML de salida).

Actualmente, el único valor soportado para *entorno-local-solicitado* es `en_US`.

entrada-xml

Actualmente, este procedimiento no acepta ninguna entrada. Debe especificar NULL en este parámetro, o se emitirá un error (+20458) para indicar que la entrada no es válida.

filtro-xml

Argumento de entrada de tipo BLOB(4K) que especifica una serie de consulta XPath válida. Utilice un filtro si desea recuperar un único valor de un documento XML de salida. Para obtener más información, consulte el tema que describe el filtrado XPath.

El ejemplo siguiente selecciona el valor de la versión del producto de servidor de datos del documento XML de salida: `/plist/dict/key[.='Versión del producto de servidor de datos']/following-sibling::string`. Si la clave no va seguida del elemento hermano especificado, se devuelve un error.

salida-xml

Parámetro de salida de tipo BLOB(32MB) que devuelve un documento XML de salida completo en UTF-8. Si se especifica un filtro, este parámetro devuelve un valor de serie. Si el procedimiento almacenado no puede devolver un documento de salida completo (por ejemplo, si se produce un error de proceso que genera un error o aviso de SQL), este parámetro se fija en NULL.

La salida XML se determina mediante los valores que se han especificado *paraversión-principal* y *versión-secundaria*:

| Versión principal | Versión secundaria | Valor de <i>salida-xml</i> |
|-------------------|--------------------|--|
| NULL | NULL | NULL |
| 1 | 0 | Parámetros de configuración de la base de datos y del gestor de bases de datos y variables de registro, incluidos sus valores. |
| 2 | 0 | Parámetros de configuración de la base de datos y del gestor de bases de datos agrupados en categorías. Para cada parámetro, indica si se puede actualizar el parámetro. También devuelve las variables de registro y los valores fijados para la instancia. |

Cuando el procedimiento funciona en la *modalidad completa*, este parámetro devuelve un documento XML que se puede modificar y pasar de vuelta al

procedimiento como parámetro *entrada-xml*. Este método ofrece un enfoque programático para la creación de documentos de entrada XML válidos. Para obtener más información, consulte el tema sobre modalidad completa.

mensaje-xml

Parámetro de salida XML completo de tipo BLOB(64K) que devuelve un documento XML de salida completo del tipo de mensaje de servidor de datos en codificación UTF-8 que ofrece información detallada sobre una condición de aviso de SQL. Este documento se devuelve cuando la llamada a un procedimiento genera un aviso de SQL, y el mensaje de aviso indica que se devuelve información adicional en el documento de salida del mensaje XML. Si el mensaje de aviso no especifica que se devuelve información adicional, este parámetro se fija en NULL.

Autorización

- Autorización SYSADM o DBADM
- Privilegio EXECUTE para el procedimiento GET_CONFIG

Ejemplo

Ejemplo 1: Se devuelve la versión más reciente del procedimiento.

```
db2 "call sysproc.get_config(null,null,null,null,null,?,?)"
```

El ejemplo siguiente muestra los datos de salida de la consulta:

```
Valor parámetros de salida
-----
Nombre de parámetro : VERSIÓN-PRINCIPAL
Valor de parámetro  : 2

Nombre de parámetro : VERSIÓN-SECUNDARIA
Valor de parámetro  : 0

Nombre de parámetro : SALIDA-XML
Valor de parámetro  : -

Nombre de parámetro : MENSAJE-XML
Valor de parámetro  : -

Estado de devolución = 0
```

Ejemplo 2: Se devuelven los parámetros de configuración de la base de datos y del gestor de bases de datos, agrupados en categorías.

```
db2 "call sysproc.get_config(2,0,'en_US',null, null, ?,?)"
```

El ejemplo siguiente muestra los datos de salida de la consulta:

```
Valor parámetros de salida
-----
Nombre de parámetro : VERSIÓN-PRINCIPAL
Valor de parámetro  : 2

Nombre de parámetro : VERSIÓN-SECUNDARIA
Valor de parámetro  : 0

Nombre de parámetro : SALIDA-XML
Valor de parámetro  : x'3C3F78.....'

Nombre de parámetro : MENSAJE-XML
Valor de parámetro  : -

Estado de devolución = 0
```

El contenido del documento XML de salida es el siguiente:

```
<plist version="1.0">
<dict>
  <key>Document Type Name</key><string>Data Server Configuration Output</string>
  <key>Document Type Major Version</key><integer>2</integer>
  <key>Document Type Minor Version</key><integer>0</integer>
  <key>Data Server Product Name</key><string>QDB2/AIX64</string>
  <key>Data Server Product Version</key><string>9.7.0.0</string>
  <key>Data Server Major Version</key><integer>9</integer>
  <key>Data Server Minor Version</key><integer>7</integer>
  <key>Data Server Platform</key><string>AIX 64BIT</string>
  <key>Document Locale</key><string>en_US</string>
  <key>Database Manager Configuration Parameter Settings</key>
  <dict>
    <key>Display Name</key>
    <string>Valores del parámetro de configuración del gestor de bases de datos</string>
    <key>Application</key>
    <dict>
      <key>Display Name</key><string>Aplicación</string>
      <key>agentpri</key>
      <dict>
        <key>Display Name</key><string>agentpri</string>
        <key>Parameter Value</key>
        <dict>
          <key>Display Name</key><string>Valor de parámetro</string>
          <key>Value</key><string>-1</string>
          <key>Updatable</key><string>No</string>
          <key>Hint</key><string></string>
        </dict>
        <key>Value Flags</key>
        <dict>
          <key>Display Name</key><string>Distintivos de valor</string>
          <key>Value</key><string>NONE</string>
          <key>Updatable</key><string>No</string>
          <key>Hint</key><string></string>
        </dict>
        <key>Deferred Value</key>
        <dict>
          <key>Display Name</key><string>Valor diferido</string>
          <key>Value</key><string>-1</string>
          <key>Updatable</key><string>Yes</string>
          <key>Hint</key><string></string>
        </dict>
        <key>Deferred Value Flags</key>
        <dict>
          <key>Display Name</key><string>Distintivos de valor diferido</string>
          <key>Value</key><string>INTEGER</string>
          <key>Updatable</key><string>Yes</string>
          <key>Hint</key><string></string>
        </dict>
        <key>Data Type</key>
        <dict>
          <key>Display Name</key><string>Tipo de datos</string>
          <key>Value</key><string>NONE</string>
          <key>Hint</key><string></string>
        </dict>
        <key>Hint</key>
        <string>
          Especifica la prioridad que el planificador del sistema operativo otorga a un agente y a otros procesos y hebras de instancia del gestor de bases de datos. Considere la posibilidad de volver a vincular las aplicaciones después de haber cambiado este parámetro.
        </string>
      </dict>
      <key>Hint</key><string></string>
    </dict>
  </dict>
</plist>
```

```

<key>Administration</key>
.
.
<key>Communication</key>
.
.
<key>Diagnostics</key>
.
.
<key>Environment</key>
.
.
<key>Miscellaneous</key>
.
.
<key>Monitor</key>
.
.
<key>Parallel</key>
.
.
<key>Performance</key>
.
.
</dict>
<key>Database Partition</key>
<dict>
  <key>Display Name</key><string>Partición de base de datos</string>
  <key>0</key>
  <dict>
    <key>Display Name</key><string>0</string>
    <key>Database Configuration Parameter Settings</key>
    <dict>
      <key>Display Name</key>
      <string>Valores del parámetro de configuración de base de datos</string>
      <key>Application</key>
      .
      .
      <key>Environment</key>
    </dict>
  <dict>
    <key>Display Name</key><string>Entorno</string>
    <key>alt_collate</key>
    <dict>
      <key>Display Name</key><string>alt_collate</string>
      <key>Parameter Value</key>
      <dict>
        <key>Display Name</key><string>Valor de parámetro</string>
        <key>Value</key><string></string>
        <key>Updatable</key><string>No</string>
        <key>Hint</key><string></string>
      </dict>
    <key>Value Flags</key>
    <dict>
      <key>Display Name</key><string>Distintivos de valor</string>
      <key>Value</key><string>NONE</string>
      <key>Updatable</key><string>No</string>
      <key>Hint</key><string></string>
    </dict>
  </dict>
</dict>

```

```

<key>Deferred Value</key>
<dict>
  <key>Display Name</key><string>Valor diferido</string>
  <key>Value</key><string></string>
  <key>Updatable</key><string>Yes</string>
  <key>Hint</key><string></string>
</dict>
<key>Deferred Value Flags</key>
<dict>
  <key>Display Name</key><string>Distintivos de valor diferido</string>
  <key>Value</key><string>INTEGER</string>
  <key>Updatable</key><string>Yes</string>
  <key>Hint</key><string></string>
</dict>
<key>Data Type</key>
<dict>
  <key>Display Name</key><string>Tipo de datos</string>
  <key>Value</key><string>NONE</string>
  <key>Hint</key><string></string>
</dict>
<key>Hint</key>
<string>
Especifica el orden de clasificación que se debe utilizar para
tablas Unicode en una base de datos que no es Unicode. A menos
que se defina este parámetro, no se pueden crear tablas y
rutinas de Unicode en una base de datos que no sea de Unicode.
Cuando se ha establecido, este parámetro no puede cambiarse ni
restablecerse. Valor por omisión [rango] :
Null [IDENTITY_16BIT].
</string>
</dict>
.
.
.
</dict>
<key>Logs</key>
.
.
.
<key>Maintenance</key>
.
.
.
<key>Performance</key>
.
.
.
<key>Recovery</key>
.
.
.
<key>Status</key>
.
.
.
</dict>
<key>Registry Variables Settings</key>
<dict>
  <key>Display Name</key><string>Valores de variables de registro</string>
  <key>DB2CODEPAGE</key>
  <dict>
    <key>Display Name</key><string>DB2CODEPAGE</string>
    <key>Parameter Value</key>
    <dict>
      <key>Display Name</key><string>Valor de parámetro</string>
      <key>Value</key><string>1208</string>
      <key>Hint</key><string></string>
    </dict>
  </dict>
</dict>

```

```

        </dict>
        <key>Is Aggregate</key>
        <dict>
            <key>Display Name</key><string>Es agregado</string>
            <key>Value</key><integer>0</integer>
            <key>Hint</key><string></string>
        </dict>
        <key>Aggregate Name</key>
        <dict>
            <key>Display Name</key><string>Nombre agregado</string>
            <key>Value</key><string></string>
            <key>Hint</key><string></string>
        </dict>
        <key>Level</key>
        <dict>
            <key>Display Name</key><string>Nivel</string>
            <key>Value</key><string>I</string>
            <key>Hint</key><string></string>
        </dict>
        <key>Hint</key><string></string>
    </dict>
    .
    .
    .
    </dict>
    <key>Hint</key><string></string>
</dict>
</plist>

```

Ejemplo 3: Se devuelven los parámetros de configuración de la base de datos y del gestor de bases de datos.

```
db2 "call sysproc.get_config(1,0,'en_US',null, null, ?,?)"
```

El ejemplo siguiente muestra los datos de salida de la consulta:

```

Valor parámetros de salida
-----
Nombre de parámetro : VERSIÓN-PRINCIPAL
Valor de parámetro  : 1

Nombre de parámetro : VERSIÓN-SECUNDARIA
Valor de parámetro  : 0

Nombre de parámetro : SALIDA-XML
Valor de parámetro  : x'3C3F78.....'

Nombre de parámetro : MENSAJE-XML
Valor de parámetro  : -

Estado de devolución = 0

```

El contenido del documento XML de salida es similar al del ejemplo 2, pero no agrupa los parámetros de configuración en categorías.

Ejemplo 4: Se llama al procedimiento desde una función.

```

EXEC SQL BEGIN DECLARE SECTION;
    sqlint16 getconfigMaj;
    sqlint16 getconfigMin;

    SQL TYPE IS BLOB(2M) xmlOutput;
    SQL TYPE IS BLOB(2K) xmlOutMessage;
EXEC SQL END DECLARE SECTION;
    getconfigMaj = 2;
    getconfigMin = 0;

```

```
EXEC SQL CALL SYSPROC.GET_CONFIG(
    :getconfigMaj,
    :getconfigMin,
    'en_US',
    null,
    null,
    :xmlOutput,
    :xmlOutMessage );
```

Procedimiento GET_MESSAGE - Obtener texto de mensaje

El procedimiento GET_MESSAGE devuelve el texto corto del mensaje, el texto largo del mensaje y el SQLSTATE para un SQLCODE.

Sintaxis

```
►► GET_MESSAGE(—versión-principal—,—versión-secundaria—,—entorno-local-solicitado—,—entrada-xml—,—filtro-xml—,—salida-xml—,—mensaje-xml—)
```

El esquema es SYSPROC.

Parámetros del procedimiento

versión-principal

Argumento de entrada y salida de tipo INTEGER que indica la versión principal del documento. En la entrada, este argumento indica la versión principal del documento a la que da soporte el emisor de la llamada para los documentos XML pasados como parámetros en el procedimiento (consulte las descripciones de parámetros de *entrada-xml*, *salida-xml*, y *mensaje-xml*). El procedimiento procesa todos los documentos XML en la versión especificada y devuelve un error (+20458) si la versión no es válida. En la salida, este parámetro especifica la versión principal del documento más reciente a la que da soporte el procedimiento. Para determinar la versión de documento más reciente soportada, especifique NULL para este parámetro de entrada y el resto de parámetros requeridos.

Si el documento XML del parámetro *entrada-xml* especifica una clave Document Type Major Version, el valor de dicha clave debe ser igual al valor proporcionado en el parámetro *versión-principal*, o se emite un error (+20458).

Versiones que reciben soporte: 1 y 2

versión-secundaria

Argumento de entrada y salida de tipo INTEGER que indica la versión secundaria del documento. En la entrada, este argumento especifica la versión secundaria del documento a la que da soporte el emisor de la llamada para los documentos XML pasados como parámetros para este procedimiento (consulte las descripciones de parámetros de *entrada-xml*, *salida-xml*, y *mensaje-xml*). El procedimiento procesa todos los documentos XML en la versión especificada y devuelve un error si la versión no es válida. En la salida, este parámetro indica la versión secundaria del documento más reciente que se soporta para la versión principal soportada más reciente. Para determinar la versión de documento más reciente soportada, especifique NULL para este parámetro de entrada y el resto de parámetros requeridos.

Si el documento XML del parámetro *entrada-xml* especifica una clave Document Type Minor Version, el valor de dicha clave debe ser igual al valor proporcionado en el parámetro *versión-secundaria*, o se emite un error (+20458).

Versiones que reciben soporte: 0

entorno-local-solicitado

Argumento de entrada de tipo VARCHAR(33) que especifica un entorno local. Si el servidor da soporte al idioma especificado, se devuelve contenido traducido en los parámetros *salida-xml* y *mensaje-xml*. De lo contrario, el contenido se devuelve en el idioma por omisión. Sólo se utiliza el idioma y en ocasiones la información de territorio procedente del entorno local. El entorno local no se utiliza para dar formato a los números o influir en la codificación del documento. Por ejemplo, los valores y los nombres de claves no se traducen. La única parte traducida de los documentos de mensaje XML y salida XML son los textos de las sugerencias, el nombre de visualización y la unidad de visualización de cada entrada. El emisor de la llamada debe comparar siempre el idioma solicitado con el idioma que se está utilizando en el documento XML de salida (consulte la entrada de entorno local de documento en el documento XML de salida).

Actualmente, el único valor soportado para *entorno-local-solicitado* es en_US.

entrada-xml

Un argumento de entrada de tipo BLOB(32MB) que especifica un documento XML de entrada (con codificación UTF-8) que contiene los valores de entrada para el procedimiento.

Para este procedimiento, el documento XML de entrada contiene un SQLCODE y utiliza el formato siguiente:

```
<?xml version="1.0" encoding="UTF-8"?>
<plist version="1.0">
<dict>
  <key>Document Type Name</key><string>Data Server Message Input</string>
  <key>Required Parameters</key>
  <!-- Se especifica SQLCODE o el identificador de mensaje, así como los
  símbolos de mensaje, de los valores clave siguientes. -->
  <dict>
    <key>SQL Code</key><integer></integer>
    <key>Message Identifier</key><integer></integer>
    <key>Message Tokens</key><array><string>...</string></array>
  </dict>
  <key>Optional Parameters</key>
  <dict>
    <key>Message Token Delimiter<key><string>;</string>
  </key></key></dict>
</dict>
</plist>
```

filtro-xml

Argumento de entrada de tipo BLOB(4K) que especifica una serie de consulta XPath válida. Utilice un filtro si desea recuperar un único valor de un documento XML de salida. Para obtener más información, consulte el tema que describe el filtrado XPath.

El ejemplo siguiente selecciona el valor de SQLSTATE desde el documento XML de salida: `/plist/dict/key[.="SQLSTATE"]/following-sibling::dict[1]/key[.="Value"]/following-sibling::string[1]`. Si la clave no va seguida del elemento hermano especificado, se devuelve un error.

salida-xml

Parámetro de salida de tipo BLOB(32MB) que devuelve un documento XML de

salida completo en UTF-8. Si se especifica un filtro, este parámetro devuelve un valor de serie. Si el procedimiento almacenado no puede devolver un documento de salida completo (por ejemplo, si se produce un error de proceso que genera un error o aviso de SQL), este parámetro se fija en NULL.

La salida XML se determina mediante los valores que se han especificado *paraversión-principal* y *versión-secundaria*:

| Versión principal | Versión secundaria | Valor de <i>salida-xml</i> |
|-------------------|--------------------|---|
| NULL | NULL | NULL |
| 1 | 0 | Devuelve el mensaje de texto corto y el SQLSTATE para el SQLCODE correspondiente pasado en la <i>entrada-xml</i> . |
| 2 | 0 | Devuelve el mensaje de texto corto, el mensaje de texto largo y el SQLSTATE para el SQLCODE correspondiente pasado en la <i>entrada-xml</i> . |

Cuando el procedimiento funciona en la *modalidad completa*, este parámetro devuelve un documento XML que se puede modificar y pasar de vuelta al procedimiento como parámetro *entrada-xml*. Este método ofrece un enfoque programático para la creación de documentos de entrada XML válidos. Para obtener más información, consulte el tema sobre modalidad completa.

mensaje-xml

Parámetro de salida XML completo de tipo BLOB(64K) que devuelve un documento XML de salida completo del tipo de mensaje de servidor de datos en codificación UTF-8 que ofrece información detallada sobre una condición de aviso de SQL. Este documento se devuelve cuando la llamada a un procedimiento genera un aviso de SQL, y el mensaje de aviso indica que se devuelve información adicional en el documento de salida del mensaje XML. Si el mensaje de aviso no especifica que se devuelve información adicional, este parámetro se fija en NULL.

Autorización

- Autorización SYSADM o DBADM
- Privilegio EXECUTE para el procedimiento GET_MESSAGE

Ejemplo

Ejemplo 1: Se devuelve la versión soportada más reciente del procedimiento.
 db2 "call sysproc.get_message(null,null,null,null,null,?,?)"

El ejemplo siguiente muestra los datos de salida de la consulta:

```

Valor parámetros de salida
-----
Nombre de parámetro : VERSIÓN-PRINCIPAL
Valor de parámetro  : 2

Nombre de parámetro : VERSIÓN-SECUNDARIA
Valor de parámetro  : 0

Nombre de parámetro : SALIDA-XML
Valor de parámetro  : -
  
```

Nombre de parámetro : MENSAJE-XML
Valor de parámetro : -

Estado de devolución = 0

Ejemplo 2: Se ejecuta un script denominado getmsglong.sql para devolver el mensaje de texto corto y el mensaje de texto largo de SQL1034.

getmsglong.sql:

```
call sysproc.get_message(2,0, 'en_US', blob('
<?xml version="1.0" encoding="UTF-8"?>
<plist version="1.0">
<dict>
  <key>Document Type Name</key><string>Data Server Message Input</string>
  <key>Document Type Major Version</key><integer>2</integer>
  <key>Document Type Minor Version</key><integer>0</integer>
  <key>Required Parameters</key>
  <dict>
    <key>SQLCODE</key><string>SQL1034</string>
  </dict>
</dict>
</plist>'), null, ? , ?)@
```

El siguiente ejemplo muestra una salida de esta consulta:

Valor parámetros de salida

Nombre de parámetro : VERSIÓN-PRINCIPAL
Valor de parámetro : 2

Nombre de parámetro : VERSIÓN-SECUNDARIA
Valor de parámetro : 0

Nombre de parámetro : SALIDA-XML
Valor de parámetro : x'3C3F786D6C20766572.....'

Nombre de parámetro : MENSAJE-XML
Valor de parámetro : -

Estado de devolución = 0

El contenido del documento XML de salida es el siguiente:

```
<plist version="1.0">
<dict>
  <key>Document Type Name</key>
  <string>Data Server Message Output</string>
  <key>Document Type Major Version</key>
  <integer>2</integer>
  <key>Document Type Minor Version</key>
  <integer>0</integer>
  <key>Data Server Product Name</key>
  <string>QDB2/AIX64</string>
  <key>Data Server Product Version</key>
  <string>9.7.0.0</string>
  <key>Data Server Major Version</key>
  <integer>9</integer>
  <key>Data Server Minor Version</key>
  <integer>7</integer>
  <key>Data Server Platform</key>
  <string>AIX 64BIT</string>
  <key>Document Locale</key>
  <string>en_US</string>
  <key>Short Message Text</key>
  <dict>
    <key>Display Name</key><string>Texto corto del mensaje</string>
    <key>Value</key>
```

```

<string>
SQL1034C La base de datos está dañada. Todas las aplicaciones que procesan
la base de datos se han detenido.
</string>
<key>Hint</key><string></string>
</dict>
<key>SQLSTATE</key>
<dict>
  <key>Display Name</key><string>SQLSTATE</string>
  <key>Value</key><string> 58031</string>
  <key>Hint</key><string></string>
</dict>
<key>Long Message Text</key>
<dict>
  <key>Display Name</key><string>Texto de mensaje largo</string>
  <key>Value</key>
  <array>
    <string>
      SQL1034C La base de datos está dañada. Todas las aplicaciones
      que procesan
    </string>
    <string> la base de datos se han detenido.</string>
    <string></string>
    <string>Explicación: </string>
    <string></string>
    <string>
      La base de datos ha quedado dañada. No puede utilizarse hasta haber resuelto
    </string>
    <string>
      el problema. Todas las aplicaciones conectadas a la base de datos se han
    </string>
    <string>
      desconectado y todos los procesos que ejecutan dichas aplicaciones en la
      base de datos se
    </string>
    <string>han detenido.</string>
    <string></string>
    <string>No se puede procesar el mandato.</string>
    <string></string>
    <string>Respuesta del usuario: </string>
    <string></string>
    <string>
      Emita el mandato RESTART DATABASE para recuperar la base de datos.
      Si el mandato RESTART
    </string>
    <string>
      también falla constantemente, tal vez sea necesario restaurar la base de
      datos a partir de una
    </string>
    <string>
      copia de seguridad. En un entorno de servidor de base de datos particionada,
      compruebe syslog
    </string>
    <string>
      para saber si el mandato RESTART ha fallado debido a anomalías de
      comunicaciones o
    </string>
    <string>
      del nodo antes de restaurar la base de datos desde una copia de seguridad.
      En tal
    </string>
    <string>
      caso, asegúrese de que el gestor de bases de datos está activo y funciona
      correctamente y que la comunicación está
    </string>
    <string>
      disponible entre todos los nodos; acto seguido, vuelva a someter el

```

```

mandato restart.
</string>
<string></string>
<string>
Si ha encontrado este error durante el proceso de recuperación en avance, debe
</string>
<string>
restaurar la base de datos desde una copia de seguridad y volver a realizar
la recuperación en avance.
</string>
<string></string>
<string>
Tenga en cuenta que, en un entorno de bases de datos particionadas, el
mandato RESTART
</string>
<string>
de la base de datos se ejecuta nodo por nodo. Para asegurarse de que la base de
datos sea
</string>
<string>restaurada en todos los nodos, utilice el mandato: </string>
<string></string>
<string>db2_all db2 restart database</string>
<string><nombre_basedatos></string>
<string></string>
<string>
Es posible que este mandato tenga que ejecutarse varias veces para
tener la certeza
</string>
<string>de que se hayan resuelto todas las transacciones dudosas.</string>
<string></string>
<string>
Si está instalando la base de datos de ejemplo, descártela e instálela
</string>
<string>de nuevo.</string>
<string></string>
<string> sqlcode: -1034</string>
<string></string>
<string> sqlstate: 58031</string>
<string></string>
<string></string>
<string></string>
</array>
<key>Hint</key><string></string>
</dict>
</dict>
</plist>

```

Ejemplo 3: Se ejecuta un script denominado `getmsgshort.sql` para devolver únicamente el mensaje de texto corto de SQL1034.

`getmsgshort.sql`:

```

call sysproc.get_message(1,0,'en_US', blob('
<?xml version="1.0" encoding="UTF-8"?>
<plist version="1.0">
<dict>
  <key>Document Type Name</key><string>Data Server Message Input</string>
  <key>Document Type Major Version</key><integer>1</integer>
  <key>Document Type Minor Version</key><integer>0</integer>
  <key>Required Parameters</key>
  <dict>
    <key>SQLCODE</key><string>SQL1034</string>
  </dict>
</dict>
</plist>'), null, ? , ?)@

```

El siguiente ejemplo muestra una salida de esta consulta:

Valor parámetros de salida

Nombre de parámetro : VERSIÓN-PRINCIPAL

Valor de parámetro : 2

Nombre de parámetro : VERSIÓN-SECUNDARIA

Valor de parámetro : 0

Nombre de parámetro : SALIDA-XML

Valor de parámetro : x'3C3F786D6C20766572.....'

Nombre de parámetro : MENSAJE-XML

Valor de parámetro : -

Estado de devolución = 0

SQL20460W El procedimiento "SYSPROC.GET_MESSAGE" proporciona soporte a una versión superior, "2", a la versión especificada, "1", para el parámetro "1".

El contenido del documento XML de salida es el siguiente:

```
<plist version="1.0">
<dict><key>Document Type Name</key><string>Data Server Message Output</string>
  <key>Document Type Major Version</key><integer>1</integer>
  <key>Document Type Minor Version</key><integer>0</integer>
  <key>Data Server Product Name</key><string>QDB2/AIX64</string>
  <key>Data Server Product Version</key><string>9.7.0.0</string>
  <key>Data Server Major Version</key><integer>9</integer>
  <key>Data Server Minor Version</key><integer>7</integer>
  <key>Data Server Platform</key><string>AIX 64BIT</string>
  <key>Document Locale</key><string>en_US</string>
  <key>Short Message Text</key>
  <dict>
    <key>Display Name</key><string>Texto corto del mensaje</string>
    <key>Value</key>
    <string>
      SQL1034C La base de datos está dañada. Todas las aplicaciones que procesan
      la base de datos se han detenido.
    </string>
    <key>Hint</key><string></string>
  </dict>
  <key>SQLSTATE</key>
  <dict>
    <key>Display Name</key><string>SQLSTATE</string>
    <key>Value</key><string> 58031</string>
    <key>Hint</key><string></string>
  </dict>
</dict>
</plist>
```

Ejemplo 4: Se especifica un filtro para devolver SQLSTATE para SQL1034.

```
db2 "call sysproc.get_message(2,0, 'en_US', blob('
```

```
<plist version="1.0">
```

```
<dict>
```

```
  <key>Document Type Name</key>
```

```
  <string>Data Server Message Input</string>
```

```
  <key>Required Parameters</key>
```

```
  <dict>
```

```
    <key>SQLCODE</key><string>SQL1034</string>
```

```
  </dict>
```

```
</dict>
```

```
</plist>'),
```

```
blob('/plist/dict/key[.="SQLSTATE"]/following-sibling::dict[1]/
```

```
key[.="Value"]/following-sibling::string[1]'), ? , ?)"
```

El ejemplo siguiente muestra los datos de salida de la consulta:

Valor parámetros de salida

Nombre de parámetro : VERSIÓN-PRINCIPAL
Valor de parámetro : 2

Nombre de parámetro : VERSIÓN-SECUNDARIA
Valor de parámetro : 0

Nombre de parámetro : SALIDA-XML
Valor de parámetro : x'203538303331'

Nombre de parámetro : MENSAJE-XML
Valor de parámetro : -

Estado de devolución = 0

Se devuelve el valor siguiente para *salida-xml*:
58031

Ejemplo 5: Se llama al procedimiento desde una función.

```
EXEC SQL BEGIN DECLARE SECTION;
  sqlint16  getMsgMaj;
  sqlint16  getMsgMin;

  SQL TYPE IS BLOB(2M) xmlOutput;
  SQL TYPE IS BLOB(2K) xmlOutMessage;
EXEC SQL END DECLARE SECTION;
  getMsgMaj = 2;
  getMsgMin = 0;

EXEC SQL CALL SYSPROC.GET_MESSAGE(
  :getMsgMaj,
  :getMsgMin,
  'en_US',
  BLOB('
  <?xml version="1.0" encoding="UTF-8"?>
  <plist version="1.0">
  <dict>
    <key>Document Type Name</key>
    <string>
    Data Server Message Input
    </string>
    <key>Document Type Major Version</key><integer>2</integer>
    <key>Document Type Minor Version</key><integer>0</integer>
    <key>Required Parameters</key>
    <dict>
      <key>SQLCODE</key><string>SQL1034</string>
    </dict>
  </dict>
  </plist>'),
  null,
  :xmlOutput,
  :xmlOutMessage );
```

Procedimiento GET_SYSTEM_INFO - Obtener información de sistema

El procedimiento GET_SYSTEM_INFO devuelve información sobre el servidor de datos, incluida información sobre el sistema, la instancia actual, los productos de servidor de datos instalados, las variables de entorno, las CPU disponibles y otra información de sistema.

Sintaxis

```
► GET_SYSTEM_INFO(—versión-principal—,—versión-secundaria—,—entorno-local-solicitado—,—  
► entrada-xml—,—filtro-xml—,—salida-xml—,—mensaje-xml—)
```

El esquema es SYSPROC.

Parámetros del procedimiento

versión-principal

Argumento de entrada y salida de tipo INTEGER que indica la versión principal del documento. En la entrada, este argumento indica la versión principal del documento a la que da soporte el emisor de la llamada para los documentos XML pasados como parámetros en el procedimiento (consulte las descripciones de parámetros de *entrada-xml*, *salida-xml*, y *mensaje-xml*). El procedimiento procesa todos los documentos XML en la versión especificada y devuelve un error (+20458) si la versión no es válida. En la salida, este parámetro especifica la versión principal del documento más reciente a la que da soporte el procedimiento. Para determinar la versión de documento más reciente soportada, especifique NULL para este parámetro de entrada y el resto de parámetros requeridos.

Si el documento XML del parámetro *entrada-xml* especifica una clave Document Type Major Version, el valor de dicha clave debe ser igual al valor proporcionado en el parámetro *versión-principal*, o se emite un error (+20458).

Versiones que reciben soporte: 1

versión-secundaria

Argumento de entrada y salida de tipo INTEGER que indica la versión secundaria del documento. En la entrada, este argumento especifica la versión secundaria del documento a la que da soporte el emisor de la llamada para los documentos XML pasados como parámetros para este procedimiento (consulte las descripciones de parámetros de *entrada-xml*, *salida-xml*, y *mensaje-xml*). El procedimiento procesa todos los documentos XML en la versión especificada y devuelve un error si la versión no es válida. En la salida, este parámetro indica la versión secundaria del documento más reciente que se soporta para la versión principal soportada más reciente. Para determinar la versión de documento más reciente soportada, especifique NULL para este parámetro de entrada y el resto de parámetros requeridos.

Versiones que reciben soporte: 0

entorno-local-solicitado

Argumento de entrada de tipo VARCHAR(33) que especifica un entorno local. Si el servidor da soporte al idioma especificado, se devuelve contenido traducido en los parámetros *salida-xml* y *mensaje-xml*. De lo contrario, el contenido se devuelve en el idioma por omisión. Sólo se utiliza el idioma y en ocasiones la información de territorio procedente del entorno local. El entorno local no se utiliza para dar formato a los números o influir en la codificación del documento. Por ejemplo, los valores y los nombres de claves no se traducen. La única parte traducida de los documentos de mensaje XML y salida XML son los textos de las sugerencias, el nombre de visualización y la unidad de visualización de cada entrada. El emisor de la llamada debe comparar siempre el idioma solicitado con el idioma que se está utilizando en el documento XML de salida (consulte la entrada de entorno local de documento en el documento XML de salida).

Actualmente, el único valor soportado para *entorno-local-solicitado* es en_US.

entrada-xml

Actualmente, este procedimiento no acepta ninguna entrada. Debe especificar NULL en este parámetro, o se emitirá un error (+20458) para indicar que la entrada no es válida.

filtro-xml

Argumento de entrada de tipo BLOB(4K) que especifica una serie de consulta XPath válida. Utilice un filtro si desea recuperar un único valor de un documento XML de salida. Para obtener más información, consulte el tema que describe el filtrado XPath.

El ejemplo siguiente selecciona el valor de la versión del producto de servidor de datos del documento XML de salida: `/plist/dict/key[.='Versión del producto de servidor de datos']/following-sibling::string`. Si la clave no va seguida del elemento hermano especificado, se devuelve un error.

salida-xml

Parámetro de salida de tipo BLOB(32MB) que devuelve un documento XML de salida completo en UTF-8. Si se especifica un filtro, este parámetro devuelve un valor de serie. Si el procedimiento almacenado no puede devolver un documento de salida completo (por ejemplo, si se produce un error de proceso que genera un error o aviso de SQL), este parámetro se fija en NULL.

El documento XML de salida contiene información sobre la instancia, incluida información sobre el nivel de fixpack, el release, la información de sistema y las variables de entorno.

mensaje-xml

Parámetro de salida XML completo de tipo BLOB(64K) que devuelve un documento XML de salida completo del tipo de mensaje de servidor de datos en codificación UTF-8 que ofrece información detallada sobre una condición de aviso de SQL. Este documento se devuelve cuando la llamada a un procedimiento genera un aviso de SQL, y el mensaje de aviso indica que se devuelve información adicional en el documento de salida del mensaje XML. Si el mensaje de aviso no especifica que se devuelve información adicional, este parámetro se fija en NULL.

Autorización

- Autorización SYSADM o DBADM
- Privilegio EXECUTE para el procedimiento GET_SYSTEM_INFO

Ejemplo

Ejemplo 1: Se devuelve la versión más reciente del procedimiento.

```
db2 "call sysproc.get_system_info(null,null,null,null,null,?,?)"
```

El ejemplo siguiente muestra los datos de salida de la consulta:

Valor parámetros de salida

```
-----  
Nombre de parámetro : VERSIÓN-PRINCIPAL  
Valor de parámetro  : 1  
  
Nombre de parámetro : VERSIÓN-SECUNDARIA  
Valor de parámetro  : 0  
  
Nombre de parámetro : SALIDA-XML  
Valor de parámetro  : -
```


Nombre de parámetro : MENSAJE-XML
Valor de parámetro : -

Estado de devolución = 0

Ejemplo 2: Se devuelve la información de sistema.

```
db2 "call sysproc.get_system_info(1,0,'en_US',null,null,?,?)"
```

El ejemplo siguiente muestra los datos de salida de la consulta:

Valor parámetros de salida

Nombre de parámetro : VERSIÓN-PRINCIPAL
Valor de parámetro : 1

Nombre de parámetro : VERSIÓN-SECUNDARIA
Valor de parámetro : 0

Nombre de parámetro : SALIDA-XML
Valor de parámetro : x'3C3F786D6C20766572.....

Nombre de parámetro : MENSAJE-XML
Valor de parámetro : -

Estado de devolución = 0

El contenido del documento XML de salida es similar al siguiente:

```
<plist version="1.0">
<dict><key>Document Type Name</key><string>Data Server System Output</string>
  <key>Document Type Major Version</key><integer>1</integer>
  <key>Document Type Minor Version</key><integer>0</integer>
  <key>Data Server Product Name</key><string>QDB2/AIX64</string>
  <key>Data Server Product Version</key><string>9.7.0.0</string>
  <key>Data Server Major Version</key><integer>9</integer>
  <key>Data Server Minor Version</key><integer>7</integer>
  <key>Data Server Platform</key><string>AIX 64BIT</string>
  <key>Document Locale</key><string>en_US</string>
  <key>Instance Information</key>
  <dict>
    <key>Display Name</key><string>Información de instancia</string>
    <key>Instance Name</key>
    <dict>
      <key>Display Name</key><string>Nombre de instancia</string>
      <key>Value</key><string>myinstance</string>
      <key>Hint</key><string></string>
    </dict>
    <key>Partitionable State</key>
    <dict>
      <key>Display Name</key><string>Estado particionable</string>
      <key>Value</key><integer>0</integer>
      <key>Hint</key><string></string>
    </dict>
    <key>Number of Database Partitions</key>
    <dict>
      <key>Display Name</key><string>Número de particiones de base de datos</string>
      <key>Value</key><integer>1</integer>
      <key>Hint</key><string></string>
    </dict>
    .
    .
  </dict>
<key>Product Information</key>
<dict>
  <key>Display Name</key><string>Información de producto</string>
  .
</dict>
```

```

.
.
<key>DB2_ENTERPRISE_SERVER_EDITION</key>
<dict>
  <key>Display Name</key><string>DB2_ENTERPRISE_SERVER_EDITION</string>
  <key>Product short name</key>
  <dict>
    <key>Display Name</key><string>Nombre corto del producto</string>
    <key>Value</key><string>ESE</string>
    <key>Hint</key><string></string>
  </dict>
  <key>Licence</key>
  <dict>
    <key>Display Name</key><string>Licencia</string>
    <key>Value</key><string>Y</string>
    <key>Hint</key><string></string>
  </dict>
  <key>Product Release</key>
  <dict>
    <key>Display Name</key><string>Release del producto</string>
    <key>Value</key><string>9.7</string>
    <key>Hint</key><string></string>
  </dict>
  <key>Licence type</key>
  <dict>
    <key>Display Name</key><string>Tipo de licencia</string>
    <key>Value</key><string>DEVELOPER</string>
    <key>Hint</key><string></string>
  </dict>
  <key>Hint</key><string></string>
</dict>
.
.
<key>Operating System Information</key>
<dict>
  <key>Display Name</key><string>Información de sistema operativo</string>
  <key>Name</key>
  <dict>
    <key>Display Name</key><string>Nombre</string>
    <key>Value</key><string>AIX</string>
    <key>Hint</key><string></string>
  </dict>
  <key>Version</key>
  <dict>
    <key>Display Name</key><string>Versión</string>
    <key>Value</key><string>5</string>
    <key>Hint</key><string></string>
  </dict>
  <key>Release</key>
  <dict>
    <key>Display Name</key><string>Release</string>
    <key>Value</key><string>3</string>
    <key>Hint</key><string></string>
  </dict>
  <key>Hostname</key>
  <dict>
    <key>Display Name</key><string>Nombre de sistema principal</string>
    <key>Value</key><string>achilles</string>
    <key>Hint</key><string></string>
  </dict>
.
.
</dict>
<key>Workload Management Configuration</key>
<dict>

```

```

<key>Display Name</key><string>Configuración de gestión de cargas de trabajo</string>
<key>Service Class Information</key>
<dict>
  <key>Display Name</key><string>Información de clase de servicio</string>
  <key>1</key>
  <dict>
    <key>Display Name</key><string>1</string>
    <key>Service Class Name</key>
    <dict>
      <key>Display Name</key><string>Nombre de clase de servicio</string>
      <key>Value</key><string>SYSDEFAULTSYSTEMCLASS</string>
      <key>Hint</key><string></string>
    </dict>
  </dict>
  <key>Parent Identifier</key>
  <dict>
    <key>Display Name</key><string>Identificador de padre</string>
    <key>Value</key><integer>0</integer>
    <key>Hint</key><string></string>
  </dict>
  <key>Parent Class Name</key>
  <dict>
    <key>Display Name</key><string>Nombre de clase padre</string>
    <key>Value</key><string></string>
    <key>Hint</key><string></string>
  </dict>
  <key>Creation Time</key>
  <dict>
    <key>Display Name</key><string>Hora de creación</string>
    <key>Value</key><string>2008-04-21-15.14.32.956930</string>
    <key>Hint</key><string></string>
  </dict>
  <key>Alter Time</key>
  <dict>
    <key>Display Name</key><string>Hora alternativa</string>
    <key>Value</key><string>2008-04-21-15.14.32.956930</string>
    <key>Hint</key><string></string>
  </dict>
  <key>Enabled</key>
  <dict>
    <key>Display Name</key><string>Habilitado</string>
    <key>Value</key><string>Y</string>
    <key>Hint</key><string></string>
  </dict>
  <key>Agent Priority</key>
  <dict>
    <key>Display Name</key><string>Prioridad de agente</string>
    <key>Value</key><integer>-32768</integer>
    <key>Hint</key><string></string>
  </dict>
  <key>Prefetcher Priority</key>
  <dict>
    <key>Display Name</key><string>Prioridad de captador previo</string>
    <key>Value</key><string> </string>
    <key>Hint</key><string></string>
  </dict>
  .
  .
  .
</dict>
.
.
.

<key>Workload Information</key>
<dict>
  <key>Display Name</key><string>Información de carga de trabajo</string>
  <key>1</key>

```

```

<dict>
  <key>Display Name</key><string>1</string>
  <key>Workload Name</key>
  <dict>
    <key>Display Name</key><string>Nombre de carga de trabajo</string>
    <key>Value</key><string>SYSDEFAULTUSERWORKLOAD</string>
    <key>Hint</key><string></string>
  </dict>
  <key>Evaluation Order</key>
  <dict>
    <key>Display Name</key><string>Orden de evaluación</string>
    <key>Value</key><integer>1</integer>
    <key>Hint</key><string></string>
  </dict>
  <key>Creation Time</key>
  <dict>
    <key>Display Name</key><string>Hora de creación</string>
    <key>Value</key><string>2008-04-21-15.14.32.955296</string>
    <key>Hint</key><string></string>
  </dict>
  <key>Alter Time</key>
  <dict>
    <key>Display Name</key><string>Hora alternativa</string>
    <key>Value</key><string>2008-04-21-15.14.32.955296</string>
    <key>Hint</key><string></string>
  </dict>
  <key>Enabled</key>
  <dict>
    <key>Display Name</key><string>Habilitado</string>
    <key>Value</key><string>Y</string>
    <key>Hint</key><string></string>
  </dict>
  <key>Allow Access</key>
  <dict>
    <key>Display Name</key><string>Permitir acceso</string>
    <key>Value</key><string>Y</string>
    <key>Hint</key><string></string>
  </dict>
  <key>Service Class Name</key>
  <dict>
    <key>Display Name</key><string>Nombre de clase de servicio</string>
    <key>Value</key><string>SYSDEFAULTSUBCLASS</string>
    <key>Hint</key><string></string>
  </dict>
  <key>Parent Service Class Name</key>
  <dict>
    <key>Display Name</key><string>Nombre de clase de servicio padre</string>
    <key>Value</key><string>SYSDEFAULTUSERCLASS</string>
    <key>Hint</key><string></string>
  </dict>
  .
  .
  .
</dict>
  <key>Hint</key><string></string>
</dict>
</dict>
</dict></dict></dict></plist>

```

Ejemplo 3: Se llama al procedimiento GET_SYSTEM_INFO y se pasa un entorno local no soportado.

```
db2 "call sysproc. get_system_info(1,0,'ja_JP',null,null,?,?)"
```

El ejemplo siguiente muestra los datos de salida de la consulta:

```

Valor parámetros de salida
-----
Nombre de parámetro : VERSIÓN-PRINCIPAL
Valor de parámetro  : 1

Nombre de parámetro : VERSIÓN-SECUNDARIA
Valor de parámetro  : 0

Nombre de parámetro : SALIDA-XML
Valor de parámetro  : x'3C3F786D6C20766572.....

Nombre de parámetro : MENSAJE-XML
Valor de parámetro  : -

Estado de devolución = 0

```

SQL20461W El procedimiento "SYSPROC.GET_SYSTEM_INFO" ha devuelto salida en el entorno local alternativo, "en_US", en vez de en el entorno local, "ja_JP", especificado en el parámetro "3". SQLSTATE=01H57

El contenido del documento XML de salida será el mismo que el del ejemplo 2.

Ejemplo 4: Se llama al procedimiento desde una función.

```

EXEC SQL BEGIN DECLARE SECTION;
sqlint16 getSysInfMaj;
sqlint16 getSysInfMin;

SQL TYPE IS BLOB(2M) xmlOutput;
SQL TYPE IS BLOB(2K) xmlOutMessage;
EXEC SQL END DECLARE SECTION;
getSysInfMaj = 1;
getSysInfMin = 0;

EXEC SQL CALL SYSPROC.GET_SYSTEM_INFO(
    :getSysInfMaj,
    :getSysInfMin,
    'en_US',
    null,
    null,
    :xmlOutput,
    :xmlOutMessage );

```

Procedimiento SET_CONFIG - Establecer parámetros de configuración

El procedimiento almacenado SET_CONFIG actualiza los parámetros de configuración del gestor de bases de datos y de la base de datos que devuelve el procedimiento GET_CONFIG.

El procedimiento SET_CONFIG acepta un documento XML de entrada que contiene los parámetros de configuración y sus valores, utiliza esta información para actualizar los parámetros de configuración especificados y devuelve un documento XML de salida que indica el estado de actualización de cada parámetro de configuración.

Sintaxis

```

▶▶SET_CONFIG(—versión-principal—,—versión-secundaria—,—entorno-local-solicitado—,—▶▶
▶—entrada-xml—,—filtro-xml—,—salida-xml—,—mensaje-xml—)▶▶

```

El esquema es SYSPROC.

Parámetros del procedimiento

versión-principal

Argumento de entrada y salida de tipo INTEGER que indica la versión principal del documento. En la entrada, este argumento indica la versión principal del documento a la que da soporte el emisor de la llamada para los documentos XML pasados como parámetros en el procedimiento (consulte las descripciones de parámetros de *entrada-xml*, *salida-xml*, y *mensaje-xml*). El procedimiento procesa todos los documentos XML en la versión especificada y devuelve un error (+20458) si la versión no es válida. En la salida, este parámetro especifica la versión principal del documento más reciente a la que da soporte el procedimiento. Para determinar la versión de documento más reciente soportada, especifique NULL para este parámetro de entrada y el resto de parámetros requeridos.

Si el documento XML del parámetro *entrada-xml* especifica una clave Document Type Major Version, el valor de dicha clave debe ser igual al valor proporcionado en el parámetro *versión-principal*, o se emite un error (+20458).

Versiones que reciben soporte: 1

versión-secundaria

Argumento de entrada y salida de tipo INTEGER que indica la versión secundaria del documento. En la entrada, este argumento especifica la versión secundaria del documento a la que da soporte el emisor de la llamada para los documentos XML pasados como parámetros para este procedimiento (consulte las descripciones de parámetros de *entrada-xml*, *salida-xml*, y *mensaje-xml*). El procedimiento procesa todos los documentos XML en la versión especificada y devuelve un error si la versión no es válida. En la salida, este parámetro indica la versión secundaria del documento más reciente que se soporta para la versión principal soportada más reciente. Para determinar la versión de documento más reciente soportada, especifique NULL para este parámetro de entrada y el resto de parámetros requeridos.

Si el documento XML del parámetro *entrada-xml* especifica una clave Document Type Minor Version, el valor de dicha clave debe ser igual al valor proporcionado en el parámetro *versión-secundaria*, o se emite un error (+20458).

Versiones que reciben soporte: 0

entorno-local-solicitado

Argumento de entrada de tipo VARCHAR(33) que especifica un entorno local. Si el servidor da soporte al idioma especificado, se devuelve contenido traducido en los parámetros *salida-xml* y *mensaje-xml*. De lo contrario, el contenido se devuelve en el idioma por omisión. Sólo se utiliza el idioma y en ocasiones la información de territorio procedente del entorno local. El entorno local no se utiliza para dar formato a los números o influir en la codificación del documento. Por ejemplo, los valores y los nombres de claves no se traducen. La única parte traducida de los documentos de mensaje XML y salida XML son los textos de las sugerencias, el nombre de visualización y la unidad de visualización de cada entrada. El emisor de la llamada debe comparar siempre el idioma solicitado con el idioma que se está utilizando en el documento XML de salida (consulte la entrada de entorno local de documento en el documento XML de salida).

Actualmente, el único valor soportado para *entorno-local-solicitado* es en_US.

entrada-xml

Un argumento de entrada de tipo BLOB(32MB) que especifica un documento XML de entrada (con codificación UTF-8) que contiene los valores de entrada para el procedimiento.

Para este procedimiento, el documento XML de entrada contiene los valores de configuración de la base de datos y del gestor de bases de datos.

filtro-xml

Argumento de entrada de tipo BLOB(4K) que especifica una serie de consulta XPath válida. Utilice un filtro si desea recuperar un único valor de un documento XML de salida. Para obtener más información, consulte el tema que describe el filtrado XPath.

El ejemplo siguiente selecciona el valor para un parámetro de configuración específico del documento XML de salida: `/plist/dict/key[.="Valores del parámetro de configuración del gestor de bases de datos"]/following-sibling::dict[1]/key[3]/following-sibling::dict[1]/dict[1]/key[.="Value"]/following-sibling::string[1]`. Si la clave no va seguida del elemento hermano especificado, se devuelve un error.

salida-xml

Parámetro de salida de tipo BLOB(32MB) que devuelve un documento XML de salida completo en UTF-8. Si se especifica un filtro, este parámetro devuelve un valor de serie. Si el procedimiento almacenado no puede devolver un documento de salida completo (por ejemplo, si se produce un error de proceso que genera un error o aviso de SQL), este parámetro se fija en NULL.

Cuando este procedimiento funciona en la modalidad *completa*, este parámetro devuelve un documento XML que contiene los valores de configuración actuales determinados en el servidor. Puede modificar este documento y pasarlo de vuelta al procedimiento como parámetro *entrada-xml*. Este método ofrece un enfoque programático para la creación de documentos de entrada XML válidos.

mensaje-xml

Parámetro de salida XML completo de tipo BLOB(64K) que devuelve un documento XML de salida completo del tipo de mensaje de servidor de datos en codificación UTF-8 que ofrece información detallada sobre una condición de aviso de SQL. Este documento se devuelve cuando la llamada a un procedimiento genera un aviso de SQL, y el mensaje de aviso indica que se devuelve información adicional en el documento de salida del mensaje XML. Si el mensaje de aviso no especifica que se devuelve información adicional, este parámetro se fija en NULL.

Autorización

- Autorización SYSADM o DBADM
- Privilegio EXECUTE para el procedimiento SET_CONFIG

Ejemplo

Ejemplo 1: Se devuelve la versión más reciente del procedimiento.

```
db2 "call sysproc.set_config (null,null,null,null,null,?,?)"
```

El ejemplo siguiente muestra los datos de salida de la consulta:

Valor parámetros de salida

Nombre de parámetro : VERSIÓN-PRINCIPAL

Valor de parámetro : 1

Nombre de parámetro : VERSIÓN-SECUNDARIA
Valor de parámetro : 0

Nombre de parámetro : SALIDA-XML
Valor de parámetro : -

Nombre de parámetro : MENSAJE-XML
Valor de parámetro : -

Estado de devolución = 0

Ejemplo 2: Se ejecuta un script denominado setconfig.sql que actualiza algunos parámetros de configuración de la base de datos y el gestor de bases de datos.

setconfig.sql:

```
call sysproc.set_config(1,0,'en_US',blob('
<?xml version="1.0" encoding="UTF-8"?>
<plist version="1.0">
<dict>
  <key>Document Type Name</key><string>Data Server Set Configuration Input</string>
  <key>Document Type Major Version</key><integer>1</integer>
  <key>Document Type Minor Version</key><integer>0</integer>
  <key>Document Locale</key><string>en_US</string>
  <key>Database Manager Configuration Parameter Settings</key>
  <dict>
    <key>diaglevel</key><dict><key>Parameter Value</key>
      <dict>
        <key>Value</key><string>4</string>
      </dict>
    </dict>
  <key>fcm_num_buffers</key>
  <dict>
    <key>Parameter Value</key>
    <dict>
      <key>Value</key><string>4096</string>
    </dict>
    <key>Value Flags</key>
    <dict>
      <key>Value</key><string>MANUAL</string>
    </dict>
  </dict>
  <key>instance_memory</key>
  <dict>
    <key>Deferred Value</key>
    <dict>
      <key>Value</key><string>7424</string>
    </dict>
    <key>Deferred Value Flags</key>
    <dict>
      <key>Value</key><string>AUTOMATIC</string>
    </dict>
  </dict>
</dict>
</dict>
<key>Database Partition</key>
<dict>
  <key>All</key>
  <dict>
    <key>Database Configuration Parameter Settings</key>
    <dict>
      <key>avg_appls</key>
      <dict>
        <key>Parameter Value</key>
        <dict>
          <key>Value</key><string>2</string>
        </dict>
      </dict>
    </dict>
  </dict>
</dict>
</dict>
```



```

        <key>Value Flags</key>
        <dict>
          <key>Value</key><string>AUTOMATIC</string>
        </dict>
      </dict>
    <key>database_memory</key>
    <dict>
      <key>Deferred Value</key>
      <dict>
        <key>Value</key><string>2</string>
      </dict>
      <key>Deferred Value Flags</key>
      <dict>
        <key>Value</key><string>MANUAL</string>
      </dict>
    </dict>
  </dict>
</dict>
</plist>'), null, ?,?)@

```

El siguiente ejemplo muestra una salida de esta consulta:

Valor parámetros de salida

```

-----
Nombre de parámetro : VERSIÓN-PRINCIPAL
Valor de parámetro  : 1

```

```

Nombre de parámetro : VERSIÓN-SECUNDARIA
Valor de parámetro  : 0

```

```

Nombre de parámetro : SALIDA-XML
Valor de parámetro  : x'3C3F78...'

```

```

Nombre de parámetro : MENSAJE-XML
Valor de parámetro  : -

```

Estado de devolución = 0

El contenido del documento XML de salida es similar al siguiente:

```

<?xml version="1.0" encoding="UTF-8"?>
<plist version="1.0">
<dict>
  <key>Document Type Name</key>
  <string>Data Server Set Configuration Output</string>
  <key>Document Type Major Version</key><integer>1</integer>
  <key>Document Type Minor Version</key><integer>0</integer>
  <key>Data Server Product Name</key><string>QDB2/AIX64</string>
  <key>Data Server Product Version</key><string>9.7.0.0</string>
  <key>Data Server Major Version</key><integer>9</integer>
  <key>Data Server Minor Version</key><integer>7</integer>
  <key>Data Server Platform</key><string>AIX 64BIT</string>
  <key>Document Locale</key><string>en_US</string>
  <key>Database Manager Configuration Parameter Settings</key>
  <dict>
    <key>Display Name</key>
    <string>Valores del parámetro de configuración del gestor de bases de datos</string>
    <key>diaglevel</key>
    <dict>
      <key>Display Name</key><string>niveldiag</string>
      <key>Parameter Value</key>
      <dict>
        <key>Display Name</key><string>Valor de parámetro</string>
        <key>Value</key><string>4</string>
      </dict>
    <key>Parameter Update Status</key>

```

```

<dict>
  <key>Display Name</key><string>Estado de actualización de parámetros</string>
  <key>SQLCODE</key>
  <dict>
    <key>Display Name</key><string>SQLCODE</string>
    <key>Value</key><integer>0</integer>
  </dict>
  <key>Message Tokens</key>
  <dict>
    <key>Display Name</key><string>Símbolos de mensaje</string>
    <key>Value</key><array><string></string></array>
  </dict>
  <key>SQLSTATE</key>
  <dict>
    <key>Display Name</key><string>SQLSTATE</string>
    <key>Value</key><string></string>
  </dict>
</dict>
</dict>
<key>fcm_num_buffers</key>
<dict>
  <key>Display Name</key><string>fcm_num_buffers</string>
  <key>Parameter Value</key>
  <dict>
    <key>Display Name</key><string>Valor de parámetro</string>
    <key>Value</key><string>4096</string>
  </dict>
  <key>Value Flags</key>
  <dict>
    <key>Display Name</key><string>Distintivos de valor</string>
    <key>Value</key><string>MANUAL</string>
  </dict>
  <key>Parameter Update Status</key>
  <dict>
    <key>Display Name</key><string>Estado de actualización de parámetros</string>
    <key>SQLCODE</key><dict>
      <key>Display Name</key><string>SQLCODE</string>
      <key>Value</key> <integer>0</integer>
    </dict>
    <key>Message Tokens</key>
    <dict>
      <key>Display Name</key><string>Símbolos de mensaje</string>
      <key>Value</key><array><string></string></array>
    </dict>
    <key>SQLSTATE</key>
    <dict>
      <key>Display Name</key><string>SQLSTATE</string>
      <key>Value</key><string></string>
    </dict>
  </dict>
</dict>
</dict>
<key>instance_memory</key>
<dict>
  <key>Display Name</key><string>memoria_instancia</string>
  <key>Deferred Value</key>
  <dict>
    <key>Display Name</key><string>Valor diferido</string>
    <key>Value</key><string>7424</string>
  </dict>
  <key>Deferred Value Flags</key>
  <dict>
    <key>Display Name</key><string>Distintivos de valor diferido</string>
    <key>Value</key><string>AUTOMATIC</string>
  </dict>
  <key>Parameter Update Status</key>
  <dict>
    <key>Display Name</key><string>Estado de actualización de parámetros</string>

```

```

<key>SQLCODE</key>
<dict>
  <key>Display Name</key><string>SQLCODE</string>
  <key>Value</key><integer>0</integer>
</dict>
<key>Message Tokens</key>
<dict>
  <key>Display Name</key><string>Símbolos de mensaje</string>
  <key>Value</key><array><string></string></array>
</dict>
<key>SQLSTATE</key>
<dict>
  <key>Display Name</key><string>SQLSTATE</string>
  <key>Value</key><string></string>
</dict>
</dict>
</dict>
<key>Database Partition</key>
<dict>
  <key>Display Name</key><string>Partición de base de datos</string>
  <key>All</key>
<dict>
  <key>Display Name</key><string>Todo</string>
<key>Database Configuration Parameter Settings</key>
<dict>
  <key>Display Name</key>
  <string>Valores del parámetro de configuración de base de datos</string>
  <key>avg_appls</key>
  <dict>
    <key>Display Name</key><string>avg_appls</string>
    <key>Parameter Value</key>
    <dict>
      <key>Display Name</key><string>Valor de parámetro</string>
      <key>Value</key><string>2</string>
    </dict>
  </dict>
  <key>Value Flags</key>
  <dict>
    <key>Display Name</key><string>Distintivos de valor</string>
    <key>Value</key><string>AUTOMATIC</string>
  </dict>
  <key>Parameter Update Status</key>
  <dict>
    <key>Display Name</key><string>Estado de actualización de parámetros</string>
    <key>Update Coverage</key>
    <dict>
      <key>Display Name</key><string>Cobertura de actualización</string>
      <key>Value</key><string>Complete</string>
    </dict>
  </dict>
  <key>SQLCODE</key>
  <dict>
    <key>Display Name</key><string>SQLCODE</string>
    <key>Value</key><integer>0</integer>
  </dict>
  <key>Message Tokens</key>
  <dict>
    <key>Display Name</key><string>Símbolos de mensaje</string>
    <key>Value</key><array><string></string> </array>
  </dict>
  <key>SQLSTATE</key>
  <dict>
    <key>Display Name</key><string>SQLSTATE</string>
    <key>Value</key><string></string>
  </dict>
</dict>
</dict>
<key>database_memory</key>

```

```

<dict>
  <key>Display Name</key><string>memoria-base-datos</string>
  <key>Deferred Value</key>
  <dict>
    <key>Display Name</key><string>Valor diferido</string>
    <key>Value</key><string>2</string>
  </dict>
  <key>Deferred Value Flags</key>
  <dict>
    <key>Display Name</key><string>Distintivos de valor diferido</string>
    <key>Value</key><string>MANUAL</string>
  </dict>
  <key>Parameter Update Status</key>
  <dict>
    <key>Display Name</key><string>Estado de actualización de parámetros</string>
    <key>Update Coverage</key>
    <dict>
      <key>Display Name</key><string>Cobertura de actualización</string>
      <key>Value</key><string>Complete</string>
    </dict>
    <key>SQLCODE</key>
    <dict>
      <key>Display Name</key><string>SQLCODE</string>
      <key>Value</key><integer>0</integer>
    </dict>
    <key>Message Tokens</key>
    <dict>
      <key>Display Name</key><string>Símbolos de mensaje</string>
      <key>Value</key><array><string></string></array>
    </dict>
    <key>SQLSTATE</key>
    <dict>
      <key>Display Name</key><string>SQLSTATE</string>
      <key>Value</key><string></string>
    </dict>
  </dict>
</dict>
</dict>
</dict>
</dict>
</dict>
</plist>

```

Ejemplo 3: Se especifica un filtro para devolver el valor de un parámetro de configuración específico.

```

db2 "call sysproc.set_config(1,0, 'en_US', blob('
<plist version="1.0">
<dict>
  <key>Document Type Name</key><string>Data Server Set Configuration Input</string>
  <key>Document Type Major Version</key><integer>1</integer>
  <key>Document Type Minor Version</key><integer>0</integer>
  <key>Document Locale</key><string>en_US</string>
  <key>Database Manager Configuration Parameter Settings</key>
  <dict>
    <key>diaglevel</key>
    <dict>
      <key>Parameter Value</key>
      <dict>
        <key>Value</key><string>4</string>
      </dict>
    </dict>
    <key>fcm_num_buffers</key>
    <dict>
      <key>Parameter Value</key>
      <dict>
        <key>Value</key><string>4096</string>

```

```

        </dict>
        <key>Value Flags</key>
        <dict>
            <key>Value</key><string>MANUAL</string>
        </dict>
    </dict>
    <key>instance_memory</key>
    <dict>
        <key>Deferred Value</key>
        <dict>
            <key>Value</key><string>7424</string>
        </dict>
        <key>Deferred Value Flags</key>
        <dict>
            <key>Value</key><string>AUTOMATIC</string>
        </dict>
    </dict>
</dict>
<key>Database Partition</key>
<dict>
    <key>All</key>
    <dict>
        <key>Database Configuration Parameter Settings</key>
        <dict>
            <key>avg_appls</key>
            <dict>
                <key>Parameter Value</key>
                <dict>
                    <key>Value</key><string>2</string>
                </dict>
                <key>Value Flags</key>
                <dict>
                    <key>Value</key><string>AUTOMATIC</string>
                </dict>
            </dict>
            <key>database_memory</key>
            <dict>
                <key>Deferred Value</key>
                <dict>
                    <key>Value</key><string>2</string>
                </dict>
                <key>Deferred Value Flags</key>
                <dict>
                    <key>Value</key><string>MANUAL</string>
                </dict>
            </dict>
        </dict>
    </dict>
</dict>
</plist>'),
blob('/plist/dict/key[.="Database Manager Configuration Parameter Settings"]
/following-sibling::dict[1]/key[3]/following-sibling::dict[1]/dict[1]/key[.="Value"]
/following-sibling::string[1]'),?,?)"
```

El siguiente ejemplo muestra una salida de esta consulta:

Valor parámetros de salida

```
-----
Nombre de parámetro : VERSIÓN-PRINCIPAL
Valor de parámetro  : 1
```

```
Nombre de parámetro : VERSIÓN-SECUNDARIA
Valor de parámetro  : 0
```

```
Nombre de parámetro : SALIDA-XML
Valor de parámetro  : x'34303936'
```

Nombre de parámetro : MENSAJE-XML

Valor de parámetro : -

Estado de devolución = 0

Se devuelve el valor siguiente para *salida-xml*:

4096

Ejemplo 4: Se llama al procedimiento desde una función.

```
EXEC SQL BEGIN DECLARE SECTION;
sqlint16 getconfigMaj;
sqlint16 getconfigMin;

SQL TYPE IS BLOB(2M) xmlOutput;
SQL TYPE IS BLOB(2K) xmlOutMessage;
EXEC SQL END DECLARE SECTION;
getconfigMaj = 1;
getconfigMin = 0;

EXEC SQL CALL SYSPROC.SET_CONFIG(
: getconfigMaj,
: getconfigMin,
' en_US',
BLOB(' blob('
<?xml version="1.0" encoding="UTF-8"?>
<plist version="1.0">
<dict>
<key>Document Type Name</key>
<string>Data Server Set Configuration Input</string>
<key>Document Type Major Version</key><integer>1</integer>
<key>Document Type Minor Version</key><integer>0</integer>
<key>Document Locale</key><string>en_US</string>
<key>Database Manager Configuration Parameter Settings</key>
<dict>
<key>diaglevel</key><dict><key>Parameter Value</key>
<dict>
<key>Value</key><string>4</string>
</dict>
</dict>
</dict>
<key>fcm_num_buffers</key>
<dict>
<key>Parameter Value</key>
<dict>
<key>Value</key><string>4096</string>
</dict>
<key>Value Flags</key>
<dict>
<key>Value</key><string>MANUAL</string>
</dict>
</dict>
<key>instance_memory</key>
<dict>
<key>Deferred Value</key>
<dict>
<key>Value</key><string>7424</string>
</dict>
<key>Deferred Value Flags</key>
<dict>
<key>Value</key><string>AUTOMATIC</string>
</dict>
</dict>
</dict>
</dict>
<key>Database Partition</key>
<dict>
<key>All</key>
```

```

<dict>
  <key>Database Configuration Parameter Settings</key>
  <dict>
    <key>avg_appls</key>
    <dict>
      <key>Parameter Value</key>
      <dict>
        <key>Value</key><string>2</string>
      </dict>
      <key>Value Flags</key>
      <dict>
        <key>Value</key><string>AUTOMATIC</string>
      </dict>
    </dict>
    <key>database_memory</key>
    <dict>
      <key>Deferred Value</key>
      <dict>
        <key>Value</key><string>2</string>
      </dict>
      <key>Deferred Value Flags</key>
      <dict>
        <key>Value</key><string>MANUAL</string>
      </dict>
    </dict>
  </dict>
</dict>
</plist>'),
null,
:xmlOutput,
:xmlOutMessage );

```

Capítulo 9. Rutinas y vistas de configuración

DB_PARTITIONS

La función de tabla DB_PARTITIONS devuelve el contenido del archivo db2nodes.cfg en formato de tabla.

Sintaxis

►►—DB_PARTITIONS—(—)—◄◄

El esquema es SYSPROC.

Autorización

Privilegio EXECUTE para la función de tabla DB_PARTITIONS.

Parámetros de la función de tabla

La función no tiene parámetros de entrada.

Ejemplo

Recuperar información desde una base de datos de 3 particiones lógicas.

```
SELECT * FROM TABLE(DB_PARTITIONS()) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| PARTITION_NUMBER | HOST_NAME | PORT_NUMBER | SWITCH_NAME |
|------------------|--------------------------|-------------|-------------|
| 0 | jessicae.torolab.ibm.com | 0 | jessicae |
| 1 | jessicae.torolab.ibm.com | 1 | jessicae |
| 2 | jessicae.torolab.ibm.com | 2 | jessicae |

3 registro(s) seleccionado(s).

Información devuelta

Tabla 82. Información devuelta por la función de tabla DB_PARTITIONS

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| PARTITION_NUMBER | SMALLINT | Número exclusivo comprendido entre 0 y 999 que identifica un servidor de partición de base de datos en un entorno de bases de datos particionadas. |
| HOST_NAME | VARCHAR(128) | El nombre de sistema principal TCP/IP del servidor de partición de base de datos. |
| PORT_NUMBER | SMALLINT | Número de puerto del servidor de partición de base de datos. |

Tabla 82. Información devuelta por la función de tabla DB_PARTITIONS (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| SWITCH_NAME | VARCHAR(128) | Nombre de una interconexión o conmutador de alta velocidad para las comunicaciones de partición de base de datos. |

Vista administrativa DBCFG - Recuperar información sobre los parámetros de configuración de la base de datos

La vista administrativa DBCFG devuelve información sobre parámetros de configuración de base de datos correspondiente a la base de datos conectada actualmente para todas las particiones de base de datos.

El esquema es SYSIBMADM.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa DBCFG
- Privilegio CONTROL en la vista administrativa DBCFG
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE para la función de tabla DB_GET_CFG
- Autorización DATAACCESS

Ejemplos

Ejemplo 1: Recuperar los valores de mantenimiento automático de la configuración de base de datos que están almacenados en memoria para todas las particiones de base de datos.

```
SELECT DBPARTITIONNUM, NAME, VALUE FROM SYSIBMADM.DBCFG WHERE NAME LIKE 'auto_%'
```

El siguiente ejemplo muestra los datos de salida de la consulta.

```
DBPARTITIONNUM NAME VALUE
-----
0 auto_maint OFF
0 auto_db_backup OFF
0 auto_tbt_maint OFF
0 auto_runstats OFF
0 auto_stats_prof OFF
0 auto_prof_upd OFF
0 auto_reorg OFF
0 autorestart ON
```

8 registro(s) seleccionado(s).

Ejemplo 2: Recuperar los valores de todos los parámetros de configuración de base de datos almacenados en disco para todas las particiones de base de datos.

```
SELECT NAME, DEFERRED_VALUE, DBPARTITIONNUM FROM SYSIBMADM.DBCFG
```

El siguiente ejemplo muestra los datos de salida de la consulta.

| NAME | DEFERRED_VALUE | DBPARTITIONNUM |
|-----------------|----------------|----------------|
| appctl_heap_sz | 128 | 0 |
| appgroup_mem_sz | 30000 | 0 |
| applheapsz | 256 | 0 |
| archretrydelay | 20 | 0 |
| ... | | |
| autorestart | ON | 0 |
| avg_appls | 1 | 0 |
| blk_log_dsk_ful | NO | 0 |
| catalogcache_sz | -1 | 0 |
| ... | | |

Información devuelta

Tabla 83. Información devuelta por la vista administrativa DBCFG

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| NAME | VARCHAR(32) | Nombre del parámetro de configuración. |
| VALUE | VARCHAR(1024) | El valor actual del parámetro de configuración almacenado en memoria. |
| VALUE_FLAGS | VARCHAR(10) | Proporciona información específica para el valor actual del parámetro de configuración. Los valores válidos son: <ul style="list-style-type: none"> • NONE - no hay información adicional • AUTOMATIC - el parámetro de configuración se ha establecido como automático |
| DEFERRED_VALUE | VARCHAR(1024) | El valor del parámetro de configuración en disco. Para algunos parámetros de configuración de base de datos, los cambios sólo entran en vigor cuando la base de datos se vuelve a activar. En estos casos, primero se tienen que desconectar todas las aplicaciones de la base de datos. (Si la base de datos se había activado, se debe desactivar y volver a activar.) Los cambios entran en vigor en la siguiente conexión con la base de datos. |

Tabla 83. Información devuelta por la vista administrativa DBCFG (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|----------------------|---------------|---|
| DEFERRED_VALUE_FLAGS | VARCHAR(10) | Proporciona información específica para el valor diferido del parámetro de configuración. Los valores válidos son: <ul style="list-style-type: none"> • NONE - no hay información adicional • AUTOMATIC - el parámetro de configuración se ha establecido como automático |
| DATATYPE | VARCHAR(128) | Tipo de datos del parámetro de configuración. |
| DBPARTITIONNUM | SMALLINT | Número de partición de base de datos. |

Vista administrativa DBMCFG - Recuperar información sobre los parámetros de configuración del gestor de bases de datos

La vista administrativa DBMCFG devuelve información sobre parámetros de configuración del gestor de bases de datos, incluidos los valores en memoria y los valores almacenados en disco.

El esquema es SYSIBMADM.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT sobre la vista administrativa DBMCFG
- Privilegio CONTROL sobre la vista administrativa DBMCFG
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla DBM_GET_CFG
- Autorización DATAACCESS

Ejemplos

Ejemplo 1: Recuperar valores correspondientes a los parámetros de configuración del gestor de bases de datos almacenados en disco:

```
SELECT NAME, DEFERRED_VALUE FROM SYSIBMADM.DBMCFG
```

El siguiente ejemplo muestra los datos de salida de la consulta.

```
NAME                                DEFERRED_VALUE
-----
agent_stack_sz                       0
agentpri                              -1
alternate_auth_enc                   AES_ONLY
aslheapsz                             15
audit_buf_sz                          0
```

```

authentication          SERVER
catalog_noauth          YES
clnt_krb_plugin
...
comm_bandwidth          0.000000e+00
conn_elapse             0
cpuspeed                4.000000e-05
dft_account_str
dft_mon_bufpool         OFF
...
dft_mon_timestamp       ON
dft_mon_uow             OFF
...
jdk_path                 /wsdb/v91/bldsupp/AIX5L
...
ssl_svcename            22711
ssl_svr_keydb           /GSKit/Keystore/key.kdb
ssl_svr_label
ssl_svr_stash           /GSKit/Keystore/key.sth

```

Ejemplo 2: Recuperar los valores de todos los parámetros de configuración del gestor de bases de datos.

```
SELECT * FROM SYSIBMADM.DBMCFG
```

El siguiente ejemplo muestra los datos de salida de la consulta.

| NAME | VALUE | VALUE_FLAGS | ... |
|--------------------|---------------|-------------|-----|
| agent_stack_sz | 0 | NONE | ... |
| agentpri | -1 | NONE | ... |
| alternate_auth_enc | NOT_SPECIFIED | NONE | |
| aslheapsz | 15 | NONE | ... |
| audit_buf_sz | 0 | NONE | ... |
| authentication | SERVER | NONE | ... |
| catalog_noauth | YES | NONE | ... |
| clnt_krb_plugin | | NONE | ... |
| clnt_pw_plugin | | NONE | ... |
| comm_bandwidth | 0.000000e+00 | NONE | ... |
| conn_elapse | 0 | NONE | ... |
| cpuspeed | 4.000000e-05 | NONE | ... |
| dft_account_str | | NONE | ... |
| dft_mon_bufpool | OFF | NONE | ... |
| dft_mon_lock | OFF | NONE | ... |
| dft_mon_sort | OFF | NONE | ... |
| dft_mon_stmt | OFF | NONE | ... |
| dft_mon_table | OFF | NONE | ... |
| ... | | | ... |
| dir_cache | YES | NONE | ... |
| discover | SEARCH | NONE | ... |
| discover_inst | ENABLE | NONE | ... |
| fcm_num_anchors | 0 | AUTOMATIC | ... |
| fcm_num_buffers | 0 | AUTOMATIC | ... |
| fcm_num_connect | 0 | AUTOMATIC | ... |

Datos de salida de esta consulta (continuación).

| ... DEFERRED_VALUE | DEFERRED_VALUE_FLAGS | DATATYPE |
|--------------------|----------------------|-------------|
| ... 0 | NONE | INTEGER |
| ... -1 | NONE | INTEGER |
| ... AES_ONLY | NONE | VARCHAR(32) |
| ... 15 | NONE | BIGINT |
| ... 0 | NONE | BIGINT |
| ... SERVER | NONE | VARCHAR(32) |
| ... YES | NONE | VARCHAR(3) |
| ... | NONE | VARCHAR(32) |
| ... | NONE | VARCHAR(32) |

| | | |
|------------------|-----------|-------------|
| ... 0.000000e+00 | NONE | REAL |
| ... 0 | NONE | INTEGER |
| ... 4.000000e-05 | NONE | REAL |
| ... | NONE | VARCHAR(25) |
| ... OFF | NONE | VARCHAR(3) |
| ... OFF | NONE | VARCHAR(3) |
| ... OFF | NONE | VARCHAR(3) |
| ... OFF | NONE | VARCHAR(3) |
| ... OFF | NONE | VARCHAR(3) |
| ... | | |
| ... YES | NONE | VARCHAR(3) |
| ... SEARCH | NONE | VARCHAR(8) |
| ... ENABLE | NONE | VARCHAR(8) |
| ... 0 | AUTOMATIC | BIGINT |
| ... 512 | AUTOMATIC | BIGINT |
| ... 0 | AUTOMATIC | BIGINT |
| ... | | |

Información devuelta

Tabla 84. Información devuelta por la vista administrativa DBMCFG

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| NAME | VARCHAR(32) | Nombre del parámetro de configuración. |
| VALUE | VARCHAR(256) | El valor actual del parámetro de configuración almacenado en memoria. |
| VALUE_FLAGS | VARCHAR(10) | Proporciona información específica para el valor actual del parámetro de configuración. Los valores válidos son: <ul style="list-style-type: none"> • NONE - no hay información adicional • AUTOMATIC - el parámetro de configuración se ha establecido como automático |
| DEFERRED_VALUE | VARCHAR(256) | El valor del parámetro de configuración en disco. Para algunos parámetros de configuración del gestor de bases de datos, este se debe detener (db2stop) y volver a iniciar (db2start) para que este valor entre en vigor. |

Tabla 84. Información devuelta por la vista administrativa DBMCFG (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|----------------------|---------------|---|
| DEFERRED_VALUE_FLAGS | VARCHAR(10) | Proporciona información específica para el valor diferido del parámetro de configuración. Los valores válidos son: <ul style="list-style-type: none"> • NONE - no hay información adicional • AUTOMATIC - el parámetro de configuración se ha establecido como automático |
| DATATYPE | VARCHAR(128) | Tipo de datos del parámetro de configuración. |

Vista administrativa REG_VARIABLES - Recuperar valores de registro de DB2 en uso

La vista administrativa REG_VARIABLES devuelve los valores de registro de DB2 procedentes de todas las particiones de bases de datos. Los valores de las variables de registro de DB2 que se devuelven cuando se consulta la vista administrativa REG_VARIABLES pueden diferir de los que devuelve el mandato db2set si una variable de registro de DB2 se ha configurado mediante el mandato db2set una vez iniciada la instancia. La diferencia se produce porque REG_VARIABLES sólo devuelve los valores que estaban en vigor cuando se inició la instancia.

El esquema es SYSIBMADM.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa REG_VARIABLES
- Privilegio CONTROL en la vista administrativa REG_VARIABLES
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE para la función de tabla REG_LIST_VARIABLES
- Autorización DATAACCESS

Ejemplo

Solicitar los valores de registro de DB2 que se utilizan actualmente.

```
SELECT * from SYSIBMADM.REG_VARIABLES
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| DBPARTITIONNUM | REG_VAR_NAME | REG_VAR_VALUE | IS_AGGREGATE | AGGREGATE_NAME |
|----------------|----------------|---------------|--------------|----------------|
| 0 | DB2ADMINSERVER | DB2DAS00 | 0 | - |
| 0 | DB2INSTPROF | D:\SQLLIB | 0 | - |
| 0 | DB2PATH | D:\SQLLIB | 0 | - |
| 0 | DB2SYSTEM | D570 | 0 | - |

```

0 DB2TEMPDIR          D:\SQLLIB\          0 -
0 DB2_EXTSECURITY    YES                 0 -

```

6 registro(s) seleccionado(s).

Información devuelta

Tabla 85. Información devuelta por la vista administrativa REG_VARIABLES

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| DBPARTITIONNUM | SMALLINT | Número de partición lógica de cada partición de base de datos sobre la que actúa la función. |
| REG_VAR_NAME | VARCHAR(256) | Nombre de la variable del registro de DB2. |
| REG_VAR_VALUE | VARCHAR(2048) | Valor actual de la variable de registro de DB2. |
| IS_AGGREGATE | SMALLINT | Indica si la variable del registro de DB2 es un valor de agregado. Los valores de retorno posibles son 0 (si no es una variable de agregado) y 1 (si es una variable de agregado) . |
| AGGREGATE_NAME | VARCHAR(256) | Nombre del agregado si la variable del registro de DB2 obtiene actualmente su valor a partir de un agregado configurado. Si la variable del registro no se define mediante un agregado, o si se define mediante un agregado pero su valor no se tiene en cuenta, el valor de AGGREGATE_NAME es NULL. |
| LEVEL | CHAR(1) | Indica el nivel en el que la variable del registro de DB2 obtiene su valor. Los valores de retorno posibles y los niveles correspondientes que representan son los siguientes: <ul style="list-style-type: none"> • I = instancia • G = global • N = partición de base de datos • E = entorno |

Capítulo 10. Vistas del entorno

Vista administrativa ENV_FEATURE_INFO - Devolver información de licencia para las características de DB2

La vista administrativa ENV_FEATURE_INFO devuelve información sobre todas las características disponibles para las que se necesita una licencia. Para cada característica existe información sobre si se ha instalado o no una licencia válida para la característica.

El esquema es SYSIBMADM.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT sobre la vista administrativa ENV_FEATURE_INFO
- Privilegio CONTROL sobre la vista administrativa ENV_FEATURE_INFO
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla ENV_GET_FEATURE_INFO
- Autorización DATAACCESS

Ejemplo

Solicite la información de licencia de las características de DB2.

```
SELECT * FROM SYSIBMADM.ENV_FEATURE_INFO
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| FEATURE_NAME | FEATURE_FULLNAME | ... |
|--------------|--|-----|
| DPF | DB2_DATABASE_PARTITIONING_FEATURE | ... |
| POESE | DB2_PERFORMANCE_OPTIMIZATION_FEATURE_FOR_ESE | ... |
| SO | DB2_STORAGE_OPTIMIZATION_FEATURE | ... |
| AAC | DB2_ADVANCED_ACCESS_CONTROL_FEATURE | ... |
| GEO | DB2_GEODETTIC_DATA_MANAGEMENT_FEATURE | ... |
| HFESI | IBM_HOMOGENEOUS_FEDERATION_FEATURE_FOR_ESE | ... |
| XMLSI | DB2_PUREXML_FEATURE_FOR_ESE | ... |

Datos de salida de esta consulta (continuación).

| ... | LICENSE_INSTALLED | PRODUCT_NAME | FEATURE_USE_STATUS |
|-----|-------------------|--------------|--------------------|
| ... | Y | ESE | IN_COMPLIANCE |
| ... | Y | ESE | IN_COMPLIANCE |
| ... | Y | ESE | IN_COMPLIANCE |
| ... | Y | ESE | NOT_USED |
| ... | Y | ESE | NOT_USED |
| ... | Y | ESE | NOT_USED |
| ... | N | ESE | IN_VIOLATION |

Metadatos de la vista administrativa ENV_FEATURE_INFO

Tabla 86. Metadatos de la vista administrativa ENV_FEATURE_INFO

| Nombre de columna | Tipo de datos | Descripción |
|--------------------|---------------|---|
| FEATURE_NAME | VARCHAR(26) | Nombres cortos para las características de DB2 disponibles en los servidores de DB2 con licencia. |
| FEATURE_FULLNAME | VARCHAR(100) | Nombre entero de las características de DB2. Los valores de la columna aparecerán en inglés y en mayúsculas. Las palabras se separan con un carácter de subrayado en vez de un carácter de espacio. |
| LICENSE_INSTALLED | CHAR(1) | Indica si la característica tiene licencia. Si el valor es 'N', la característica no tendrá licencia. Si el valor es 'Y', la característica tendrá licencia. |
| PRODUCT_NAME | VARCHAR(26) | Identificadores para el producto DB2 Server en que la característica está disponible. Los valores de retorno posibles son: <ul style="list-style-type: none">• ESE - DB2 Enterprise Server Edition• WSE - DB2 Workgroup Server Edition• EXP - DB2 Express Edition |
| FEATURE_USE_STATUS | VARCHAR(30) | Indica el estado de cumplimiento de la licencia. Este valor indica el estado de uso de la característica. Hay tres valores posibles: <ul style="list-style-type: none">• IN_COMPLIANCE: La característica se ha utilizado al menos una vez y dicha característica tiene una licencia válida.• IN_VIOLATION: La característica se ha utilizado al menos una vez y dicha característica tiene ninguna licencia válida.• NOT_USED: La característica no se ha utilizado. |

Vista administrativa ENV_INST_INFO - Recuperar información sobre la instancia actual

La vista administrativa ENV_INST_INFO devuelve información sobre la instancia actual.

El esquema es SYSIBMADM.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa ENV_INST_INFO
- Privilegio CONTROL en la vista administrativa ENV_INST_INFO
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE para la función de tabla ENV_GET_INST_INFO
- Autorización DATAACCESS

Ejemplo

Este ejemplo solicita información sobre la instancia actual.

```
SELECT * FROM SYSIBMADM.ENV_INST_INFO
```

El siguiente ejemplo muestra los datos de salida de la consulta.

```
INST_NAME          IS_INST_PARTITIONABLE NUM_DBPARTITIONS INST_PTR_SIZE ...
-----
DB2                0                      1                32 ...
```

1 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```
... RELEASE_NUM    SERVICE_LEVEL        BLD_LEVEL          PTF                FIXPACK_NUM
... -----
... 01010107       DB2 v9.1.0.115      n051106           .....
```

Información devuelta

Tabla 87. Información devuelta por la vista administrativa ENV_INST_INFO

| Nombre de columna | Tipo de datos | Descripción |
|-----------------------|---------------|---|
| INST_NAME | VARCHAR(128) | Nombre de la instancia actual. |
| IS_INST_PARTITIONABLE | SMALLINT | Indica si la instancia actual es o no una instancia de servidor de base de datos particionable. Los valores de retorno posibles son 0 (si la instancia no es una instancia de servidor de base de datos particionable) y 1 (si la instancia es una instancia de servidor de base de datos particionable). |
| NUM_DBPARTITIONS | INTEGER | Número de particiones de base de datos. Si no es un entorno de bases de datos particionadas, la función devuelve el valor 1. |
| INST_PTR_SIZE | INTEGER | Tamaño en bits de la instancia actual (32 o 64). |
| RELEASE_NUM | VARCHAR(128) | Número de release interno, tal como es devuelto por el mandato db2level; por ejemplo, 03030106. |
| SERVICE_LEVEL | VARCHAR(128) | Nivel de servicio, tal como es devuelto por el mandato db2level; por ejemplo, DB2 v8.1.1.80. |
| BLD_LEVEL | VARCHAR(128) | Nivel de creación, tal como es devuelto por el mandato db2level; por ejemplo, n041021. |
| PTF | VARCHAR(128) | Identificador de un arreglo temporal de programa (PTF), tal como es devuelto por el mandato db2level; por ejemplo, U498350. |
| FIXPACK_NUM | INTEGER | Número de Fix Pak, tal como lo devuelve el mandato db2level; por ejemplo, 9. |

Vista administrativa ENV_PROD_INFO - Recuperar información sobre productos DB2 instalados

La vista administrativa ENV_PROD_INFO devuelve información acerca de los productos DB2 instalados.

El esquema es SYSIBMADM.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT sobre la vista administrativa ENV_PROD_INFO
- Privilegio CONTROL sobre la vista administrativa ENV_PROD_INFO
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla ENV_GET_PROD_INFO_V95
- Autorización DATAACCESS

Ejemplo

Solicitar la información sobre los productos DB2 instalados.

```
SELECT * FROM SYSIBMADM.ENV_PROD_INFO
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
INSTALLED_PROD  INSTALLED_PROD_FULLNAME  ...
-----
ESE              DB2_ENTERPRISE_SERVER_EDITION  ...
WSE              DB2_WORKGROUP_SERVER_EDITION  ...
EXP              DB2_EXPRESS_EDITION            ...
```

Datos de salida de esta consulta (continuación).

```
... LICENSE_INSTALLED  PROD_RELEASE  LICENSE_TYPE
... -----
... Y                  9.5          AUTHORIZED_USER_OPTION
... N                  9.5          LICENSE_NOT_REGISTERED
... Y                  9.5          RESTRICTED
```

Metadatos de la vista administrativa ENV_PROD_INFO

Tabla 88. Metadatos de la vista administrativa ENV_PROD_INFO

| Nombre de columna | Tipo de datos | Descripción |
|-------------------------|---------------|--|
| INSTALLED_PROD | VARCHAR(26) | Identificadores de los productos DB2 instalados en el sistema. |
| INSTALLED_PROD_FULLNAME | VARCHAR(100) | Nombre completo de los productos DB2 instalados. Los valores de columnas se mostrarán en inglés y en mayúsculas. Las palabras se separan mediante el signo de subrayado. |
| LICENSE_INSTALLED | CHAR(1) | Indica si existe licencia del producto. Si el valor es N, no existe licencia del producto. Si el valor Y, existe licencia del producto. |
| PROD_RELEASE | VARCHAR(26) | Número del release del producto. |

Tabla 88. Metadatos de la vista administrativa ENV_PROD_INFO (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| LICENSE_TYPE | VARCHAR(50) | Nombre del tipo de licencia instalada para el producto. Los valores de retorno posibles son: <ul style="list-style-type: none"> • 12_MONTHS_LICENSE_AND_SUBSCRIPTION • AUTHORIZED_USER • AUTHORIZED_USER_OPTION • CLIENT_DEVICE • CPU • CPU_OPTION • HOST_SERVER_AND_MSU • LICENSE_NOT_REGISTERED • MANAGED_PROCESSOR • N/A • RESTRICTED • TRIAL • UNWARRANTED • USER |

Vista administrativa ENV_SYS_INFO – Recuperar información sobre el sistema

La vista administrativa ENV_SYS_INFO devuelve información sobre el sistema.

El esquema es SYSIBMADM.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa ENV_SYS_INFO
- Privilegio CONTROL en la vista administrativa ENV_SYS_INFO
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE para la función de tabla ENV_GET_SYS_INFO
- Autorización DATAACCESS

Ejemplo

Este ejemplo solicita información sobre el sistema.

```
SELECT * from SYSIBMADM.ENV_SYS_INFO
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
OS_NAME      OS_VERSION  OS_RELEASE  HOST_NAME
-----
WIN32_NT    5.1         Service Pack 1  D570
```

1 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```

... TOTAL_CPUS CONFIGURED_CPUS TOTAL_MEMORY
... -----
...          1          2          1527

```

Información devuelta

Tabla 89. Información devuelta por la vista administrativa ENV_SYS_INFO

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| OS_NAME | VARCHAR(256) | Nombre del sistema operativo. |
| OS_VERSION | VARCHAR(256) | Número de versión del sistema operativo. |
| OS_RELEASE | VARCHAR(256) | Número de release del sistema operativo. |
| HOST_NAME | VARCHAR(256) | Nombre del sistema. |
| TOTAL_CPUS | INTEGER | Número total de las CPU físicas del sistema. |
| CONFIGURED_CPUS | INTEGER | Número de las CPU físicas configuradas del sistema. |
| TOTAL_MEMORY | INTEGER | Cantidad total de memoria del sistema (en megabytes). |

Vista administrativa ENV_SYS_RESOURCES - Devolver información de sistema

La vista administrativa ENV_SYS_RESOURCES indica el sistema operativo, la CPU, la memoria y demás información relacionada con el sistema.

El esquema es SYSIBMADM.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT sobre la vista administrativa ENV_SYS_RESOURCES
- Privilegio CONTROL sobre la vista administrativa ENV_SYS_RESOURCES
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla ENV_GET_SYS_RESOURCES
- Autorización DATAACCESS

Ejemplo

```

SELECT SUBSTR(NAME,1,20) AS NAME, SUBSTR(VALUE,1,10) AS VALUE,
       SUBSTR(DATATYPE,1,10) AS DATATYPE, DBPARTITIONNUM
FROM SYSIBMADM.ENV_SYS_RESOURCES
WHERE SUBSTR(NAME,1,8)='CPU_LOAD' OR NAME='CPU_USAGE_TOTAL'

```

El ejemplo siguiente muestra los datos de salida de la consulta.

```

NAME                VALUE          DATATYPE  DBPARTITIONNUM
-----
CPU_LOAD_SHORT      0.044052      DECIMAL   0
CPU_LOAD_MEDIUM    0.087250      DECIMAL   0

```

| | | | |
|-----------------|----------|----------|---|
| CPU_LOAD_LONG | 0.142059 | DECIMAL | 0 |
| CPU_USAGÉ_TOTAL | 7 | SMALLINT | 0 |

4 registro(s) seleccionado(s).

Metadatos de la vista administrativa ENV_SYS_RESOURCES

Tabla 90. Metadatos de la vista administrativa ENV_SYS_RESOURCES

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| NAME | VARCHAR(128) | Nombre del atributo. Consulte la Tabla 91 para ver los valores posibles. Nota: Es posible que algunos atributos no estén disponibles en el sistema operativo y configuración de hardware en el servidor. |
| VALUE | VARCHAR(1024) | El valor del atributo. |
| DATATYPE | VARCHAR(128) | El tipo de datos del atributos. |
| UNIT | VARCHAR(128) | Unidad utilizada para la columna VALUE, si corresponde. Se devuelve NULL, si no corresponde. |
| DBPARTITIONNUM | SMALLINT | La partición de base de datos desde la que se han recuperado los datos para esta fila. |

Tabla 91. Posibles valores para la columna NAME

| Tipo de información | Nombre | Tipos de datos | Descripción | Plataformas que devuelven esta información | UNIT |
|---------------------|------------------------|----------------|--|--|------|
| Sistema operativo | OS_NAME | VARCHAR(256) | Nombre del software del sistema operativo. | Todos | NULL |
| | HOST_NAME | VARCHAR(256) | Nombre de sistema principal del sistema. | Todos | NULL |
| | OS_VERSION | VARCHAR(256) | Versión del sistema operativo. Por ejemplo, AIX: 4.3 version = 4. | Todos | NULL |
| | OS_RELEASE | VARCHAR(256) | Release del sistema operativo. Por ejemplo, AIX: 4.3 release = 3. | Todos | NULL |
| | MACHINE_IDENTIFICATION | VARCHAR(256) | Identificación del hardware de la máquina. | Todos | NULL |
| | OS_LEVEL | VARCHAR(256) | Nivel de mantenimiento de la versión y release actuales. Por ejemplo, LINUX: 2.4.9, nivel = 9. | Linux | NULL |

Tabla 91. Posibles valores para la columna NAME (continuación)

| Tipo de información | Nombre | Tipos de datos | Descripción | Plataformas que devuelven esta información | UNIT |
|---------------------|----------------------|----------------|--|--|------|
| CPU | CPU_TOTAL | BIGINT | Número total de CPU. | Todos | NULL |
| | CPU_ONLINE | BIGINT | Número de CPU en línea. | Todos | NULL |
| | CPU_CONFIGURED | BIGINT | Número de CPU configuradas. | Todos | NULL |
| | CPU_SPEED | BIGINT | Velocidad de las CPUs. | Todos | MHz |
| | CPU_TIMEBASE | BIGINT | Frecuencia del incremento de registro en función del tiempo. | Linux PowerPC | Hz |
| | CPU_HMT_DEGREE | BIGINT | En sistemas que soportan hardware multihebra (HMT), se refiere al número de procesadores que un procesador físico aparecerá al sistema operativo. En sistemas no-HMT, este valor es 1. En sistemas HMT, "total" indicará el número de CPUs lógicas. Para obtener el número de CPUs físicas, divida el "total" por "threadingDegree". | Todos | NULL |
| | CPU_CORES_PER_SOCKET | BIGINT | Número de CPU básica por socket. En sistemas básicos individuales este valor es 1. | Todos | NULL |
| Memoria física | MEMORY_TOTAL | BIGINT | Tamaño total de la memoria física. | Todos | MB |
| | MEMORY_FREE | BIGINT | Cantidad de memoria física libre. | Todos | MB |
| | MEMORY_SWAP_TOTAL | BIGINT | Cantidad total de espacio de intercambio. | Todos | MB |
| | MEMORY_SWAP_FREE | BIGINT | Cantidad de espacio de intercambio libre. | Todos | MB |

Tabla 91. Posibles valores para la columna NAME (continuación)

| Tipo de información | Nombre | Tipos de datos | Descripción | Plataformas que devuelven esta información | UNIT |
|---------------------|----------------------|----------------|--|---|------------|
| Memoria virtual | VIRTUAL_MEM_TOTAL | BIGINT | Cantidad total de memoria virtual en el sistema. | Todos | MB |
| | VIRTUAL_MEM_RESERVED | BIGINT | Cantidad de memoria virtual reservada. | Todos | MB |
| | VIRTUAL_MEM_FREE | BIGINT | Cantidad de memoria virtual libre. | Todos | MB |
| Carga de CPU | CPU_LOAD_SHORT | DECIMAL | Duración de período más corta. Por ejemplo, ejemplos de carga en los últimos 5 minutos. | Todos menos los sistemas operativos Windows | NULL |
| | CPU_LOAD_MEDIUM | DECIMAL | Duración de período media. Por ejemplo, ejemplos de carga en los últimos 10 minutos. | Todos menos los sistemas operativos Windows | NULL |
| | CPU_LOAD_LONG | DECIMAL | Duración de período más larga. Por ejemplo, ejemplos de carga en los últimos 15 minutos. | Todos menos los sistemas operativos Windows | NULL |
| | CPU_USAGE_TOTAL | DECIMAL | Porcentaje del uso general de la CPU en la máquina. | Todos | Porcentaje |

Capítulo 11. Rutinas del supervisor

Las funciones de tabla de supervisor hacen referencia a funciones de tabla cuyo nombre comienza por "MON", como MON_GET_SERVICE_SUBCLASS. Estas funciones de tabla permiten acceder a los elementos de supervisor disponibles en la infraestructura de supervisión incluida en DB2 Versión 9.7.

Funciones de tabla de supervisor (MON)

Un conjunto de funciones de tabla de supervisor informa sobre diversos aspectos de la carga de trabajo del sistema general:

- MON_GET_CONNECTION y MON_GET_CONNECTION_DETAILS
- MON_GET_SERVICE_SUBCLASS y MON_GET_SERVICE_SUBCLASS_DETAILS
- MON_GET_UNIT_OF_WORK y MON_GET_UNIT_OF_WORK_DETAILS
- MON_GET_WORKLOAD y MON_GET_WORKLOAD_DETAILS

Estas funciones de tabla tienen dos versiones, una de las cuales tiene el sufijo "_DETAILS". La versión sin el sufijo "_DETAILS" proporciona una interfaz SQL relacional que devuelve los datos utilizados con más frecuencia. La versión con el sufijo "_DETAILS" proporciona acceso basado en XML a los datos de supervisor y devuelve un conjunto de datos más completo.

Todas las funciones de tabla incluyen un conjunto común de elementos de supervisor. Estos elementos ofrecen información sobre un conjunto diverso de indicadores de rendimiento del sistema que pueden influir en el tiempo de respuesta de la aplicación. También puede obtener datos de supervisor correspondientes a un subconjunto de la carga de trabajo que le interesa.

Otro conjunto de funciones de tabla devuelve datos para un tipo de objeto de datos determinado.

- MON_GET_BUFFERPOOL
- MON_GET_CONTAINER
- MON_GET_INDEX
- MON_GET_TABLE
- MON_GET_TABLESPACE
- MON_GET_PKG_CACHE_STMT

Utilice estas funciones de tabla para investigar las anomalías de rendimiento asociadas con un objeto de datos concreto.

Otras funciones de tabla resultan útiles para examinar los detalles de actividades y sentencias individuales:

- MON_GET_ACTIVITY_DETAILS devuelve detalles sobre cada actividad que se esté ejecutando en el sistema; estos detalles incluyen información de actividad general (como el texto de sentencia) y un conjunto de métricas.

Además, esta función de tabla desempeña un rol de supervisión del progreso:

- MON_GET_EXTENT_MOVEMENT_STATUS devuelve el estado de la operación de traslado de extensiones.

Características de las funciones de tabla del supervisor (MON)

- Las métricas devueltas por las funciones de tabla de supervisión nunca se restauran. Comienzan por 0 cuando la base de datos se activa y van aumentando hasta que la base de datos se desactiva.
- Con la mayoría de las funciones de tabla, puede optar por recibir datos de un único objeto (por ejemplo, la clase de servicio "A") o de todos los objetos.
- Al igual que con la mayoría de las funciones de tabla, al utilizar estas funciones de tabla en entornos de bases de datos particionadas puede optar por recibir datos de una única partición o de todas ellas. Si decide recibir datos de todas las particiones, las funciones de tabla devuelven una fila para cada partición. Puede sumar los valores de las distintas particiones para obtener el valor de un elemento de supervisor en varias particiones.

Rutinas de supervisor de sucesos (EVMON)

La versión 9.7 de DB2 también ofrece dos nuevas rutinas cuya finalidad es algo distinta de las de las demás funciones de tabla "MON". Estas rutinas extraen y dan formato a los datos de los supervisores de sucesos que graban sucesos en una tabla de sucesos sin formato. Los tipos de supervisor de sucesos LOCKING y UNIT OF WORK utilizan tablas de sucesos sin formato. Los nombres de las rutinas son los siguientes:

- Función de tabla EVMON_FORMAT_UE_TO_XML
- Procedimiento EVMON_FORMAT_UE_TO_TABLES

Estas rutinas permiten acceder a los datos de los supervisores de sucesos, ya sea a través de un documento XML mediante el uso de la función de tabla EVMON_FORMAT_UE_TO_XML, o a través de tablas relacionales mediante el uso del procedimiento EVMON_FORMAT_UE_TO_TABLES.

Procedimiento EVMON_FORMAT_UE_TO_TABLES - Trasladar un documento XML a tablas relacionales

El procedimiento EVMON_FORMAT_UE_TO_TABLES recupera los datos almacenados en una tabla de sucesos sin formato utilizando la función de tabla EVMON_FORMAT_UE_TO_XML y a continuación emplea la tecnología de descomposición para trasladar el documento XML a un conjunto de tablas relacionales.

Sintaxis

```
▶▶—EVMON_FORMAT_UE_TO_TABLES—(—tipo_supsucesos—,—esquemaxsr—,——————▶  
▶—nombreobjetoxsr—,—archivoesquemaxml—,—esquematabla—,——————▶  
▶—nombre_espaciotablas—,—opciones—,—recuento_conf—,—selección completa—)————▶▶
```

El esquema es SYSPROC.

Parámetros de la función de tabla

tipo_supsucesos

Argumento de entrada de tipo VARCHAR(128) que representa el tipo de los datos almacenados en la tabla de sucesos sin formato. Los valores posibles son los siguientes:

LOCKING

Los datos almacenados en la tabla de sucesos sin formato proceden de un supervisor de sucesos de bloqueo.

UOW Los datos almacenados en la tabla de sucesos sin formato proceden de un supervisor de sucesos de unidad de trabajo.

esquemaxsr

Entrada del tipo VARCHAR (128) que especifica la primera parte del nombre de objeto XSR relacional. La segunda parte se deriva del parámetro *nombreobjetoXsr*. El nombre de objeto XSR completo se define como *esquemaxsr.nombreobjetoXsr*. Si este valor es NULL, se utiliza el ID de autorización del usuario de la sesión actual.

nombreobjetoXsr

Argumento de entrada de tipo VARCHAR (128) que especifica la segunda parte del nombre de objeto XSR. La primera parte del nombre de objeto XSR se deriva del parámetro *esquemaxsr*. El nombre de objeto XSR completo se define como *esquemaxsr.nombreobjetoXsr* y es exclusivo entre todos los objetos del XSR.

Si este valor es NULL, el *nombreobjetoXsr* se deriva de la forma siguiente: `EVMON_<tipo_supsucesos>_SCHEMA_<nivel de release de SQL>`. Por ejemplo, un supervisor de sucesos de bloqueo en DB2 Versión 9.7 tendría un *nombreXsr* derivado de `EVMON_LOCKING_SCHEMA_SQL09070`. El objeto XSR hace referencia al archivo de esquema XML que se almacena en el depósito de esquema XSR y se ha habilitado para la descomposición, que se refiere al proceso de crear una dependencia entre la anotación y las tablas a las que se hace referencia en el esquema XML anotado.

archivoesquemaxml

Argumento de entrada de tipo VARCHAR (1024) que es una vía de acceso totalmente calificada para el documento de esquema XML totalmente anotado del disco. Anotación en este caso hace referencia a las anotaciones de descomposición de DB2 que correlacionan los atributos y elementos XML con las tablas relacionales base y sus columnas.

Si este valor es NULL, el procedimiento utilizará el esquema XML especificado:

LOCKING

`sqllib/misc/DB2EvmonLocking.xsd`

UOW `sqllib/misc/DB2EvmonUOW.xsd`

esquematabla

Argumento de entrada de tipo VARCHAR (128) que representa el nombre de esquema de SQL donde se crean las tablas relacionales del supervisor de sucesos. Si este valor es NULL, se utiliza el ID de autorización del usuario de la sesión actual. El esquema de SQL bajo el que se crearán las tablas se determina de la forma siguiente:

- Si `<db2-xdb:SQLSchema>` está especificado, utilice este esquema;
- Si `<db2-xdb:defaultSchema>` está especificado, utilice este esquema;
- Si ninguno de estos esquemas está especificado, utilice el valor del argumento de entrada *sqlschema*.

Nota: Cuando se registra un esquema XML para la descomposición, el depósito de esquema XSR crea una dependencia entre cada tabla a la que se hace referencia en el esquema y el objeto XSR que corresponde a este esquema. Esto significa que el nombre de objeto XSR está vinculado con un conjunto exclusivo de tablas relacionales en la base de datos. Si hace referencia a un

objeto XSR existente, sus datos siempre se descomponen e insertan en las tablas a las que estaba vinculado el objeto XSR.

nombre_espaciotablas

Argumento de entrada de tipo VARCHAR (128) que indica el espacio de tablas en el que se crean las tablas relacionales. El valor por omisión de este parámetro es NULL. Tenga en cuenta que el nombre del espacio de tablas especificado en la sentencia CREATE TABLE dentro del archivo de esquemas XML tiene prioridad ante este parámetro de entrada.

opciones

Argumento de entrada de tipo VARCHAR(1024) que representa la lista de opciones de palabras clave soportadas por esta función de tabla. Cada opción debe estar delimitada mediante un punto y coma (;). Los valores posibles son:

RECREATE_FORCE

Indica que se deberían descartar y recrear las tablas relacionales antes de la descomposición.

RECREATE_ONERROR

Indica que se deberían descartar y recrear las tablas relacionales en las situaciones siguientes:

1. Si el objeto XSR no está registrado, pero la tabla existe.
2. En el primer intento fallido de descomposición. Las anomalías posteriores se devuelven y no se intenta volver a construir las tablas.

Si se produce un error, por ejemplo, un error de espacio de tablas lleno o un error de autorización, el procedimiento no filtrará el SQLCODE devuelto por el procedimiento de descomposición. El procedimiento trata todos los SQLCODES negativos de la misma forma e intenta recrear las tablas.

recuento_conf

Argumento de entrada de tipo INTEGER. Los valores posibles son los siguientes:

- 1 Confirmación tras cada 100 documentos descompuestos satisfactoriamente. Es el valor por omisión.
- 0 Nunca se realiza la confirmación.
- n* Confirmación tras cada *n* documentos descompuestos satisfactoriamente.

selección completa

Argumento de entrada de tipo CLOB(2M) que representa la sentencia fullselect de una tabla de sucesos sin formato. La sentencia fullselect es una consulta que se ajusta a las normas de la sentencia SELECT. La consulta debe seguir las normas siguientes:

- La consulta debe utilizar la cláusula "*" o especificar todas las columnas de la tabla de sucesos sin formato. De lo contrario, se devuelve un error. Las columnas deben especificarse en el mismo orden en que los devuelve la sentencia DESCRIBE de la tabla de sucesos sin formato.
- La consulta debe realizar una selección únicamente en la tabla de sucesos sin formato.
- La cláusula WHERE puede utilizar cualquiera de las columnas no LOB de la tabla de sucesos sin formato para filtrar los sucesos.

Autorización

Privilegio EXECUTE sobre el procedimiento almacenado EVMON_FORMAT_UE_TO_TABLES.

Privilegio SELECT sobre la tabla de sucesos sin formato, si no la ha creado.

Privilegio CREATE para crear las tablas relacionales en el esquema de SQL especificado.

Privilegio INSERT para insertar en las tablas relacionales, si no las ha creado.

Todos los privilegios que requiere el procedimiento XDB_DECOMP_XMP_FROM_QUERY.

Ejemplos

Ejemplo 1: Uso de los parámetros por omisión.

Un usuario llamado Paul invoca el procedimiento utilizando los parámetros por omisión y solicita que todos los sucesos que forman parte de la clase de servicio STUDENTS se inserten en las tablas relacionales.

```
EVMON_FORMAT_UE_TO_TABLES (  
  'UOW', NULL, NULL, NULL, NULL, NULL, NULL, -1,  
  'SELECT * FROM UOWUE  
    WHERE service_subclass_name = 'STUDENTS'  
    ORDER BY event_id, event_timestamp')
```

Los resultados de la llamada son los siguientes:

1. El procedimiento analiza el archivo DB2EvmonUOW.xsd, el archivo de esquema XML por omisión, para identificar el conjunto de tablas relacionales por crear.
2. Las tablas relacionales se crean bajo el esquema de SQL Paul.
3. El esquema XML se registra con un nombre de objeto XSR PAUL.EVMON_UOW_SCHEMA_SQL09070
4. El objeto XSR está habilitado para la descomposición.
5. Los datos están comprimidos y se han insertado en las tablas bajo el esquema de SQL Paul.

Ejemplo 2: Intentar usar tablas en un esquema distinto.

Continuando con el ejemplo anterior, un usuario llamado Dave invoca el procedimiento almacenado estableciendo el parámetro *esquematabla* en Paul.

```
EVMON_FORMAT_UE_TO_TABLES (  
  'UOW', NULL, NULL, NULL, 'Paul', NULL, NULL, -1,  
  'SELECT * FROM UOWTBLE  
    ORDER BY event_timestamp')
```

Los resultados de la llamada son los siguientes:

1. El procedimiento analiza el archivo DB2EvmonUOW.xsd, el archivo de esquema XML por omisión, para identificar el conjunto de tablas relacionales por crear.
2. El procedimiento intenta crear las tablas bajo el esquema Paul. Sin embargo, se devuelve un error ya que las tablas relacionales existen actualmente bajo el esquema de SQL PAUL. Las tablas existentes anteriormente no pueden utilizarse cuando un nuevo objeto XSR se está registrando.

Ejemplo 3: Intentar usar tablas en un esquema distinto.

Continuando con el ejemplo anterior, un usuario llamado Greg invoca el procedimiento almacenado estableciendo el parámetro de entrada *esquemaxsr* en Paul.

```
EVMON_FORMAT_UE_TO_TABLES (  
  'UOW', 'Paul', NULL, NULL, NULL, NULL, NULL, -1,  
  'SELECT * FROM UOWTBL  
  ORDER BY event_timestamp')
```

Los resultados de la llamada son los siguientes:

1. El objeto XSR Paul.EVMON_UOW_SCHEMA_SQL09070, que existe, está habilitado para la descomposición.
2. Si Greg tiene privilegios INSERT en las tablas, los datos se descomponen y se insertan en las tablas relacionales bajo el esquema SQL Paul. El objeto XSR existente Paul.EVMON_UOW_SCHEMA_SQL09070 se utiliza de modo que el esquema de SQL para las tablas relacionales se obtenga del objeto XSR, en vez de proporcionarlo como parámetro de entrada al procedimiento.

Ejemplo 4: Utilizar la opción RECREATE_FORCE.

Continuando con el ejemplo anterior, Paul desea volver a crear las tablas, pero en el espacio de tablas MYSPACE. Paul llama al procedimiento con la opción RECREATE_FORCE y el parámetro *nombre_espaciotablas*.

```
EVMON_FORMAT_UE_TO_TABLES (  
  'UOW', NULL, NULL, NULL, NULL, 'MYSPACE', 'RECREATE_FORCE', -1,  
  'SELECT * FROM UOWTBL  
  ORDER BY event_timestamp')
```

Los resultados de la llamada son los siguientes:

1. El objeto XSR Paul.EVMON_UOW_SCHEMA_SQL09070, que existe, está habilitado para la descomposición.
2. La opción RECREATE_FORCE está establecida.
3. El archivo de esquema XML se recupera del depósito de esquema y se analiza para identificar el conjunto de archivos relacionales.
4. Las tablas actuales se descartan y se vuelven a crear en el espacio de tablas MYSPACE.
5. Los datos se descomponen y se insertan en las tablas nuevas.

Ejemplo 5: Recoger elementos nuevos en una actualización de fixpack.

Se ha añadido un elemento XML nuevo llamado “db2EventNew” al archivo de esquema XML del supervisor de sucesos de bloqueo en el último fixpack. Paul desea recoger el nuevo elemento para utilizarlo en la descomposición de un archivo XML. Para ello, sigue los pasos siguientes:

1. Mediante un editor, abre el archivo del supervisor de sucesos de bloqueo DB2EvmonLocking.xml.
2. Busca el elemento <db2-mon:createStmt> que contiene la sentencia CREATE TABLE para la tabla LOCK_EVENT.
3. Paul crea una columna nueva en la tabla, como se muestra en el ejemplo siguiente, tomando especial nota del valor del conjunto de filas:

```
<xs:annotation>  
  <xs:appinfo>  
    <db2-xdb:table>
```



```

<db2-xdb:name>LOCK_CONTROL</db2-xdb:name>
  <db2-mon:createStmt>
    CREATE TABLE LOCK_EVENT (EVENT_ID INT,
                              EVENT_TIMESTAMP TIMESTAMP,
                              EVENT_TYPE VARCHAR(128),
                              ?
                              EVENT_NEW VARCHAR(128)
                              )
  </db2-mon:createStmt>
  <db2-xdb:rowSet>lock_event_rowset</db2-xdb:rowSet>
  ?
</db2-xdb:table>
</xs:appinfo>
</xs:annotation>

```

4. En el archivo de esquema XML del supervisor de sucesos, localiza el nuevo elemento XML y lo anota para descomposición, añadiendo el conjunto de filas del que debería formar parte el elemento nuevo, así como el nombre de columna.

```

<xs:complexType name="db2lockevent">
  <xs:annotation>
    <xs:documentation>
      Este esquema describe la estructura de cada suceso de bloqueo capturado por el
      supervisor de sucesos. El esquema está formado por los elementos siguientes:
    </xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="db2EventNew"
                type="xs:string"
                minOccurs="0"
                maxOccurs="1"
                db2-xdb:rowRowSet="lock_event_rowset"
                db2-xdb:column="EVENT_NEW" >
      <xs:annotation>
        <xs:documentation>
          Event New es una columna agregada en el FIXPACK X.
        </xs:documentation>
      </xs:annotation>
    </xs:element>
  </xs:sequence>
  ?

```

5. A continuación, Paul descarta el objeto XSR creado en el release original:


```
DROP XSROBJECT EVMON_LOCKING_SCHEMA_SQL09070
```
6. Por último, llama al procedimiento con la opción RECREATE_ONERROR.

```

EVMON_FORMAT_UE_TO_TABLES (
  'LOCKING', NULL, NULL, NULL, NULL, NULL, 'RECREATE_ONERROR', -1,
  'SELECT * FROM LOCK
   ORDER BY event_timestamp')

```

Los resultados de la llamada son los siguientes:

- a. El objeto XSR no existe, por lo que el archivo de esquema DB2EvmonLocking.xsd por omisión se analiza para identificar el conjunto de tablas relacionales.
- b. Puesto que la opción RECREATE_ONERROR se ha especificado, las tablas existentes se descartan y vuelven a crear.

Ejemplo 6: Recoger elementos nuevos en una actualización de release.

Paul está actualizando a un nuevo release de DB2 y desea recoger los cambios nuevos en el archivo de esquema XML del supervisor de sucesos. Paul llama al procedimiento con la opción RECREATE_ONERROR.

```

EVMON_FORMAT_UE_TO_TABLES (
  'LOCKING', NULL, NULL, NULL, NULL, NULL, 'RECREATE_ONERROR', -1,
  'SELECT * FROM LOCK
  ORDER BY event_timestamp')

```

Los resultados de la llamada son los siguientes:

1. El objeto XSR Paul.EVMON_LOCKING_SCHEMA_SQL1000 no existe.
2. Puesto que la opción RECREATE_ONERROR se ha especificado, las tablas se descartan y vuelven a crear.

Ejemplo 7: Usar las tablas relacionales anteriores en una actualización de release.

Greg ha actualizado a un nuevo release de DB2 y no desea recoger los cambios nuevos en el archivo de esquema XML del supervisor de sucesos. Greg llama al procedimiento con el valor *xsobjectname* del release anterior.

```

EVMON_FORMAT_UE_TO_TABLES (
  'LOCKING', NULL, 'EVMON_LOCKING_SCHEMA_SQL09070', NULL, NULL, NULL, NULL, -1,
  'SELECT * FROM LOCK
  ORDER BY event_timestamp')

```

Notas de uso

Creación de tablas

Para que se produzca la descomposición, debe existir un conjunto de tablas relacionales. El procedimiento EVMON_FORMAT_UE_TO_TABLES crea las tablas relacionales automáticamente, de la siguiente manera:

- El procedimiento analiza el archivo de esquema XML del supervisor de sucesos para buscar los elementos <db2-mon:createStmt>. Cada elemento contiene una sentencia CREATE TABLE completa.
- El procedimiento extrae y ejecuta las sentencias CREATE TABLE.

<db2-mon:createStmt> es el elemento hijo del elemento <db2-xdb:table> existente. Solamente el procedimiento EVMON_FORMAT_UE_TO_TABLES reconoce y utiliza este elemento. Los demás procedimientos que analizan el archivo de esquema XML, como los objetos XSR, omiten este elemento.

No califique el nombre de tabla de <db2-mon:createStmt>.

Actualizaciones de fixpack

Los fixpacks de DB2 pueden contener una versión actualizada del archivo de esquema XML del supervisor de sucesos, que puede incluir nuevos elementos XML para el esquema. No obstante, las sentencias de creación bajo los elementos <db2-mon:createStmt> no incluirán columnas nuevas. Asimismo, los atributos o elementos XML nuevos no estarán anotados. Por lo tanto, puede utilizar las tablas relacionales por omisión creadas en un nivel de fixpack específico sin efectos negativos.

Actualizaciones de release

Es probable que las columnas nuevas se añadan a las tablas relacionales del supervisor de sucesos en releases futuros de DB2. Los atributos o elementos XML nuevos contendrán anotaciones que los correlacionarán con las columnas nuevas. Si desea utilizar las tablas antiguas, llame al procedimiento utilizando el nombre

de objeto XSR del release anterior. De lo contrario, llame al procedimiento utilizando la opción RECREATE_ONERROR, para descartar las tablas antiguas y crear las nuevas.

Archivos de esquema XML de release a release

Los archivos de esquema XML por omisión que se suministran con cada supervisor de sucesos reflejan siempre el esquema XML del release actual.

Información devuelta

No hay salida del procedimiento, salvo SQLCA. SQLCA indica el estado de finalización. Los SQLCODES posibles son:

- 0 Todos los sucesos se insertaron satisfactoriamente en las tablas relacionales.
- 16278 No se insertaron uno o varios sucesos en las tablas relacionales. Los símbolos de SQLCA contienen el número total de documentos intentados y el número total de documentos en los que la descomposición resultó satisfactoria.

También se crea un archivo de diagnóstico; y el nombre y ubicación de ese archivo de diagnóstico están almacenados en los archivos de anotaciones cronológicas db2diag, ubicados en la vía de acceso de diagnóstico de DB2.

sqlcode negativo

Se ha producido un error; si se investiga el mensaje SQLCODE, pueden obtenerse detalles adicionales sobre la anomalía. Para ver mensajes de diagnóstico adicionales, consulte los archivos de anotaciones cronológicas db2diag ubicados en la vía de acceso de diagnóstico de DB2.

Función de tabla EVMON_FORMAT_UE_TO_XML - Convertir sucesos sin formatear a XML

La función de tabla EVMON_FORMAT_UE_TO_XML extrae sucesos binarios de una tabla de sucesos sin formato, les da formato y los incluye en un documento XML.

Sintaxis

```
►►—EVMON_FORMAT_UE_TO_XML—(—opciones—, —————)—————►  
►—FOR EACH ROW OF—(—sentencia-selecc-completa—)—)—————►►
```

El esquema es SYSPROC.

Parámetros de la función de tabla

opciones

Argumento de entrada de tipo VARCHAR(1024) que representa la lista de opciones de palabras clave soportadas por esta función de tabla.

LOG_TO_FILE

Indica que la función de tabla grabará el documento XML como archivo en el caso de que dicho documento supere los 100 MB. El tamaño máximo de cada documento devuelto por fila mediante esta función de tabla es de 100 MB. El archivo se grabará en el archivo <id_documento_xml>.xml, siendo <id_documento_xml> el ID exclusivo

generado para cada documento. El archivo de salida se graba en el directorio de vía de acceso de diagnósticos de DB2.

LOG_PARTIAL_EVENTS

Indica que la función de tabla grabará todos los sucesos parciales (incompletos) en un archivo. Se inserta un mensaje de diagnóstico que indica el archivo concreto en los archivos de anotaciones cronológicas de db2diag.

NULL No se selecciona ninguna opción.

sentencia-selecc-completa

La sentencia fullselect es una consulta que se ajusta a las normas de la sentencia SELECT. La consulta debe seguir las normas siguientes:

- La consulta debe utilizar la cláusula "*" o especificar todas las columnas de la tabla de sucesos sin formato. De lo contrario, se devuelve un error. Las columnas deben especificarse en el mismo orden en que los devuelve la sentencia DESCRIBE de la tabla de sucesos sin formato.
- La consulta debe realizar una selección únicamente en la tabla de sucesos sin formato.
- La cláusula WHERE puede utilizar cualquiera de las columnas no LOB de la tabla de sucesos sin formato para filtrar los sucesos.
- La sentencia SELECT debe especificarse en la palabra clave FOR EACH ROWS OF entre corchetes.

Autorización

Privilegio EXECUTE sobre la función EVMON_FORMAT_UE_TO_XML.

Privilegio SELECT sobre la tabla de sucesos sin formato.

Ejemplos

Ejemplo 1: Consultar todos los sucesos de la tabla de sucesos sin formato "MYLOCKS".

```
SELECT evmon.* FROM TABLE (  
  EVMON_FORMAT_UE_TO_XML (  
    NULL ,  
    FOR EACH ROW OF (  
      select * from MYLOCKS  
      order by EVENT_TIMESTAMP )))  
AS evmon;
```

Ejemplo 2: Consultar todos los sucesos de tipo "LOCKWAIT" que se hayan producido en las últimas 5 horas en la tabla de sucesos sin formato "LOCK".

```
SELECT evmon.* FROM TABLE (  
  EVMON_FORMAT_UE_TO_XML (  
    NULL ,  
    FOR EACH ROW OF (  
      select * from LOCK order by EVENT_TIMESTAMP  
      where EVENT_TYPE = 'LOCKWAIT'  
      and EVENT_TIMESTAMP >= CURRENT_TIMESTAMP - 5 hours )))  
AS evmon;
```

Ejemplo 3: Obtener todos los sucesos que pertenecen a la carga de trabajo "PAYROLL" que se hayan producido en las últimas 32 horas en la tabla de sucesos sin formato "UOW". Grabar el resultado en un archivo si algún documento tiene más de 100 MB.

```

SELECT evmon.* FROM TABLE (
  EVMON_FORMAT_UE_TO_XML(
    'LOG TO FILE',
    FOR EACH ROW OF (
      select * from UOW order by EVENT_TIMESTAMP
      where WORKLOAD_NAME = 'PAYROLL'
      and EVENT_TIMESTAMP = CURRENT_TIMESTAMP - 32 hours )))
AS evmon;

```

Ejemplo 4: Consultar todos los sucesos de unidad de trabajo de la tabla "UOWEVMON" y utilizar la función de tabla XMLTABLE para presentar el ID de la unidad de trabajo (UOW), las horas de inicio y fin de la UOW y el ID de usuario de la persona que ha emitido la unidad de trabajo.

```

SELECT EVENT.UOW_ID, EVENT.APPLICATION_ID, EVENT.SESSION_AUTHID,
  EVENT.START_TIME, EVENT.STOP_TIME
FROM TABLE(
  EVMON_FORMAT_UE_TO_XML(
    'LOG TO FILE',
    FOR EACH ROW OF (
      select * from UOWEVMON )))
AS UEXML,
XMLTABLE (XMLNAMESPACES( DEFAULT 'http://www.ibm.com/xmlns/prod/db2/mon'),
'$uowevent/db2_uow_event'
  PASSING XMLPARSE( DOCUMENT UEXML.XMLREPORT ) as "uowevent"
  COLUMNS UOW_ID INTEGER PATH 'uow_id',
  MEMBER SMALLINT PATH '@member',
  APPLICATION_ID VARCHAR(128) PATH 'application_id',
  SESSION_AUTHID VARCHAR(128) PATH 'session_authid',
  START_TIME TIMESTAMP PATH 'start_time',
  STOP_TIME TIMESTAMP PATH 'stop_time'
)
AS EVENT

```

Notas de uso

La función de tabla EVMON_FORMAT_UE_TO_XML se puede utilizar con los supervisores de sucesos de bloqueo y de unidad de trabajo para extraer datos de una tabla de sucesos sin formato.

En función del tipo de supervisor de sucesos, la función de tabla correlaciona varios registros de la tabla de sucesos sin formato en un único suceso. En tal caso, los registros se colocan en la antememoria hasta que se reciban todos los registros que forman el suceso. Podría ser necesaria más memoria si los registros pasados a la función de tabla no están en el orden en que se crearon e insertaron en la tabla. Si los registros no están ordenados de esta forma, la función de tabla debe colocar en la antememoria los registros de los múltiples sucesos. Para evitar este problema, cualifique el parámetro *sentencia-selecc-completa* con una cláusula ORDER BY que contenga las columnas siguientes: EVENT_ID, EVENT_TIMESTAMP, EVENT_TYPE y MEMBER. El consumo de memoria se reduce porque en todo momento la función de tabla procesa y coloca en la antememoria los registros de un solo suceso.

Información devuelta

Tabla 92. Información devuelta para EVMON_FORMAT_UE_TO_XML.

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|---|
| XMLID | VARCHAR(1024) | Un ID de documento exclusivo. El ID se deriva como se indica a continuación: <cabecera_suceso>_<id_suceso>_<tipo_suceso>_ <indic_hora_fecha_suceso>_<partición> |
| XMLREPORT | BLOB(100M) | Un documento XML que contiene un único suceso completo. Cada documento tiene un tamaño máximo de 100 MB. |

Función de tabla MON_GET_ACTIVITY_DETAILS - Obtener detalles completos sobre la actividad

La función de tabla MON_GET_ACTIVITY_DETAILS devuelve detalles sobre una actividad, incluida información de actividad general (como el texto de sentencia) y un conjunto de métricas para la actividad.

Sintaxis

```
►►—MON_GET_ACTIVITY_DETAILS—(—descriptorcontexto_aplicación—,—uow_id—,—  
►—id_actividad—,—miembro—)—————►►
```

El esquema es SYSPROC.

Parámetros de la función de tabla

descriptorcontexto_aplicación

Es un argumento de entrada de tipo BIGINT que especifica un descriptor de contexto de aplicación válido. Si el argumento es nulo, esta función no devuelve ninguna fila y se devuelve el error SQL0171N.

id_uow

Argumento de entrada de tipo INTEGER que especifica un identificador de unidad de trabajo válido exclusivo dentro de la aplicación. Si el argumento es nulo, esta función no devuelve ninguna fila y se devuelve el error SQL0171N.

id_actividad

Argumento de entrada de tipo INTEGER que especifica un ID de actividad válido exclusivo dentro de la unidad de trabajo. Si el argumento es nulo, esta función no devuelve ninguna fila y se devuelve el error SQL0171N.

miembro

Argumento de entrada de tipo INTEGER que especifica un número de miembro válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique -1 para el miembro de base de datos actual o -2 para todos los miembros de base de datos. Si se especifica el valor nulo, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE sobre la función MON_GET_ACTIVITY_DETAILS.

Ejemplo

Investigar una consulta de ejecución prolongada para determinar si está dedicando su tiempo a ejecutar o a esperar (por ejemplo, bloqueada en los bloqueos o E/S).

1. Utilizar primero la función de tabla WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES para obtener una lista de las actividades y sus horas de inicio.

```
SELECT application_handle,
       activity_id,
       uow_id,
       local_start_time
FROM TABLE(
  WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES(
    cast(NULL as bigint), -1)
) AS T
```

A continuación se muestra un ejemplo de salida de esta consulta.

| APPLICATION_HANDLE | ACTIVITY_ID | UOW_ID | LOCAL_START_TIME |
|--------------------|-------------|--------|----------------------------|
| 7 | 1 | 2 | 2008-06-10-10.06.55.675668 |
| 16 | 1 | 7 | 2008-06-10-10.08.38.613610 |

2 registro(s) seleccionado(s).

2. A continuación, utilizar la función de tabla MON_GET_ACTIVITY_DETAILS para visualizar el porcentaje de tiempo que la actividad ha pasado en espera.

```
SELECT actmetrics.application_handle,
       actmetrics.activity_id,
       actmetrics.uow_id,
       varchar(actmetrics.stmt_text, 50) as stmt_text,
       actmetrics.total_act_time,
       actmetrics.total_act_wait_time,
       CASE WHEN actmetrics.total_act_time > 0
            THEN DEC(
              FLOAT(actmetrics.total_act_wait_time) /
              FLOAT(actmetrics.total_act_time)) * 100, 5, 2)
            ELSE NULL
       END AS PERCENTAGE_WAIT_TIME
FROM TABLE(MON_GET_ACTIVITY_DETAILS(7, 2, 1, -2)) AS ACTDETAILS,
XMLTABLE (XMLNAMESPACES( DEFAULT 'http://www.ibm.com/xmlns/prod/db2/mon'),
 '$actmetrics/db2_activity_details'
 PASSING XMLPARSE(DOCUMENT ACTDETAILS.DETAILS) as "actmetrics"
 COLUMNS "APPLICATION_HANDLE" INTEGER PATH 'application_handle',
          "ACTIVITY_ID" INTEGER PATH 'activity_id',
          "UOW_ID" INTEGER PATH 'uow_id',
          "STMT_TEXT" VARCHAR(1024) PATH 'stmt_text',
          "TOTAL_ACT_TIME" INTEGER PATH 'activity_metrics/total_act_time',
          "TOTAL_ACT_WAIT_TIME" INTEGER PATH 'activity_metrics/total_act_wait_time'
) AS ACTMETRICS;
```

A continuación se muestra un ejemplo de salida de esta consulta.

| APPLICATION_HANDLE | ACTIVITY_ID | UOW_ID | ... |
|--------------------|-------------|--------|-----|
| 7 | 1 | 2 | ... |

1 registro(s) seleccionado(s).

Salida de la consulta (continuación).

```
... STMT_TEXT ...
... ----- ...
... select * from syscat.tables optimize for 1 row ...
```

Salida de la consulta (continuación).

```

... TOTAL_ACT_TIME TOTAL_ACT_WAIT_TIME PERCENTAGE_WAIT_TIME
... -----
...                459                0                0.00

```

Notas de uso

La función `MON_GET_ACTIVITY_DETAILS` ofrece la máxima flexibilidad para dar formato a la salida ya que devuelve la información detallada de una única actividad como un documento XML. La salida XML incluye tanto información descriptiva (por ejemplo, texto de sentencia) como métricas. La salida puede analizarse directamente mediante un analizador XML, o puede convertirse a formato relacional mediante la función `XMLTABLE` como se muestra en el ejemplo.

Las métricas notificadas con esta función (por ejemplo, el uso de la CPU) se avanzan periódicamente a la actividad durante la vida útil de la actividad. Por consiguiente, los valores notificados por esta función de tabla reflejan el estado actual del sistema en el momento del avance más reciente.

Las métricas de actividad se controlan mediante la cláusula `COLLECT ACTIVITY METRICS` en las cargas de trabajo, o mediante el parámetro de configuración de base de datos `mon_act_metrics` en el nivel de la base de datos. Las métricas se recopilan si la conexión que envía la actividad está asociada con una carga de trabajo o una base de datos para la que están habilitadas las métricas de actividad. Si no se recopilan métricas de actividad para una actividad, todas las métricas se notifican como 0.

La función de tabla `MON_GET_ACTIVITY_DETAILS` devuelve una fila de datos por cada miembro en el que existe la actividad. No se realiza una agregación en los miembros para las métricas. Sin embargo, se puede lograr la agregación mediante consultas SQL.

El esquema para el documento XML devuelto en la columna `DETAILS` está disponible en el archivo `sql1ib/misc/DB2MonRoutines.xsd`. Para obtener más información, consulte el archivo `sql1ib/misc/DB2MonCommon.xsd`.

Información devuelta

Tabla 93. Información devuelta para `MON_GET_ACTIVITY_DETAILS`

| Nombre de columna | Tipo de datos | Descripción |
|---------------------------------|-----------------------|--|
| <code>APPLICATION_HANDLE</code> | <code>BIGINT</code> | <code>application_handle</code> - Descriptor de contexto de aplicación |
| <code>UOW_ID</code> | <code>INTEGER</code> | <code>uow_id</code> - ID de unidad de trabajo |
| <code>ACTIVITY_ID</code> | <code>INTEGER</code> | <code>activity_id</code> - ID de actividad |
| <code>MEMBER</code> | <code>SMALLINT</code> | <code>member</code> - Miembro de base de datos |
| <code>DETAILS</code> | <code>BLOB(8M)</code> | Documento XML que contiene los detalles de actividad. Consulte la Tabla 94 en la página 373 para ver una descripción de los elementos de este documento. |

El ejemplo siguiente muestra la estructura del documento XML que se devuelve en la columna `DETAILS`.

```

<db2_activity_details xmlns="http://www.ibm.com/xmlns/prod/db2/mon" release="90700000">
  <member>0</member>
  <application_handle>70</application_handle>
  <activity_id>1</activity_id>
  <activity_state>IDLE</activity_state>
  <activity_type>READ_DML</activity_type>

```



```

<uow_id>1</uow_id>
...
<activity_metrics release="90700000">
  <lock_wait_time>2000</lock_wait_time>
...
</activity_metrics>
</db2_activity_details>

```

Para ver el esquema completo, consulte `sqllib/misc/DB2MonRoutines.xsd`. Este documento emplea las definiciones de tipo no primitivo XML siguientes:

```

<xs:simpleType name = "executable_id_type" >
  <xs:annotation>
    <xs:documentation>
      El ID ejecutable binario
    </xs:documentation>
  </xs:annotation>
  <xs:restriction base="xs:hexBinary">
    <xs:maxLength value = "32" />
  </xs:restriction>
</xs:simpleType>

```

Métricas detalladas devueltas

Tabla 94. Métricas detalladas devueltas para `MON_GET_ACTIVITY_DETAILS`

| Nombre de elemento | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|----------------------------------|-----------------------|--|
| member | xs:nonNegativeInteger | member - Miembro de base de datos |
| client_userid | xs:string(255) | Registro especial CURRENT CLIENT_USERID |
| client_wrkstnname | xs:string(255) | Registro especial CURRENT CLIENT_WRKSTNNAME |
| client_applname | xs:string(255) | Registro especial CURRENT CLIENT_APPLNAME |
| client_acctng | xs:string(255) | Registro especial CURRENT CLIENT_ACCTNG |
| application_handle | xs:nonNegativeInteger | application_handle - Descriptor de contexto de aplicación |
| coord_member | xs:nonNegativeInteger | coord_member - Miembro del coordinador |
| uow_id | xs:nonNegativeInteger | uow_id - ID de unidad de trabajo |
| activity_id | xs:nonNegativeInteger | activity_id - ID de actividad |
| parent_uow_id | xs:nonNegativeInteger | parent_uow_id - ID de unidad de trabajo padre |
| parent_activity_id | xs:nonNegativeInteger | parent_activity_id - ID de actividad padre |
| activity_state | xs:string | activity_state - Estado de actividad |
| activity_type | xs:string | activity_type - Tipo de actividad |
| nesting_level | xs:nonNegativeInteger | nesting_level - Nivel de anidamiento |
| invocation_id | xs:nonNegativeInteger | invocation_id - ID de invocación |
| routine_id | xs:nonNegativeInteger | routine_id - ID de rutina |
| utility_id | xs:nonNegativeInteger | utility_id - ID de programa de utilidad |
| service_class_id | xs:integer | service_class_id - Clase de servicio |
| database_work_action_set_id | xs:nonNegativeInteger | db_work_action_set_id - ID de conjunto de acciones de trabajo de base de datos |
| database_work_class_id | xs:nonNegativeInteger | db_work_class_id - ID de clase de trabajo de base de datos |
| service_class_work_action_set_id | xs:nonNegativeInteger | sc_work_action_set_id - ID de conjunto de acciones de trabajo de clase de servicio |
| service_class_work_class_id | xs:nonNegativeInteger | sc_work_class_id - ID de clase de trabajo de clase de servicio |

Tabla 94. Métricas detalladas devueltas para MON_GET_ACTIVITY_DETAILS (continuación)

| Nombre de elemento | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---|---------------------------|---|
| entry_time | xs:dateTime | entry_time - Hora de entradaLa hora a la que esta actividad ha llegado al sistema. |
| local_start_time | xs:dateTime | local_start_time - Hora de inicio local. |
| last_reference_time | xs:dateTime | last_reference_time - Hora de última referencia. Cada vez que se produce una petición en esta actividad, este campo se actualiza. |
| package_name | xs:string (128) | package_name - Nombre de paquete |
| package_schema | xs:string (128) | package_schema - Esquema de paquete |
| package_version_id | xs:string (128) | package_version_id - Versión del paquete |
| section_number | xs:integer | section_number - Número de sección |
| stmt_pkg_cache_id | xs:nonNegativeInteger | stmt_pkgcache_id - Identificador de antememoria del paquete de la sentencia |
| stmt_text | xs:string | stmt_text - Texto de sentencia de SQL. Si la actividad es SQL dinámico o es SQL estático para el que el texto de la sentencia está disponible, este campo contiene los 1024 primeros caracteres del texto de la sentencia. En caso contrario, contiene una serie vacía. |
| effective_isolation | xs:string | effective_isolation - Aislamiento efectivo |
| effective_lock_timeout | xs:integer | effective_lock_timeout - Tiempo de espera de bloqueo efectivo |
| effective_query_degree | xs:integer | effective_query_degree - Grado de consulta efectivo |
| query_cost_estimate | xs:integer | query_cost_estimate - Estimación del coste de la consulta |
| qp_query_id | xs:nonNegativeInteger | qp_query_id - ID de consulta de Query Patroller |
| num_remaps | xs:nonNegativeInteger | num_remaps - Número de nuevas correlaciones |
| concurrentdbcoordactivities_db_threshold_id | xs:int | concurrentdbcoordactivities_db_threshold_id - ID de umbral de las actividades del coordinador de bases de datos simultáneas |
| concurrentdbcoordactivities_db_threshold_value | xs:long | concurrentdbcoordactivities_db_threshold_value - Actividades del coordinador de bases de datos simultáneas |
| concurrentdbcoordactivities_db_threshold_queued | xs:short (1 = sí, 0 = no) | concurrentdbcoordactivities_db_threshold_queued - Actividades del coordinador de bases de datos simultáneas |
| concurrentdbcoordactivities_db_threshold_violated | xs:short (1 = sí, 0 = no) | concurrentdbcoordactivities_db_threshold_violated - Umbral de actividades del coordinador de bases de datos simultáneas violado |
| concurrentdbcoordactivities_superclass_threshold_id | xs:int | concurrentdbcoordactivities_superclass_threshold_id - Superclase de actividades del coordinador de bases de datos simultáneas |
| concurrentdbcoordactivities_superclass_threshold_value | xs:long | concurrentdbcoordactivities_superclass_threshold_value - Valor de umbral de la superclase de actividades del coordinador de bases de datos simultáneas |
| concurrentdbcoordactivities_superclass_threshold_queued | xs:short (1 = sí, 0 = no) | concurrentdbcoordactivities_superclass_threshold_queued - Umbral de la superclase de actividades del coordinador de bases de datos simultáneas en cola |
| concurrentdbcoordactivities_superclass_threshold_violated | xs:short (1 = sí, 0 = no) | concurrentdbcoordactivities_superclass_threshold_violated - Umbral de la superclase de actividades del coordinador de bases de datos simultáneas violado |

Tabla 94. Métricas detalladas devueltas para MON_GET_ACTIVITY_DETAILS (continuación)

| Nombre de elemento | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--|---------------------------|---|
| concurrentdbcoordactivities_subclass_threshold_id | xs:int | concurrentdbcoordactivities_subclass_threshold_id - ID de umbral de la subclase de actividades del coordinador de bases de datos simultáneas |
| concurrentdbcoordactivities_subclass_threshold_value | xs:long | concurrentdbcoordactivities_subclass_threshold_value - Valor de umbral de la subclase de actividades del coordinador de bases de datos simultáneas |
| concurrentdbcoordactivities_subclass_threshold_queued | xs:short (1 = sí, 0 = no) | concurrentdbcoordactivities_subclass_threshold_queued - Umbral de la subclase de actividades del coordinador de bases de datos simultáneas en cola |
| concurrentdbcoordactivities_subclass_threshold_violated | xs:short (1 = sí, 0 = no) | concurrentdbcoordactivities_subclass_threshold_violated - Umbral de la subclase de actividades del coordinador de bases de datos simultáneas violado |
| concurrentdbcoordactivities_work_action_set_threshold_id | xs:int | concurrentdbcoordactivities_work_action_set_threshold_id - ID de umbral de conjunto de acciones de trabajo de actividades del coordinador de bases de datos simultáneas |
| concurrentdbcoordactivities_work_action_set_threshold_value | xs:long | concurrentdbcoordactivities_work_action_set_threshold_value - Valor de umbral de conjunto de acciones de trabajo de actividades del coordinador de bases de datos simultáneas |
| concurrentdbcoordactivities_work_action_set_threshold_queued | xs:short (1 = sí, 0 = no) | concurrentdbcoordactivities_work_action_set_threshold_queued - Umbral de conjunto de acciones de trabajo de actividades del coordinador de bases de datos simultáneas en cola |
| concurrentdbcoordactivities_work_action_set_threshold_violated | xs:short (1 = sí, 0 = no) | concurrentdbcoordactivities_work_action_set_threshold_violated - Umbral de conjunto de acciones de trabajo de actividades del coordinador de bases de datos simultáneas violado |
| estimatedsqlcost_threshold_id | xs:int | estimatedsqlcost_threshold_id - ID de umbral de coste de SQL estimado |
| estimatedsqlcost_threshold_value | xs:long | estimatedsqlcost_threshold_value - Valor de umbral de coste de SQL estimado |
| estimatedsqlcost_threshold_violated | xs:short (1 = sí, 0 = no) | estimatedsqlcost_threshold_violated - Umbral de coste de SQL estimado violado |
| sqltempespace_threshold_id | xs:int | sqltempespace_threshold_id - ID de umbral de espacio temporal de SQL |
| sqltempespace_threshold_value | xs:long | sqltempespace_threshold_value - Valor de umbral de espacio temporal de SQL |
| sqltempespace_threshold_violated | xs:short (1 = sí, 0 = no) | sqltempespace_threshold_violated - Umbral de espacio temporal de SQL violado |
| sqlrowsreturned_threshold_id | xs:int | sqlrowsreturned_threshold_id - ID de umbral de filas de SQL leídas devueltas |
| sqlrowsreturned_threshold_value | xs:long | sqlrowsreturned_threshold_value - Valor de umbral de filas de SQL leídas devueltas |
| sqlrowsreturned_threshold_violated | xs:short (1 = sí, 0 = no) | sqlrowsreturned_threshold_violated - Umbral de filas de SQL leídas devueltas violado |
| activitytotaltime_threshold_id | xs:int | activitytotaltime_threshold_id - ID de umbral de tiempo total de actividad |
| activitytotaltime_threshold_value | xs:dateTime | activitytotaltime_threshold_value - Valor de umbral de tiempo total de actividad |

Tabla 94. Métricas detalladas devueltas para MON_GET_ACTIVITY_DETAILS (continuación)

| Nombre de elemento | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------------------------|---------------------------|--|
| activitytotaltime_threshold_violated | xs:short (1 = sí, 0 = no) | activitytotaltime_threshold_violated - Umbral de tiempo total de actividad violado |
| cputime_threshold_id | xs:int | cputime_threshold_id - ID de umbral de tiempo de CPU |
| cputime_threshold_value | xs:long | cputime_threshold_value - Valor de umbral de tiempo de CPU |
| cputime_threshold_violated | xs:short (1 = sí, 0 = no) | cputime_threshold_violated - Umbral de tiempo de CPU violado |
| cputimeinsc_threshold_id | xs:int | cputimeinsc_threshold_id - ID de umbral de tiempo en servicio de CPU |
| cputimeinsc_threshold_value | xs:long | cputimeinsc_threshold_value - Valor de umbral de tiempo en servicio de CPU |
| cputimeinsc_threshold_violated | xs:short (1 = sí, 0 = no) | cputimeinsc_threshold_violated - Umbral de tiempo en servicio de CPU violado |
| sqlrowsread_threshold_id | xs:int | sqlrowsread_threshold_ID - ID de umbral de filas de SQL leídas |
| sqlrowsread_threshold_value | xs:long | sqlrowsread_threshold_value - Valor de umbral de filas de SQL leídas |
| sqlrowsread_threshold_violated | xs:short (1 = sí, 0 = no) | sqlrowsread_threshold_violated - Umbral de filas de SQL leídas violado |
| sqlrowsreadinsc_threshold_id | xs:int | sqlrowsreadinsc_threshold_id - ID de umbral de filas de SQL leídas en servicio |
| sqlrowsreadinsc_threshold_value | xs:long | sqlrowsreadinsc_threshold_value - Valor de umbral de filas de SQL leídas en servicio |
| sqlrowsreadinsc_threshold_violated | xs:short (1 = sí, 0 = no) | sqlrowsreadinsc_threshold_violated - Umbral de filas de SQL leídas en servicio violado |
| aggsqltempespace_threshold_id | xs:int | aggsqltempespace_threshold_id - ID de umbral de espacio temporal de AggSQL. |
| aggsqltempespace_threshold_value | xs:long | aggsqltempespace_threshold_value - Valor de umbral de espacio temporal de AggSQL |
| aggsqltempespace_threshold_violated | xs:short (1 = sí, 0 = no) | aggsqltempespace_threshold_violated - Umbral de espacio temporal de AggSQL violado |
| audit_events_total | xs:nonNegativeInteger | audit_events_total - Total de sucesos de auditoría |
| audit_subsystem_wait_time | xs:nonNegativeInteger | audit_subsystem_wait_time - Tiempo de espera del subsistema de auditoría |
| audit_subsystem_waits_total | xs:nonNegativeInteger | audit_subsystem_waits_total - Total de esperas del subsistema de auditoría |
| audit_file_write_wait_time | xs:nonNegativeInteger | audit_file_write_wait_time - Tiempo de espera de grabación del archivo de auditoría |
| audit_file_writes_total | xs:nonNegativeInteger | audit_file_writes_total - Total de archivos de auditoría grabados |
| deadlocks | xs:nonNegativeInteger | deadlocks - Puntos muertos detectados |
| diaglog_writes_total | xs:nonNegativeInteger | diaglog_writes_total - Total de grabaciones de anotaciones cronológicas de diagnóstico |
| diaglog_write_wait_time | xs:nonNegativeInteger | diaglog_write_wait_time - Tiempo de grabación de anotaciones cronológicas de diagnóstico |
| direct_read_time | xs:nonNegativeInteger | direct_read_time - Tiempo de lectura directa |

Tabla 94. Métricas detalladas devueltas para MON_GET_ACTIVITY_DETAILS (continuación)

| Nombre de elemento | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|----------------------------|-----------------------|--|
| direct_write_time | xs:nonNegativeInteger | direct_write_time - Tiempo de grabación directa |
| direct_read_reqs | xs:nonNegativeInteger | direct_read_reqs - Peticiones de lectura directa |
| direct_reads | xs:nonNegativeInteger | direct_reads - Lecturas directas de la base de datos |
| direct_write_reqs | xs:nonNegativeInteger | direct_write_reqs - Peticiones de grabación directa |
| direct_writes | xs:nonNegativeInteger | direct_writes - Grabaciones directas en la base de datos |
| fcm_recv_volume | xs:nonNegativeInteger | fcm_recv_volume - Volumen de recepción de FCM |
| fcm_recv_wait_time | xs:nonNegativeInteger | fcm_recv_wait_time - Tiempo de espera de recepción de FCM |
| fcm_recvs_total | xs:nonNegativeInteger | fcm_recvs_total - Total de recepciones de FCM |
| fcm_message_recv_volume | xs:nonNegativeInteger | fcm_message_recv_volume - Volumen de recepción de mensaje de FCM |
| fcm_message_recvs_total | xs:nonNegativeInteger | fcm_message_recvs_total - Total de recepciones de mensaje de FCM |
| fcm_message_recv_wait_time | xs:nonNegativeInteger | fcm_message_recv_wait_time - Tiempo de espera de recepción de mensaje de FCM |
| fcm_message_send_volume | xs:nonNegativeInteger | fcm_message_send_volume - Volumen de envío de mensaje de FCM |
| fcm_message_send_wait_time | xs:nonNegativeInteger | fcm_message_send_wait_time - Tiempo de espera de envío de mensaje de FCM |
| fcm_message_sends_total | xs:nonNegativeInteger | fcm_message_sends_total - Total de envíos de mensaje de FCM |
| fcm_send_volume | xs:nonNegativeInteger | fcm_send_volume - Volumen de envío de FCM |
| fcm_send_wait_time | xs:nonNegativeInteger | fcm_send_wait_time - Tiempo de espera de envío de FCM |
| fcm_sends_total | xs:nonNegativeInteger | fcm_sends_total - Total de envíos de FCM |
| fcm_tq_recv_wait_time | xs:nonNegativeInteger | fcm_tq_recv_wait_time - Tiempo de espera de recepción de cola de tabla de FCM |
| fcm_tq_send_wait_time | xs:nonNegativeInteger | fcm_tq_send_wait_time - Tiempo de espera de envío de cola de tabla de FCM |
| fcm_tq_recv_volume | xs:nonNegativeInteger | fcm_tq_recv_volume - Volumen de recepción de cola de tabla de FCM |
| fcm_tq_recvs_total | xs:nonNegativeInteger | fcm_tq_recvs_total - Total de recepciones de cola de tabla de FCM |
| fcm_tq_send_volume | xs:nonNegativeInteger | fcm_tq_send_volume - Volumen de envíos de cola de tabla de FCM |
| fcm_tq_sends_total | xs:nonNegativeInteger | fcm_tq_sends_total - Total de envíos de cola de tabla de FCM |
| tq_tot_send_spills | xs:nonNegativeInteger | tq_tot_send_spills - Total de almacenamientos intermedios de cola de tabla desbordados |
| lock_escals | xs:nonNegativeInteger | lock_escals - Número de escalamientos de bloqueo |
| lock_timeouts | xs:nonNegativeInteger | lock_timeouts - Número de tiempos de espera de bloqueo |
| lock_wait_time | xs:nonNegativeInteger | lock_wait_time - Tiempo esperado en bloqueos |
| lock_waits | xs:nonNegativeInteger | lock_waits - Esperas de bloqueo |
| log_buffer_wait_time | xs:nonNegativeInteger | log_buffer_wait_time - Tiempo de espera de almacenamiento intermedio de anotaciones cronológicas |

Tabla 94. Métricas detalladas devueltas para MON_GET_ACTIVITY_DETAILS (continuación)

| Nombre de elemento | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|-----------------------|---|
| log_disk_wait_time | xs:nonNegativeInteger | log_disk_wait_time - Tiempo de espera de disco de anotaciones cronológicas |
| log_disk_waits_total | xs:nonNegativeInteger | log_disk_waits_total - Total de esperas de disco de anotaciones cronológicas |
| pool_data_l_reads | xs:nonNegativeInteger | pool_data_l_reads - Lecturas lógicas de datos de la agrupación de almacenamientos intermedios |
| pool_data_p_reads | xs:nonNegativeInteger | pool_data_p_reads - Lecturas físicas de datos de la agrupación de almacenamientos intermedios |
| pool_data_writes | xs:nonNegativeInteger | pool_data_writes - Grabaciones de datos de la agrupación de almacenamientos intermedios |
| pool_index_l_reads | xs:nonNegativeInteger | pool_index_l_reads - Lecturas lógicas de índices de la agrupación de almacenamientos intermedios |
| pool_index_p_reads | xs:nonNegativeInteger | pool_index_p_reads - Lecturas físicas de índices de la agrupación de almacenamientos intermedios |
| pool_index_writes | xs:nonNegativeInteger | pool_index_writes - Grabaciones de índices de la agrupación de almacenamientos intermedios |
| pool_read_time | xs:nonNegativeInteger | pool_read_time - Tiempo total de lectura física de la agrupación de almacenamientos intermedios |
| pool_temp_data_l_reads | xs:nonNegativeInteger | pool_temp_data_l_reads - Lecturas lógicas de datos temporales de la agrupación de almacenamientos intermedios |
| pool_temp_data_p_reads | xs:nonNegativeInteger | pool_temp_data_p_reads - Lecturas físicas de datos temporales de la agrupación de almacenamientos intermedios |
| pool_temp_index_l_reads | xs:nonNegativeInteger | pool_temp_index_l_reads - Lecturas lógicas de índices temporales de la agrupación de almacenamientos intermedios |
| pool_temp_index_p_reads | xs:nonNegativeInteger | pool_temp_index_p_reads - Lecturas físicas de índices temporales de la agrupación de almacenamientos intermedios |
| pool_temp_xda_l_reads | xs:nonNegativeInteger | pool_temp_xda_l_reads - Lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |
| pool_temp_xda_p_reads | xs:nonNegativeInteger | pool_temp_xda_p_reads - Lecturas físicas de datos XDA temporales de agrupación de almacenamientos intermedios |
| pool_write_time | xs:nonNegativeInteger | pool_write_time - Tiempo total de grabación física de agrupación de almacenamientos intermedios |
| pool_xda_l_reads | xs:nonNegativeInteger | pool_xda_l_reads - Lecturas lógicas de datos de XDA de la agrupación de almacenamientos intermedios |
| pool_xda_p_reads | xs:nonNegativeInteger | pool_xda_p_reads - Lecturas físicas de datos de XDA de la agrupación de almacenamientos intermedios |
| pool_xda_writes | xs:nonNegativeInteger | pool_xda_writes - Grabaciones de datos de XDA de la agrupación de almacenamientos intermedios |
| num_log_buffer_full | xs:nonNegativeInteger | num_log_buffer_full - Número de almacenamientos intermedios de anotaciones cronológicas llenos |
| rows_modified | xs:nonNegativeInteger | rows_modified - Filas modificadas |
| rows_read | xs:nonNegativeInteger | rows_read - Filas leídas |

Tabla 94. Métricas detalladas devueltas para MON_GET_ACTIVITY_DETAILS (continuación)

| Nombre de elemento | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------------|-----------------------|--|
| rows_returned | xs:nonNegativeInteger | rows_returned - Filas devueltas |
| total_cpu_time | xs:nonNegativeInteger | total_cpu_time - Tiempo total de CPU |
| total_act_time | xs:nonNegativeInteger | total_act_time - Tiempo total de actividad |
| total_act_wait_time | xs:nonNegativeInteger | total_act_wait_time - Tiempo de espera total de actividad |
| total_section_sort_time | xs:nonNegativeInteger | total_section_sort_time - Tiempo total de clasificación de sección. |
| total_section_sort_proc_time | xs:nonNegativeInteger | total_section_sort_proc_time - Tiempo total de proceso de clasificación de sección |
| total_section_sorts | xs:nonNegativeInteger | total_section_sorts - Total de clasificaciones de sección. |
| total_sorts | xs:nonNegativeInteger | total_sorts - Número total de clasificaciones |
| post_threshold_sorts | xs:nonNegativeInteger | post_threshold_sorts - Clasificaciones de umbral posteriores |
| post_shrthreshold_sorts | xs:nonNegativeInteger | post_shrthreshold_sorts - Clasificaciones de umbral compartidas posteriores |
| sort_overflows | xs:nonNegativeInteger | sort_overflows - Desbordamientos de clasificación |
| executable_id | executable_id_type | executable_id - ID ejecutable |
| wlm_queue_time_total | xs:nonNegativeInteger | wlm_queue_time_total - Tiempo de cola total del gestor de cargas de trabajo |
| wlm_queue_assignments_total | xs:nonNegativeInteger | wlm_queue_assignments_total - Total de asignaciones de cola del gestor de cargas de trabajo |
| eff_stmt_text | xs:string | eff_stmt_text - Texto de sentencia efectivo. Los 1.024 caracteres iniciales del texto de sentencia concentrado que sigue a cualquier sustitución literal realizada por el concentrador de sentencias. Sólo está presente si el concentrador de sentencias está habilitado y modificó esta sentencia. |

Función de tabla MON_GET_BUFFERPOOL - Obtener métrica de agrupación de almacenamientos intermedios

La función de tabla MON_GET_BUFFERPOOL devuelve métricas del supervisor para una o varias agrupaciones de almacenamientos intermedios.

Sintaxis

```

▶▶—MON_GET_BUFFERPOOL—(—nombre_agrupación_almacenamiento_intermedio—, —————▶
▶—miembro—)—————▶▶

```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombre_agrupación_almacenamiento_intermedio

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de agrupación de almacenamientos intermedios válido de la base de datos conectada actualmente cuando se llama a esta función. Si el argumento es nulo

o una serie vacía, se recuperan métricas para todas las agrupaciones de almacenamientos intermedios de la base de datos.

miembro

Argumento de entrada de tipo INTEGER que especifica un miembro válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique -1 para el miembro de base de datos actual o -2 para todos los miembros de base de datos. Si se especifica el valor nulo, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE sobre la función MON_GET_BUFFERPOOL.

Ejemplo

Calcular la proporción de coincidencias de la agrupación de almacenamientos intermedios.

```
WITH BPMETRICS AS (
  SELECT bp_name,
         pool_data_l_reads + pool_temp_data_l_reads +
         pool_index_l_reads + pool_temp_index_l_reads +
         pool_xda_l_reads + pool_temp_xda_l_reads as logical_reads,
         pool_data_p_reads + pool_temp_data_p_reads +
         pool_index_p_reads + pool_temp_index_p_reads +
         pool_xda_p_reads + pool_temp_xda_p_reads as physical_reads,
         member FROM TABLE(MON_GET_BUFFERPOOL('',-2)) AS METRICS)
SELECT   VARCHAR(bp_name,20) AS bp_name,
         logical_reads,
         physical_reads,
         CASE WHEN logical_reads > 0
              THEN DEC((1 - (FLOAT(physical_reads) / FLOAT(logical_reads))) * 100,5,2)
              ELSE NULL
         END AS HIT_RATIO,
         dbpartitionnum FROM BPMETRICS;
```

A continuación se muestra un ejemplo de salida de esta consulta.

| BP_NAME | LOGICAL_READS | PHYSICAL_READS | HIT_RATIO | MEMBER |
|----------------|---------------|----------------|-----------|--------|
| IBMDEFAULTBP | 619 | 385 | 37.80 | 0 |
| IBMSYSTEMBP4K | 0 | 0 | - | 0 |
| IBMSYSTEMBP8K | 0 | 0 | - | 0 |
| IBMSYSTEMBP16K | 0 | 0 | - | 0 |
| IBMSYSTEMBP32K | 0 | 0 | - | 0 |

5 registro(s) seleccionado(s).

Salida de la consulta (continuación).

```
... HIT_RATIO MEMBER
... -----
...      37.80          0
...      -            0
...      -            0
...      -            0
...      -            0
```

Notas de uso

La función de tabla MON_GET_BUFFERPOOL devuelve una fila de datos por agrupación de almacenamientos intermedios de base de datos y por miembro de

base de datos. No se realiza una agregación en los miembros de base de datos. Sin embargo, se puede lograr la agregación mediante consultas de SQL, tal como se muestra en el ejemplo.

La métrica recopilada por esta función se controla a nivel de base de datos mediante el parámetro de configuración `mon_obj_metrics`. Por omisión, la recopilación de métrica está habilitada.

Información devuelta

Tabla 95. Información devuelta para `MON_GET_BUFFERPOOL`

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|---|
| BP_NAME | VARCHAR(128) | |
| MEMBER | SMALLINT | member - Miembro de base de datos |
| AUTOMATIC | SMALLINT | automatic - Agrupación de almacenamientos intermedios automática |
| DIRECT_READS | BIGINT | direct_reads - Lecturas directas de la base de datos |
| DIRECT_READ_REQS | BIGINT | direct_read_reqs - Peticiones de lectura directa |
| DIRECT_WRITES | BIGINT | direct_writes - Grabaciones directas en la base de datos |
| DIRECT_WRITE_REQS | BIGINT | direct_write_reqs - Peticiones de grabación directa |
| POOL_DATA_L_READS | BIGINT | pool_data_l_reads - Lecturas lógicas de datos de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_L_READS | BIGINT | pool_temp_data_l_reads - Lecturas lógicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_XDA_L_READS | BIGINT | pool_xda_l_reads - Lecturas lógicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_L_READS | BIGINT | pool_temp_xda_l_reads - Lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |
| POOL_INDEX_L_READS | BIGINT | pool_index_l_reads - Lecturas lógicas de índices de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_L_READS | BIGINT | pool_temp_index_l_reads - Lecturas lógicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_DATA_P_READS | BIGINT | pool_data_p_reads - Lecturas físicas de datos de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_P_READS | BIGINT | pool_temp_data_p_reads - Lecturas físicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_XDA_P_READS | BIGINT | pool_xda_p_reads - Lecturas físicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_P_READS | BIGINT | pool_temp_xda_p_reads - Lecturas físicas de datos XDA temporales de agrupación de almacenamientos intermedios |
| POOL_INDEX_P_READS | BIGINT | pool_index_p_reads - Lecturas físicas de índices de la agrupación de almacenamientos intermedios |

Tabla 95. Información devuelta para MON_GET_BUFFERPOOL (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|----------------------------|---------------|--|
| POOL_TEMP_INDEX_P_READS | BIGINT | pool_temp_index_p_reads - Lecturas físicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_DATA_WRITES | BIGINT | pool_data_writes - Grabaciones de datos de la agrupación de almacenamientos intermedios |
| POOL_XDA_WRITES | BIGINT | pool_xda_writes - Grabaciones de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_INDEX_WRITES | BIGINT | pool_index_writes - Grabaciones de índices de la agrupación de almacenamientos intermedios |
| DIRECT_READ_TIME | BIGINT | direct_read_time - Tiempo de lectura directa |
| DIRECT_WRITE_TIME | BIGINT | direct_write_time - Tiempo de grabación directa |
| POOL_READ_TIME | BIGINT | pool_read_time - Tiempo total de lectura física de la agrupación de almacenamientos intermedios |
| POOL_WRITE_TIME | BIGINT | pool_write_time - Tiempo total de grabación física de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_READS | BIGINT | pool_async_data_reads - Lecturas de datos asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_READ_REQS | BIGINT | pool_async_data_read_reqs - Peticiones de lectura asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_WRITES | BIGINT | pool_async_data_writes - Grabaciones de datos asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_READS | BIGINT | pool_async_index_reads - Lecturas de índice asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_READ_REQS | BIGINT | pool_async_index_read_reqs - Peticiones de lectura de índice asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_WRITES | BIGINT | pool_async_index_writes - Grabaciones de índice asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_XDA_READS | BIGINT | pool_async_xda_reads - Lecturas de datos XDA asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_XDA_READ_REQS | BIGINT | pool_async_xda_read_reqs - Peticiones de lectura XDA asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_XDA_WRITES | BIGINT | pool_async_xda_writes - Grabaciones de datos XDA asíncronas de agrupación de almacenamientos intermedios |
| POOL_NO_VICTIM_BUFFER | BIGINT | pool_no_victim_buffer - Almacenamiento intermedio sin víctimas de la agrupación de almacenamientos intermedios |
| POOL_LSN_GAP_CLNS | BIGINT | pool_lsn_gap_clns - Limpiadores de espacio de anotación cronológica de agrupación de almacenamientos intermedios activados |

Tabla 95. Información devuelta para MON_GET_BUFFERPOOL (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|--|
| POOL_DRTY_PG_STEAL_CLNS | BIGINT | pool_drtty_pg_steal_clns - Limpiadores de páginas víctima de agrupación de agrupación de almacenamientos intermedios activados |
| POOL_DRTY_PG_THRSH_CLNS | BIGINT | pool_drtty_pg_thrsh_clns - Limpiadores de umbral de agrupación de almacenamientos intermedios activados |
| VECTORED_IOS | BIGINT | vectored_ios - Número de peticiones de E/S de vector |
| PAGES_FROM_VECTORED_IOS | BIGINT | pages_from_vectored_ios - Número total de páginas leídas por E/S de vector |
| BLOCK_IOS | BIGINT | block_ios - Número de peticiones de E/S de bloque |
| PAGES_FROM_BLOCK_IOS | BIGINT | pages_from_block_ios - Número total de páginas leídas por E/S de bloque |
| UNREAD_PREFETCH_PAGES | BIGINT | unread_prefetch_pages - Páginas de captación previa no leídas |
| FILES_CLOSED | BIGINT | files_closed - Archivos de base de datos cerrados |
| ADDITIONAL_DETAILS | BLOB(100K) | Reservado para uso en el futuro |

Función de tabla MON_GET_CONNECTION - Obtener métrica de conexión

La función de tabla MON_GET_CONNECTION devuelve métricas para una o varias conexiones.

Sintaxis

►►—MON_GET_CONNECTION—(—*descriptor_contexto_aplicación*—, —*miembro*—)————►►

El esquema es SYSPROC.

Parámetros de la función de tabla

descriptor_contexto_aplicación

Argumento de entrada de tipo BIGINT que especifica un descriptor de contexto de aplicación determinado que identifica la conexión para la que se tienen que devolver métricas. Si el argumento es nulo, se devuelven las métricas correspondientes a todas las conexiones.

miembro

Argumento de entrada de tipo INTEGER que especifica un miembro válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique -1 para el miembro de base de datos actual o -2 para todos los miembros de base de datos. Si se especifica el valor nulo, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE sobre la función MON_GET_CONNECTION.

Ejemplo

Mostrar las conexiones que devuelven el volumen más alto de datos a los clientes, ordenadas por filas devueltas.

```
SELECT application_handle,  
       rows_returned,  
       tcpip_send_volume  
FROM TABLE(MON_GET_CONNECTION(cast(NULL as bigint), -2)) AS t  
ORDER BY rows_returned DESC
```

A continuación se muestra un ejemplo de salida de esta consulta.

| APPLICATION_HANDLE | ROWS_RETURNED | TCPIP_SEND_VOLUME |
|--------------------|---------------|-------------------|
| ----- | ----- | ----- |
| | 55 | 6 |
| | | 0 |

1 registro(s) seleccionado(s).

Notas de uso

Las métricas devueltas por la función de tabla `MON_GET_CONNECTION` representan la suma de todas las métricas correspondientes a las peticiones enviadas por una conexión. Las métricas se avanzan cuando llegan a los límites de la unidad de trabajo, y periódicamente durante la ejecución de las peticiones. Por consiguiente, los valores notificados por esta función de tabla reflejan el estado actual del sistema en el momento del avance más reciente. Las métricas son estrictamente de valor creciente. Para determinar el valor de una métrica dada durante un intervalo de tiempo, utilice la función de tabla `MON_GET_CONNECTION` para consultar la métrica al principio y al final del intervalo y calcular la diferencia.

Las métricas de petición se controlan mediante la cláusula `COLLECT REQUEST METRICS` en las superclases de servicio y el parámetro de configuración de base de datos `mon_req_metrics` en el nivel de la base de datos. Las métricas únicamente se recopilan para una petición si la petición se procesa a través de un agente en una subclase de servicio cuya superclase de servicio padre tiene habilitadas las métricas de petición o si la recopilación de métricas de petición está habilitada para toda la base de datos. Por omisión, las métricas de petición están habilitadas en el nivel de la base de datos. Si las métricas de petición están inhabilitadas en el nivel de la base de datos y para una superclase de servicio, las métricas notificadas para cada conexión correlacionada con dicha superclase de servicio dejarán de aumentar (o permanecerán en 0 si las métricas de petición estaban inhabilitadas en el momento de activarse la base de datos).

Consejo: Dado que una conexión se puede correlacionar con más de una superclase de servicio durante su vida útil, las métricas obtenidas mediante la función de tabla `MON_GET_CONNECTION` pueden representar un subconjunto de las métricas correspondientes a todas las peticiones enviadas a través de la conexión. Esto puede suceder si la recopilación de métricas está inhabilitada para algunas de las superclases que correlaciona la conexión.

La función de tabla `MON_GET_CONNECTION` devuelve una fila de datos por conexión y por miembro. No se realiza ninguna agregación entre miembros (correspondientes a una o varias clases de servicio). Sin embargo, se puede lograr la agregación mediante consultas de SQL.

Información devuelta

Tabla 96. Información devuelta para MON_GET_CONNECTION

| Nombre de columna | Tipo de datos | Descripción |
|-------------------------|---------------|--|
| APPLICATION_HANDLE | BIGINT | application_handle - Descriptor de contexto de aplicación |
| APPLICATION_NAME | VARCHAR(128) | Reservado para una utilización futura. |
| APPLICATION_ID | VARCHAR(128) | Reservado para una utilización futura. |
| MEMBER | SMALLINT | member - Miembro de base de datos |
| CLIENT_WRKSTNNAME | VARCHAR(255) | Registro especial CURRENT CLIENT_WRKSTNNAME |
| CLIENT_ACCTNG | VARCHAR(255) | Registro especial CURRENT CLIENT_ACCTNG |
| CLIENT_USERID | VARCHAR(255) | Registro especial CURRENT CLIENT_USERID |
| CLIENT_APPLNAME | VARCHAR(255) | Registro especial CURRENT CLIENT_APPLNAME |
| CLIENT_PID | BIGINT | Reservado para una utilización futura. |
| CLIENT_PRDID | VARCHAR(128) | Reservado para una utilización futura. |
| CLIENT_PLATFORM | VARCHAR(12) | Reservado para una utilización futura. |
| CLIENT_PROTOCOL | VARCHAR(10) | Reservado para una utilización futura. |
| SYSTEM_AUTH_ID | VARCHAR(128) | Reservado para una utilización futura. |
| SESSION_AUTH_ID | VARCHAR(128) | Reservado para una utilización futura. |
| COORD_MEMBER | SMALLINT | Reservado para una utilización futura. |
| CONNECTION_START_TIME | TIMESTAMP | Reservado para una utilización futura. |
| ACT_ABORTED_TOTAL | BIGINT | act_aborted_total - Total de actividades abortadas |
| ACT_COMPLETED_TOTAL | BIGINT | act_completed_total - Total de actividades completadas |
| ACT_REJECTED_TOTAL | BIGINT | act_rejected_total - Total de actividades rechazadas |
| AGENT_WAIT_TIME | BIGINT | agent_wait_time - Tiempo de espera del agente |
| AGENT_WAITS_TOTAL | BIGINT | agent_waits_total - Total de esperas del agente |
| POOL_DATA_L_READS | BIGINT | pool_data_l_reads - Lecturas lógicas de datos de la agrupación de almacenamientos intermedios |
| POOL_INDEX_L_READS | BIGINT | pool_index_l_reads - Lecturas lógicas de índices de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_L_READS | BIGINT | pool_temp_data_l_reads - Lecturas lógicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_L_READS | BIGINT | pool_temp_index_l_reads - Lecturas lógicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_L_READS | BIGINT | pool_temp_xda_l_reads - Lecturas lógicas de datos XDA temporales de agrupación de almacenamientos intermedios |
| POOL_XDA_L_READS | BIGINT | pool_xda_l_reads - Lecturas lógicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_DATA_P_READS | BIGINT | pool_data_p_reads - Lecturas físicas de datos de la agrupación de almacenamientos intermedios |

Tabla 96. Información devuelta para MON_GET_CONNECTION (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------------|---------------|--|
| POOL_INDEX_P_READS | BIGINT | pool_index_p_reads - Lecturas físicas de índices de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_P_READS | BIGINT | pool_temp_data_p_reads - Lecturas físicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_P_READS | BIGINT | pool_temp_index_p_reads - Lecturas físicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_P_READS | BIGINT | pool_temp_xda_p_reads - Lecturas físicas de datos XDA temporales de agrupación de almacenamientos intermedios |
| POOL_XDA_P_READS | BIGINT | pool_xda_p_reads - Lecturas físicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_DATA_WRITES | BIGINT | pool_data_writes - Grabaciones de datos de agrupación de almacenamientos intermedios |
| POOL_INDEX_WRITES | BIGINT | pool_index_writes - Grabaciones de índices de la agrupación de almacenamientos intermedios |
| POOL_XDA_WRITES | BIGINT | pool_xda_writes - Grabaciones de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_READ_TIME | BIGINT | pool_read_time - Tiempo total de lectura física de la agrupación de almacenamientos intermedios |
| POOL_WRITE_TIME | BIGINT | pool_write_time - Tiempo total de grabación física de agrupación de almacenamientos intermedios |
| CLIENT_IDLE_WAIT_TIME | BIGINT | client_idle_wait_time - Tiempo de espera de inactividad del cliente |
| DEADLOCKS | BIGINT | deadlocks - Puntos muertos detectados |
| DIRECT_READS | BIGINT | direct_reads - Lecturas directas de la base de datos |
| DIRECT_READ_TIME | BIGINT | direct_read_time - Tiempo de lectura directa |
| DIRECT_WRITES | BIGINT | direct_writes - Grabaciones directas en la base de datos |
| DIRECT_WRITE_TIME | BIGINT | direct_write_time - Tiempo de grabación directa |
| DIRECT_READ_REQS | BIGINT | direct_read_reqs - Peticiones de lectura directa |
| DIRECT_WRITE_REQS | BIGINT | direct_write_reqs - Peticiones de grabación directa |
| FCM_RECV_VOLUME | BIGINT | fcm_recv_volume - Volumen de recepción de FCM |
| FCM_RECVS_TOTAL | BIGINT | fcm_recvs_total - Total de recepciones de FCM |
| FCM_SEND_VOLUME | BIGINT | fcm_send_volume - Volumen de envío de FCM |
| FCM_SENDS_TOTAL | BIGINT | fcm_sends_total - Total de envíos de FCM |
| FCM_RECV_WAIT_TIME | BIGINT | fcm_recv_wait_time - Tiempo de espera de recepción de FCM |
| FCM_SEND_WAIT_TIME | BIGINT | fcm_send_wait_time - Tiempo de espera de envío de FCM |
| IPC_RECV_VOLUME | BIGINT | ipc_recv_volume - Volumen de recepción de comunicación entre procesos |
| IPC_RECV_WAIT_TIME | BIGINT | ipc_recv_wait_time - Tiempo de espera de recepción de comunicación entre procesos |

Tabla 96. Información devuelta para MON_GET_CONNECTION (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-----------------------------|---------------|--|
| IPC_RECVS_TOTAL | BIGINT | ipc_recv_total - Total de recepciones de comunicación entre procesos |
| IPC_SEND_VOLUME | BIGINT | ipc_send_volume - Volumen de envío de comunicación entre procesos |
| IPC_SEND_WAIT_TIME | BIGINT | ipc_send_wait_time - Tiempo de espera de envío de comunicación entre procesos |
| IPC_SENDS_TOTAL | BIGINT | ipc_sends_total - Total de envíos de comunicación entre procesos |
| LOCK_ESCALS | BIGINT | lock_escals - Número de escalamientos de bloqueo |
| LOCK_TIMEOUTS | BIGINT | lock_timeouts - Número de tiempos de espera de bloqueo |
| LOCK_WAIT_TIME | BIGINT | lock_wait_time - Tiempo esperado en bloqueos |
| LOCK_WAITS | BIGINT | lock_waits - Esperas de bloqueo |
| LOG_BUFFER_WAIT_TIME | BIGINT | log_buffer_wait_time - Tiempo de espera de almacenamiento intermedio de anotaciones cronológicas |
| NUM_LOG_BUFFER_FULL | BIGINT | num_log_buffer_full - Número de almacenamientos intermedios de anotaciones cronológicas llenos |
| LOG_DISK_WAIT_TIME | BIGINT | log_disk_wait_time - Tiempo de espera de disco de anotaciones cronológicas |
| LOG_DISK_WAITS_TOTAL | BIGINT | log_disk_waits_total - Total de esperas de disco de anotaciones cronológicas |
| NUM_LOCKS_HELD | BIGINT | Reservado para una utilización futura. |
| RQSTS_COMPLETED_TOTAL | BIGINT | rqsts_completed_total - Total de peticiones completadas |
| ROWS_MODIFIED | BIGINT | rows_modified - Filas modificadas |
| ROWS_READ | BIGINT | rows_read - Filas leídas |
| ROWS_RETURNED | BIGINT | rows_returned - Filas devueltas |
| TCPIP_RECV_VOLUME | BIGINT | tcpip_recv_volume - Volumen recibido TCP/IP |
| TCPIP_SEND_VOLUME | BIGINT | tcpip_send_volume - Volumen de envío TCP/IP |
| TCPIP_RECV_WAIT_TIME | BIGINT | tcpip_recv_wait_time - Tiempo de espera de recepción TCP/IP |
| TCPIP_RECVS_TOTAL | BIGINT | tcpip_recv_total - Total de recepciones TCP/IP |
| TCPIP_SEND_WAIT_TIME | BIGINT | tcpip_send_wait_time - Tiempo de espera de envío TCP/IP |
| TCPIP_SENDS_TOTAL | BIGINT | tcpip_sends_total - Total de envíos TCP/IP |
| TOTAL_APP_RQST_TIME | BIGINT | total_app_rqst_time - Total de tiempo de petición de aplicaciones |
| TOTAL_RQST_TIME | BIGINT | total_rqst_time - Tiempo total de petición |
| WLM_QUEUE_TIME_TOTAL | BIGINT | wlm_queue_time_total - Tiempo de cola total del gestor de cargas de trabajo |
| WLM_QUEUE_ASSIGNMENTS_TOTAL | BIGINT | wlm_queue_assignments_total - Total de asignaciones de cola del gestor de cargas de trabajo |
| TOTAL_CPU_TIME | BIGINT | total_cpu_time - Tiempo total de CPU |
| TOTAL_WAIT_TIME | BIGINT | total_wait_time - Tiempo total de espera |

Tabla 96. Información devuelta para MON_GET_CONNECTION (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|----------------------------------|---------------|--|
| APP_RQSTS_COMPLETED_TOTAL | BIGINT | app_rqsts_completed_total - Total de peticiones de aplicación completadas |
| TOTAL_SECTION_SORT_TIME | BIGINT | total_section_sort_time - Tiempo total de clasificación de sección |
| TOTAL_SECTION_SORT_PROC_TIME | BIGINT | total_section_sort_proc_time - Tiempo total de proceso de clasificación de sección |
| TOTAL_SECTION_SORTS | BIGINT | total_section_sorts - Total de clasificaciones de sección |
| TOTAL_SORTS | BIGINT | total_sorts - Número total de clasificaciones |
| POST_THRESHOLD_SORTS | BIGINT | post_threshold_sorts - Clasificaciones de umbral posteriores |
| POST_SHRTHRESHOLD_SORTS | BIGINT | post_shrthreshold_sorts - Clasificaciones de umbral compartidas posteriores |
| SORT_OVERFLOWS | BIGINT | sort_overflows - Desbordamientos de clasificación |
| TOTAL_COMPILE_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_COMPILE_PROC_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_COMPILATIONS | BIGINT | Reservado para una utilización futura. |
| TOTAL_IMPLICIT_COMPILE_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_IMPLICIT_COMPILE_PROC_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_IMPLICIT_COMPILATIONS | BIGINT | Reservado para una utilización futura. |
| TOTAL_SECTION_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_SECTION_PROC_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_APP_SECTION_EXECUTIONS | BIGINT | Reservado para una utilización futura. |
| TOTAL_ACT_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_ACT_WAIT_TIME | BIGINT | Reservado para una utilización futura. |
| ACT_RQSTS_TOTAL | BIGINT | Reservado para una utilización futura. |
| TOTAL_ROUTINE_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_ROUTINE_INVOCATIONS | BIGINT | Reservado para una utilización futura. |
| TOTAL_COMMIT_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_COMMIT_PROC_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_APP_COMMITS | BIGINT | Reservado para una utilización futura. |
| INT_COMMITS | BIGINT | Reservado para una utilización futura. |
| TOTAL_ROLLBACK_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_ROLLBACK_PROC_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_APP_ROLLBACKS | BIGINT | Reservado para una utilización futura. |
| INT_ROLLBACKS | BIGINT | Reservado para una utilización futura. |
| TOTAL_RUNSTATS_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_RUNSTATS_PROC_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_RUNSTATS | BIGINT | Reservado para una utilización futura. |
| TOTAL_REORG_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_REORG_PROC_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_REORGS | BIGINT | Reservado para una utilización futura. |

Tabla 96. Información devuelta para MON_GET_CONNECTION (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|------------------------|---------------|--|
| TOTAL_LOAD_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_LOAD_PROC_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_LOADS | BIGINT | Reservado para una utilización futura. |
| CAT_CACHE_INSERTS | BIGINT | Reservado para una utilización futura. |
| CAT_CACHE_LOOKUPS | BIGINT | Reservado para una utilización futura. |
| PKG_CACHE_INSERTS | BIGINT | Reservado para una utilización futura. |
| PKG_CACHE_LOOKUPS | BIGINT | Reservado para una utilización futura. |
| THRESH_VIOLATIONS | BIGINT | Reservado para una utilización futura. |
| NUM_LW_THRESH_EXCEEDED | BIGINT | Reservado para una utilización futura. |
| ADDITIONAL_DETAILS | BLOB(100K) | Reservado para uso en el futuro |

Función de tabla MON_GET_CONNECTION_DETAILS - Obtener métrica de conexión detallada

La función de tabla MON_GET_CONNECTION_DETAILS devuelve métricas detalladas para una o varias conexiones.

Sintaxis

```
►►—MON_GET_CONNECTION_DETAILS—(—descriptor_contexto_aplicación—,—————►
►—miembro—)—————►►
```

El esquema es SYSPROC.

Parámetros de la función de tabla

descriptor_contexto_aplicación

Argumento de entrada de tipo BIGINT que especifica un descriptor de contexto de aplicación determinado que identifica la conexión para la que se tienen que devolver métricas. Si el argumento es NULL, se devuelven las métricas correspondientes a todas las conexiones.

miembro

Argumento de entrada de tipo INTEGER que especifica un miembro válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique -1 para el miembro de base de datos actual o -2 para todos los miembros de base de datos. Si se especifica el valor nulo, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE sobre la función MON_GET_CONNECTION_DETAILS.

Ejemplo

Mostrar las conexiones que devuelven el volumen más alto de datos a los clientes, ordenadas por filas devueltas.

```

SELECT detmetrics.application_handle,
       detmetrics.rows_returned,
       detmetrics.tcPIP_send_volume
FROM TABLE(MON_GET_CONNECTION_DETAILS(CAST(NULL as bigint), -2))
AS CONNMETRICS,
XMLTABLE (XMLNAMESPACES( DEFAULT 'http://www.ibm.com/xmlns/prod/db2/mon'),
 '$detmetric/db2_connection' PASSING XMLPARSE(DOCUMENT CONNMETRICS.DETAILS)
 as "detmetric"
COLUMNS "APPLICATION_HANDLE" INTEGER PATH 'application_handle',
 "ROWS_RETURNED" BIGINT PATH 'system_metrics/rows_returned',
 "TCPIP_SEND_VOLUME" BIGINT PATH 'system_metrics/tcpip_send_volume'
 ) AS DETMETRICS
ORDER BY rows_returned DESC

```

A continuación se muestra un ejemplo de salida de esta consulta.

| APPLICATION_HANDLE | ROWS_RETURNED | TCPIP_SEND_VOLUME |
|--------------------|---------------|-------------------|
| ----- | ----- | ----- |
| | 21 | 4 |
| | | 0 |

1 registro(s) seleccionado(s).

Notas de uso

Las métricas devueltas por la función de tabla `MON_GET_CONNECTION_DETAILS` representan la suma de todas las métricas correspondientes a las peticiones enviadas por una conexión. Esta función es parecida a la función de tabla `MON_GET_CONNECTION`:

- La función de tabla `MON_GET_CONNECTION` devuelve las métricas utilizadas con más frecuencia en un formato basado en columnas, y es el método más eficiente, desde el punto de vista del rendimiento, de recuperar las métricas.
- La función de tabla `MON_GET_CONNECTION_DETAILS` devuelve todo el conjunto de métricas disponibles en formato de documento XML, lo que proporciona la máxima flexibilidad para dar formato a la salida. La salida basada en XML puede analizarse directamente mediante un analizador XML, o puede convertirse a formato relacional mediante la función `XMLTABLE` (véase el ejemplo).

Las métricas se avanzan cuando llegan a los límites de la unidad de trabajo, y periódicamente durante la ejecución de las peticiones. Por consiguiente, los valores notificados por esta función de tabla reflejan el estado actual del sistema en el momento del avance más reciente. Las métricas son estrictamente de valor creciente. Para determinar el valor de una métrica dada durante un intervalo de tiempo, utilice la función de tabla `MON_GET_CONNECTION_DETAILS` para consultar la métrica al principio y al final del intervalo y calcular la diferencia.

Las métricas de petición se controlan mediante la cláusula `COLLECT REQUEST METRICS` en las superclases de servicio y el parámetro de configuración de base de datos `mon_req_metrics` en el nivel de la base de datos. Las métricas únicamente se recopilan para una petición si la petición se procesa a través de un agente en una subclase de servicio cuya superclase de servicio padre tiene habilitadas las métricas de petición o si la recopilación de métricas de petición está habilitada para toda la base de datos. Por omisión, las métricas de petición están habilitadas en el nivel de la base de datos. Si las métricas de petición están inhabilitadas en el nivel de la base de datos para una superclase de servicio, las métricas notificadas para cada conexión correlacionada con dicha superclase de servicio dejan de aumentar (o permanecen en 0 si las métricas de petición estaban inhabilitadas en el momento de activarse la base de datos).

Consejo: Puesto que una conexión se puede correlacionar con más de una superclase de servicio durante su vida útil, si la recopilación está inhabilitada a nivel de base de datos, las métricas obtenidas mediante la función de tabla `MON_GET_CONNECTION_DETAILS` pueden representar un subconjunto de las métricas correspondientes a todas las peticiones enviadas a través de la conexión. Esto puede suceder si la recopilación de métricas está inhabilitada para algunas de las superclases con las que la conexión se correlaciona.

La función de tabla `MON_GET_CONNECTION_DETAILS` devuelve una fila de datos por conexión y por miembro. No se realiza ninguna agregación entre miembros (para una o varias clases de servicio). Sin embargo, se puede lograr la agregación mediante consultas de SQL.

El esquema para el documento XML devuelto en la columna `DETAILS` está disponible en el archivo `sql1ib/misc/DB2MonRoutines.xsd`. Para obtener más información, consulte el archivo `sql1ib/misc/DB2MonCommon.xsd`.

Información devuelta

Tabla 97. Información devuelta para `MON_GET_CONNECTION_DETAILS`

| Nombre de columna | Tipo de datos | Descripción |
|---------------------------------|-----------------------|--|
| <code>APPLICATION_HANDLE</code> | <code>BIGINT</code> | <code>application_handle</code> - Descriptor de contexto de aplicación |
| <code>MEMBER</code> | <code>SMALLINT</code> | <code>member</code> - Miembro de base de datos |
| <code>DETAILS</code> | <code>BLOB(1M)</code> | Documento XML que contiene métricas detalladas correspondientes a la unidad de trabajo. Consulte la Tabla 98 en la página 392 para ver una descripción de los elementos de este documento. |

El ejemplo siguiente muestra la estructura del documento XML que se devuelve en la columna `DETAILS`.

```
<db2_connection xmlns="http://www.ibm.com/xmlns/prod/db2/mon" release="90700000">
  <application_handle>21</application_handle>
  <member>0</member>
  <system_metrics release="90700000">
    <act_aborted_total>5</act_aborted_total>
    ...
    <wlm_queue_assignments_total>3</wlm_queue_assignments_total>
  </system_metrics>
</db2_connection>
```

Para ver el esquema completo, consulte `sql1ib/misc/DB2MonRoutines.xsd`.

Este documento emplea las definiciones de tipo no primitivo XML siguientes:

```
<xs:simpleType name="db2DbObjectString">
  <xs:restriction base="xs:string">
    <xs:maxLength value="128"/>
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name="db2PartitionNum">
  <xs:restriction base="xs:nonNegativeInteger">
    <xs:maxInclusive value="999"/>
  </xs:restriction>
</xs:simpleType>
```

Tabla 98. Métricas detalladas devueltas para MON_GET_CONNECTION_DETAILS

| Nombre de elemento | Tipo de datos | Descripción |
|-----------------------------|-----------------------|--|
| application_handle | xs:nonNegativeInteger | application_handle - Descriptor de contexto de aplicación |
| client_userid | xs:string (255) | Registro especial CURRENT CLIENT_USERID |
| client_wrkstnname | xs:string (255) | Registro especial CURRENT CLIENT_WRKSTNNAME |
| client_applname | xs:string (255) | Registro especial CURRENT CLIENT_APPLNAME |
| client_acctng | xs:string (255) | Registro especial CURRENT CLIENT_ACCTNG |
| member | xs:nonNegativeInteger | member - Miembro de base de datos |
| act_aborted_total | xs:nonNegativeInteger | act_aborted_total - Total de actividades abortadas |
| act_completed_total | xs:nonNegativeInteger | act_completed_total - Total de actividades completadas |
| act_rejected_total | xs:nonNegativeInteger | act_rejected_total - Total de actividades rechazadas |
| agent_wait_time | xs:nonNegativeInteger | agent_wait_time - Tiempo de espera del agente |
| agent_waits_total | xs:nonNegativeInteger | agent_waits_total - Total de esperas del agente |
| app_rqsts_completed_total | xs:nonNegativeInteger | app_rqsts_completed_total - Total de peticiones de aplicación completadas |
| audit_events_total | xs:nonNegativeInteger | audit_events_total - Total de sucesos de auditoría |
| audit_subsystem_wait_time | xs:nonNegativeInteger | audit_subsystem_wait_time - Tiempo de espera del subsistema de auditoría |
| audit_subsystem_waits_total | xs:nonNegativeInteger | audit_subsystem_waits_total - Total de esperas del subsistema de auditoría |
| audit_file_write_wait_time | xs:nonNegativeInteger | audit_file_write_wait_time - Tiempo de espera de grabación del archivo de auditoría |
| audit_file_writes_total | xs:nonNegativeInteger | audit_file_writes_total - Total de archivos de auditoría grabados |
| client_idle_wait_time | xs:nonNegativeInteger | client_idle_wait_time - Tiempo de espera de inactividad del cliente |
| deadlocks | xs:nonNegativeInteger | deadlocks - Puntos muertos detectados |
| diaglog_writes_total | xs:nonNegativeInteger | diaglog_writes_total - Total de grabaciones de anotaciones cronológicas de diagnóstico |
| diaglog_write_wait_time | xs:nonNegativeInteger | diaglog_write_wait_time - Tiempo de grabación de anotaciones cronológicas de diagnóstico |
| direct_read_time | xs:nonNegativeInteger | direct_read_time - Tiempo de lectura directa |
| direct_write_time | xs:nonNegativeInteger | direct_write_time - Tiempo de grabación directa |
| direct_read_reqs | xs:nonNegativeInteger | direct_read_reqs - Peticiones de lectura directa |
| direct_reads | xs:nonNegativeInteger | direct_reads - Lecturas directas de la base de datos |
| direct_write_reqs | xs:nonNegativeInteger | direct_write_reqs - Peticiones de grabación directa |
| direct_writes | xs:nonNegativeInteger | direct_writes - Grabaciones directas en la base de datos |
| fcm_recv_volume | xs:nonNegativeInteger | fcm_recv_volume - Volumen de recepción de FCM |
| fcm_recv_wait_time | xs:nonNegativeInteger | fcm_recv_wait_time - Tiempo de espera de recepción de FCM |
| fcm_recvs_total | xs:nonNegativeInteger | fcm_recvs_total - Total de recepciones de FCM |
| fcm_message_recv_volume | xs:nonNegativeInteger | fcm_message_recv_volume - Volumen de recepción de mensaje de FCM |

Tabla 98. Métricas detalladas devueltas para MON_GET_CONNECTION_DETAILS (continuación)

| Nombre de elemento | Tipo de datos | Descripción |
|----------------------------|-----------------------|--|
| fcm_message_recvs_total | xs:nonNegativeInteger | fcm_message_recvs_total - Total de recepciones de mensaje de FCM |
| fcm_message_recv_wait_time | xs:nonNegativeInteger | fcm_message_recv_wait_time - Tiempo de espera de recepción de mensaje de FCM |
| fcm_message_send_volume | xs:nonNegativeInteger | fcm_message_send_volume - Volumen de envío de mensaje de FCM |
| fcm_message_send_wait_time | xs:nonNegativeInteger | fcm_message_send_wait_time - Tiempo de espera de envío de mensaje de FCM |
| fcm_message_sends_total | xs:nonNegativeInteger | fcm_message_sends_total - Total de envíos de mensaje de FCM |
| fcm_send_volume | xs:nonNegativeInteger | fcm_send_volume - Volumen de envío de FCM |
| fcm_send_wait_time | xs:nonNegativeInteger | fcm_send_wait_time - Tiempo de espera de envío de FCM |
| fcm_sends_total | xs:nonNegativeInteger | fcm_sends_total - Total de envíos de FCM |
| fcm_tq_recv_wait_time | xs:nonNegativeInteger | fcm_tq_recv_wait_time - Tiempo de espera de recepción de cola de tabla de FCM |
| fcm_tq_send_wait_time | xs:nonNegativeInteger | fcm_tq_send_wait_time - Tiempo de espera de envío de cola de tabla de FCM |
| fcm_tq_recv_volume | xs:nonNegativeInteger | fcm_tq_recv_volume - Volumen de recepción de cola de tabla de FCM |
| fcm_tq_recvs_total | xs:nonNegativeInteger | fcm_tq_recvs_total - Total de recepciones de cola de tabla de FCM |
| fcm_tq_send_volume | xs:nonNegativeInteger | fcm_tq_send_volume - Volumen de envíos de cola de tabla de FCM |
| fcm_tq_sends_total | xs:nonNegativeInteger | fcm_tq_sends_total - Total de envíos de cola de tabla de FCM |
| tq_tot_send_spills | xs:nonNegativeInteger | tq_tot_send_spills - Total de almacenamientos intermedios de cola de tabla desbordados |
| ipc_recv_volume | xs:nonNegativeInteger | ipc_recv_volume - Volumen de recepción de comunicación entre procesos |
| ipc_recv_wait_time | xs:nonNegativeInteger | ipc_recv_wait_time - Tiempo de espera de recepción de comunicación entre procesos |
| ipc_recvs_total | xs:nonNegativeInteger | ipc_recvs_total - Total de recepciones de comunicación entre procesos |
| ipc_send_volume | xs:nonNegativeInteger | ipc_send_volume - Volumen de envío de comunicación entre procesos |
| ipc_send_wait_time | xs:nonNegativeInteger | ipc_send_wait_time - Tiempo de espera de envío de comunicación entre procesos |
| ipc_sends_total | xs:nonNegativeInteger | ipc_sends_total - Total de envíos de comunicación entre procesos |
| lock_escals | xs:nonNegativeInteger | lock_escals - Número de escalamientos de bloqueo |
| lock_timeouts | xs:nonNegativeInteger | lock_timeouts - Número de tiempos de espera de bloqueo |
| lock_wait_time | xs:nonNegativeInteger | lock_wait_time - Tiempo esperado en bloqueos |
| lock_waits | xs:nonNegativeInteger | lock_waits - Esperas de bloqueo |

Tabla 98. Métricas detalladas devueltas para MON_GET_CONNECTION_DETAILS (continuación)

| Nombre de elemento | Tipo de datos | Descripción |
|-------------------------|-----------------------|---|
| log_buffer_wait_time | xs:nonNegativeInteger | log_buffer_wait_time - Tiempo de espera de almacenamiento intermedio de anotaciones cronológicas |
| log_disk_wait_time | xs:nonNegativeInteger | log_disk_wait_time - Tiempo de espera de disco de anotaciones cronológicas |
| log_disk_waits_total | xs:nonNegativeInteger | log_disk_waits_total - Total de esperas de disco de anotaciones cronológicas |
| pool_data_l_reads | xs:nonNegativeInteger | pool_data_l_reads - Lecturas lógicas de datos de la agrupación de almacenamientos intermedios |
| pool_data_p_reads | xs:nonNegativeInteger | pool_data_p_reads - Lecturas físicas de datos de la agrupación de almacenamientos intermedios |
| pool_data_writes | xs:nonNegativeInteger | pool_data_writes - Grabaciones de datos de la agrupación de almacenamientos intermedios |
| pool_index_l_reads | xs:nonNegativeInteger | pool_index_l_reads - Lecturas lógicas de índices de la agrupación de almacenamientos intermedios |
| pool_index_p_reads | xs:nonNegativeInteger | pool_index_p_reads - Lecturas físicas de índices de la agrupación de almacenamientos intermedios |
| pool_index_writes | xs:nonNegativeInteger | pool_index_writes - Grabaciones de índices de la agrupación de almacenamientos intermedios |
| pool_read_time | xs:nonNegativeInteger | pool_read_time - Tiempo total de lectura física de la agrupación de almacenamientos intermedios |
| pool_temp_data_l_reads | xs:nonNegativeInteger | pool_temp_data_l_reads - Lecturas lógicas de datos temporales de la agrupación de almacenamientos intermedios |
| pool_temp_data_p_reads | xs:nonNegativeInteger | pool_temp_data_p_reads - Lecturas físicas de datos temporales de la agrupación de almacenamientos intermedios |
| pool_temp_index_l_reads | xs:nonNegativeInteger | pool_temp_index_l_reads - Lecturas lógicas de índices temporales de la agrupación de almacenamientos intermedios |
| pool_temp_index_p_reads | xs:nonNegativeInteger | pool_temp_index_p_reads - Lecturas físicas de índices temporales de la agrupación de almacenamientos intermedios |
| pool_temp_xda_l_reads | xs:nonNegativeInteger | pool_temp_xda_l_reads - Lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |
| pool_temp_xda_p_reads | xs:nonNegativeInteger | pool_temp_xda_p_reads - Lecturas físicas de datos XDA temporales de agrupación de almacenamientos intermedios |
| pool_write_time | xs:nonNegativeInteger | pool_write_time - Tiempo total de grabación física de agrupación de almacenamientos intermedios |
| pool_xda_l_reads | xs:nonNegativeInteger | pool_xda_l_reads - Lecturas lógicas de datos de XDA de la agrupación de almacenamientos intermedios |
| pool_xda_p_reads | xs:nonNegativeInteger | pool_xda_p_reads - Lecturas físicas de datos de XDA de la agrupación de almacenamientos intermedios |
| pool_xda_writes | xs:nonNegativeInteger | pool_xda_writes - Grabaciones de datos de XDA de la agrupación de almacenamientos intermedios |

Tabla 98. Métricas detalladas devueltas para MON_GET_CONNECTION_DETAILS (continuación)

| Nombre de elemento | Tipo de datos | Descripción |
|------------------------------|-----------------------|--|
| num_log_buffer_full | xs:nonNegativeInteger | num_log_buffer_full - Número de almacenamientos intermedios de anotaciones cronológicas llenos |
| rqsts_completed_total | xs:nonNegativeInteger | rqsts_completed_total - Total de peticiones completadas |
| rows_modified | xs:nonNegativeInteger | rows_modified - Filas modificadas |
| rows_read | xs:nonNegativeInteger | rows_read - Filas leídas |
| rows_returned | xs:nonNegativeInteger | rows_returned - Filas devueltas |
| tcpip_recv_volume | xs:nonNegativeInteger | tcpip_recv_volume - Volumen recibido TCP/IP |
| tcpip_recv_wait_time | xs:nonNegativeInteger | tcpip_recv_wait_time - Tiempo de espera de recepción TCP/IP |
| tcpip_recvs_total | xs:nonNegativeInteger | tcpip_recvs_total - Total de recepciones TCP/IP |
| tcpip_send_volume | xs:nonNegativeInteger | tcpip_send_volume - Volumen de envío TCP/IP |
| tcpip_send_wait_time | xs:nonNegativeInteger | tcpip_send_wait_time - Tiempo de espera de envío TCP/IP |
| tcpip_sends_total | xs:nonNegativeInteger | tcpip_sends_total - Total de envíos TCP/IP |
| total_app_rqst_time | xs:nonNegativeInteger | total_app_rqst_time - Total de tiempo de petición de aplicaciones |
| total_cpu_time | xs:nonNegativeInteger | total_cpu_time - Tiempo total de CPU |
| total_rqst_time | xs:nonNegativeInteger | total_rqst_time - Tiempo total de petición |
| total_wait_time | xs:nonNegativeInteger | total_wait_time - Tiempo total de espera |
| total_section_sort_time | xs:nonNegativeInteger | total_section_sort_time - Tiempo total de clasificación de sección |
| total_section_sort_proc_time | xs:nonNegativeInteger | total_section_sort_proc_time - Tiempo total de proceso de clasificación de sección |
| total_section_sorts | xs:nonNegativeInteger | total_section_sorts - Total de clasificaciones de sección |
| total_sorts | xs:nonNegativeInteger | total_sorts - Número total de clasificaciones |
| post_threshold_sorts | xs:nonNegativeInteger | post_threshold_sorts - Clasificaciones de umbral posteriores |
| post_shrthreshold_sorts | xs:nonNegativeInteger | post_shrthreshold_sorts - Clasificaciones de umbral compartidas posteriores |
| sort_overflows | xs:nonNegativeInteger | sort_overflows - Desbordamientos de clasificación |
| tq_tot_send_spills | xs:nonNegativeInteger | tq_tot_send_spills - Número total de almacenamientos intermedios de colas de tabla desbordados |
| wlm_queue_time_total | xs:nonNegativeInteger | wlm_queue_time_total - Tiempo de cola total del gestor de cargas de trabajo |
| wlm_queue_assignments_total | xs:nonNegativeInteger | wlm_queue_assignments_total - Total de asignaciones de cola del gestor de cargas de trabajo |

Función de tabla MON_GET_CONTAINER - Obtener métrica del contenedor de espacio de tablas

La función de tabla MON_GET_CONTAINER devuelve métricas del supervisor para uno o varios contenedores de espacios de tablas.

Sintaxis

►—MON_GET_CONTAINER—(—nombre_espaciotablas—,—miembro—)—————►

El esquema es SYSPROC.

Parámetros de la función de tabla

nombre_espaciotablas

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de espacio de tablas válido que está en la misma base de datos a la que se está conectado al llamar a esta función. Si el argumento es nulo o una serie vacía, se devuelven métricas para todos los contenedores de todos los espacios de tablas de la base de datos.

miembro

Argumento de entrada de tipo INTEGER que especifica un miembro válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique -1 para el miembro de base de datos actual o -2 para todos los miembros de base de datos. Si se especifica el valor nulo, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE sobre la función MON_GET_CONTAINER.

Ejemplo

Ejemplo 1: Enumerar los contenedores de todos los miembros de base de datos que tengan el tiempo de lectura más largo.

```
SELECT varchar(container_name,70) as container_name,
       varchar(tbsp_name,20) as tbsp_name,
       pool_read_time
FROM TABLE(MON_GET_CONTAINER('',-2)) AS t
ORDER BY pool_read_time DESC
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
CONTAINER_NAME          ...
-----
/home/hote155/swalkty/swalkty/NODE0000/TEST/T0000000/C0000000.CAT ...
/home/hote155/swalkty/swalkty/NODE0000/TEST/T0000002/C0000000.LRG ...
/home/hote155/swalkty/swalkty/NODE0000/TEST/T0000001/C0000000.TMP ...
```

3 registro(s) seleccionado(s).

Salida de la consulta (continuación).

```
... TBSP_NAME          POOL_READ_TIME
... -----
... SYSCATSPACE          597
... USERSPACE1           42
... TEMPSPACE1           0
```

Ejemplo 2: Enumerar cualquier contenedor que no sea accesible.

```
SELECT varchar(container_name, 70) as container_name
FROM TABLE(MON_GET_CONTAINER('',-1)) AS t
WHERE accessible = 0
```

A continuación se muestra un ejemplo de salida de esta consulta.


```
CONTAINER_NAME-----
0 registro(s) seleccionado(s).
```

Ejemplo 3: Enumerar el uso de los sistemas de archivos de contenedor, ordenados por la utilización más alta.

```
SELECT varchar(container_name, 65) as container_name,
       fs_id,
       fs_used_size,
       fs_total_size,
       CASE WHEN fs_total_size > 0
            THEN DEC(100*(FLOAT(fs_used_size)/FLOAT(fs_total_size)),5,2)
            ELSE DEC(-1,5,2)
       END as utilization
FROM TABLE(MON_GET_CONTAINER('','-1)) AS t
ORDER BY utilization DESC
```

A continuación se muestra un ejemplo de salida de esta consulta.

```
CONTAINER_NAME ...
-----
/home/hotel55/swalkty/swalkty/NODE0000/TEST/T0000000/C0000000.CAT ...
/home/hotel55/swalkty/swalkty/NODE0000/TEST/T0000001/C0000000.TMP ...
/home/hotel55/swalkty/swalkty/NODE0000/TEST/T0000002/C0000000.LRG ...
```

3 registro(s) seleccionado(s).

Salida de la consulta (continuación).

| FS_ID | FS_USED_SIZE | FS_TOTAL_SIZE | UTILIZATION |
|-------|--------------|---------------|-------------|
| 64768 | 106879311872 | 317068410880 | 33.70 |
| 64768 | 106879311872 | 317068410880 | 33.70 |
| 64768 | 106879311872 | 317068410880 | 33.70 |

Notas de uso

La función de tabla `MON_GET_CONTAINER` devuelve una fila de datos por contenedor y por miembro de base de datos. Los datos pueden devolverse para todos los contenedores en un espacio de tablas determinado o para todos los contenedores en la base de datos. No se realiza una agregación en las particiones de base de datos. Sin embargo, se puede lograr la agregación mediante consultas de SQL.

La métrica recopilada por esta función se controla a nivel de base de datos mediante el parámetro de configuración `mon_obj_metrics`. Por omisión, la recopilación de métrica está habilitada.

Información devuelta

Tabla 99. Información devuelta para `MON_GET_CONTAINER`

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|---|
| TBSP_NAME | VARCHAR(128) | tablespace_name - Nombre de espacio de tablas |
| TBSP_ID | BIGINT | tablespace_id - Identificación del espacio de tablas |
| CONTAINER_NAME | VARCHAR(256) | container_name - Nombre del contenedor |
| CONTAINER_ID | BIGINT | container_id - Identificación de contenedor |
| MEMBER | SMALLINT | member - Miembro de base de datos |

Tabla 99. Información devuelta para MON_GET_CONTAINER (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|---|
| CONTAINER_TYPE | VARCHAR(16) | container_type - Tipo de contenedor Es un identificador de texto que se basa en las sentencias define de sqlutil.h y es uno de los siguientes: <ul style="list-style-type: none"> • DISK_EXTENT_TAG • DISK_PAGE_TAG • FILE_EXTENT_TAG • FILE_PAGE_TAG • PATH |
| STRIPE_SET | BIGINT | container_stripe_set - Conjunto de bandas |
| DIRECT_READS | BIGINT | direct_reads - Lecturas directas desde base de datos |
| DIRECT_WRITES | BIGINT | direct_writes - Grabaciones directas en base de datos |
| DIRECT_READ_TIME | BIGINT | direct_read_time - Tiempo de lectura directa |
| DIRECT_WRITE_TIME | BIGINT | direct_write_time - Tiempo de grabación directa |
| PAGES_READ | BIGINT | pages_read - Número de páginas leídas |
| PAGES_WRITTEN | BIGINT | pages_written - Número de páginas escritas |
| VECTORED_IOS | BIGINT | vectored_ios - Número de peticiones de E/S de vector |
| PAGES_FROM_VECTORED_IOS | BIGINT | pages_from_vectored_ios - Número total de páginas leídas por E/S de vector |
| BLOCK_IOS | BIGINT | block_ios - Número de peticiones de E/S de bloque |
| PAGES_FROM_BLOCK_IOS | BIGINT | pages_from_block_ios - Número total de páginas leídas por E/S de bloque |
| POOL_READ_TIME | BIGINT | pool_read_time - Tiempo total de lectura física de la agrupación de almacenamientos intermedios |
| POOL_WRITE_TIME | BIGINT | pool_write_time - Tiempo total de grabación física de la agrupación de almacenamientos intermedios |
| TOTAL_PAGES | BIGINT | container_total_pages - Total de páginas en contenedor |
| USABLE_PAGES | BIGINT | container_usable_pages - Páginas utilizables en contenedor |
| ACCESSIBLE | SMALLINT | container_accessible - Accesibilidad del contenedor |
| FS_ID | VARCHAR(22) | fs_id - Número de identificación del sistema de archivos exclusivo |
| FS_TOTAL_SIZE | BIGINT | fs_total_size - Tamaño total de un sistema de archivos |
| FS_USED_SIZE | BIGINT | fs_used_size - Cantidad de espacio utilizada por un sistema de archivos |
| ADDITIONAL_DETAILS | BLOB(100K) | Reservado para una utilización futura. |

MON_GET_EXTENT_MOVEMENT_STATUS - obtener el progreso del traslado de extensiones

La función de tabla MON_GET_EXTENT_MOVEMENT_STATUS devuelve el estado de la operación de traslado de extensiones.

Sintaxis

►►—MON_GET_EXTENT_MOVEMENT_STATUS—(—nombre_espaciotablas—,—miembro—)——►►

El esquema es SYSPROC.

Parámetros de la función de tabla

nombre_espaciotablas

Argumento de entrada de tipo VARCHAR(128) que especifica el espacio de tablas al que se emitirá la consulta. Si el valor del argumento es nulo, la función devuelve información para todos los espacios de tablas.

miembro

Argumento de entrada de tipo INTEGER que especifica un miembro válido en la misma instancia que la base de datos conectada actualmente. Especifique -1 para el miembro de base de datos actual o -2 para todos los miembros de base de datos. Si el valor del argumento es nulo, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE sobre la función MON_GET_EXTENT_MOVEMENT_STATUS.

Ejemplo

Recuperar toda la información sobre el progreso del traslado de extensiones para todos los espacios de tablas:

```
SELECT * FROM TABLE(SYSPROC.MON_GET_EXTENT_MOVEMENT_STATUS('', -1))
```

A continuación se muestra un ejemplo de salida de la consulta:

| TBSP_NAME | TBSP_ID | MEMBER | CURRENT_EXTENT | LAST_EXTENT | NUM_EXTENTS_MOVED |
|--------------|---------|--------|----------------|-------------|-------------------|
| SYSCATSPACE | 0 | 0 | -1 | -1 | -1 |
| TEMPSPACE1 | 1 | 0 | -1 | -1 | -1 |
| USERSPACE1 | 2 | 0 | -1 | -1 | -1 |
| TS1 | 3 | 0 | 1 | 2 | 3 |
| SYSTOOLSPACE | 4 | 0 | -1 | -1 | -1 |

5 registro(s) seleccionado(s).

Datos de salida de la consulta (continuación):

| ... | NUM_EXTENTS_LEFT | TOTAL_MOVE_TIME | ADDITIONAL_DETAILS |
|-----|------------------|-----------------|--------------------|
| ... | -1 | -1 | - |
| ... | -1 | -1 | - |
| ... | -1 | -1 | - |
| ... | 4 | 0 | - |
| ... | -1 | -1 | - |

Información devuelta

Tabla 100. Información devuelta para MON_GET_EXTENT_MOVEMENT_STATUS

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|---|
| TBSP_NAME | VARCHAR(128) | tablespace_name - Nombre de espacio de tablas |

Tabla 100. Información devuelta para MON_GET_EXTENT_MOVEMENT_STATUS (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|--|
| TBSP_ID | BIGINT | tablespace_id - Identificador del espacio de tablas |
| MEMBER | SMALLINT | member - Miembro a partir del cual se ha recopilado esta información |
| CURRENT_EXTENT | INTEGER | current_extent - Extensión que se está trasladando |
| LAST_EXTENT | INTEGER | last_extent - Última extensión movida |
| NUM_EXTENTS_MOVED | INTEGER | num_extents_moved - Número de extensiones trasladadas hasta el momento durante esta operación de traslado de extensiones |
| NUM_EXTENTS_LEFT | INTEGER | num_extents_left - Número de extensiones que quedan por trasladar durante esta operación de traslado de extensiones |
| TOTAL_MOVE_TIME | BIGINT | total_move_time - Tiempo total de traslado para todas las extensiones trasladadas (en milisegundos) |
| ADDITIONAL_DETAILS | BLOB(100K) | Reservado para uso en el futuro |

Función de tabla MON_GET_INDEX - Obtener métrica de índice

La función de tabla MON_GET_INDEX devuelve métricas para uno o varios índices.

Sintaxis

►► MON_GET_INDEX (—esquematabla—, —nombretabla—, —miembro—) ◀◀

El esquema es SYSPROC.

Parámetros de la función de tabla

esquematabla

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de esquema de tabla válido que está en la misma base de datos a la que se está conectado al llamar a esta función. Si el argumento es NULL o una serie vacía, se recuperan métricas correspondientes a los índices de las tablas de todos los esquemas de la base de datos. Si se especifica el argumento, solamente se devuelven métricas correspondientes a los índices de las tablas del esquema especificado.

nombretabla

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de tabla válido que está en la misma base de datos a la que se está conectado al llamar a esta función. Se devuelven métricas correspondientes a todos los índices de la tabla especificada. Si el argumento es nulo o una serie vacía, se recuperan métricas correspondientes a todos los índices de todas las tablas de la base de datos.

miembro

Argumento de entrada de tipo INTEGER que especifica un miembro válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique -1 para el miembro de base de datos actual o -2 para todos los miembros de base de datos. Si se especifica el valor NULL, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE sobre la función MON_GET_INDEX.

Ejemplo

Identificar los índices utilizados con más frecuencia de la tabla DMEXT002.TABLE1 desde la última activación de la base de datos:

```
SELECT VARCHAR(S.INDSCHEMA, 10) AS INDSCHEMA,
       VARCHAR(S.INDNAME, 10) AS INDNAME,
       T.DATA_PARTITION_ID,
       T.MEMBER,
       T.INDEX_SCANS,
       T.INDEX_ONLY_SCANS
FROM TABLE(MON_GET_INDEX('DMEXT002','TABLE1', -2)) as T, SYSCAT.INDEXES AS S
WHERE T.TABSCHEMA = S.TABSCHEMA AND
      T.TABNAME = S.TABNAME AND
      T.IID = S.IID
ORDER BY INDEX_SCANS DESC
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| INDSCHEMA | INDNAME | DATA_PARTITION_ID | MEMBER | INDEX_SCANS | INDEX_ONLY_SCANS |
|-----------|---------|-------------------|--------|-------------|------------------|
| DMEXT002 | INDEX3 | - | - | 0 | 1 |
| DMEXT002 | INDEX4 | - | - | 0 | 1 |
| DMEXT002 | INDEX1 | - | - | 0 | 0 |
| DMEXT002 | INDEX2 | - | - | 0 | 0 |
| DMEXT002 | INDEX5 | - | - | 0 | 0 |
| DMEXT002 | INDEX6 | - | - | 0 | 0 |

6 registro(s) seleccionado(s).

Notas de uso

La función de tabla MON_GET_INDEX devuelve una fila de datos por índice y por miembro de base de datos. Si se utilizan índices particionados, se devuelve una fila para cada partición de índice por miembro de base de datos. No se realiza una agregación en los miembros de base de datos. Sin embargo, se puede lograr la agregación mediante consultas SQL, tal como se muestra en el ejemplo anterior.

Sólo se devolverán métricas correspondientes a los índices de las tablas a las que se ha accedido desde que se activó la base de datos. Todos los contadores representan los datos desde la activación de la base de datos actual. Por ejemplo, el contador *pseudo_empty_pages* es el número de páginas que se han identificado como pseudovacías desde que se activó la base de datos. No es el número actual de páginas pseudovacías del índice.

Las métricas siempre están habilitadas. No es necesario activar ningún conmutador del supervisor del sistema para acceder a las métricas de tabla mediante esta función.

Información devuelta

Tabla 101. Información devuelta para MON_GET_INDEX

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------------|---------------|---|
| TABSCHEMA | VARCHAR(128) | table_schema - Nombre de esquema de tabla |
| TABNAME | VARCHAR(128) | table_name - Nombre de tabla |
| IID | SMALLINT | iid - Identificador de índice |
| MEMBER | SMALLINT | member - Miembro de base de datos |
| DATA_PARTITION_ID | INTEGER | data_partition_id - Identificador de partición de datos. Si el índice no está particionado, se devuelve NULL. |
| NLEAF | BIGINT | nleaf - Número de páginas hoja |
| NLEVELS | SMALLINT | nlevels - Número de niveles de índice |
| INDEX_SCANS | BIGINT | index_scans - Exploraciones de índice |
| INDEX_ONLY_SCANS | BIGINT | index_only_scans - Exploraciones de sólo índice |
| KEY_UPDATES | BIGINT | key_updates - Actualizaciones de claves |
| INCLUDE_COL_UPDATES | BIGINT | include_col_updates - Incluir actualizaciones de columna |
| PSEUDO_DELETES | BIGINT | pseudo_deletes - Pseudosupresiones |
| DEL_KEYS_CLEANED | BIGINT | del_keys_cleaned - Limpieza de claves pseudosuprimidas |
| ROOT_NODE_SPLITS | BIGINT | root_node_splits - Divisiones de nodo raíz |
| INT_NODE_SPLITS | BIGINT | int_node_splits - Divisiones de nodo intermedio |
| BOUNDARY_LEAF_NODE_SPLITS | BIGINT | boundary_leaf_node_splits - Divisiones de nodo de hoja de límite |
| NONBOUNDARY_LEAF_NODE_SPLITS | BIGINT | nonboundary_leaf_node_splits - Divisiones de nodo de hoja que no sea de límite |
| PAGE_ALLOCATIONS | BIGINT | page_allocations - Asignaciones de página |
| PSEUDO_EMPTY_PAGES | BIGINT | pseudo_empty_pages - Páginas pseudovacías |
| EMPTY_PAGES_REUSED | BIGINT | empty_pages_reused - Páginas vacías reutilizadas |
| EMPTY_PAGES_DELETED | BIGINT | empty_pages_deleted - Páginas vacías suprimidas |
| PAGES_MERGED | BIGINT | pages_merged - Páginas fusionadas |
| ADDITIONAL_DETAILS | BLOB(100K) | Reservado para una utilización futura. |

Función de tabla MON_GET_PKG_CACHE_STMT - Obtener métrica de actividad de sentencia de SQL en la antememoria de paquetes

La función de tabla MON_GET_PKG_CACHE_STMT devuelve una vista de punto en el tiempo de sentencias de SQL tanto estáticas como dinámicas en la antememoria del paquete de bases de datos.

Sintaxis

►►—MON_GET_PKG_CACHE_STMT—(—*tipo_sección*—, —————►

►—*id_ejecutable*—, —*argumentos_búsqueda*—, —*miembro*—)—————►

El esquema es SYSPROC.

Parámetros de la función de tabla

tipo_sección

Argumento de entrada ("D" o bien "S") de tipo CHAR(1) que especifica el tipo de la información de la sentencia que se devuelve. Si el argumento es NULL o una serie vacía, se devuelve información para todas las sentencias de SQL. No es sensible a las mayúsculas y minúsculas: "D" es dinámico; "S" es estático.

id_ejecutable

Argumento de entrada de tipo VARCHAR (32) para datos de bits que especifica una sección única de la antememoria del paquete de bases de datos. Si se especifica un valor nulo, se devuelve información para todas las sentencias de SQL. Tenga en cuenta que cuando se especifica el *id_ejecutable*, se pasa por alto el argumento de *tipo_sección*. Por ejemplo, si se especifica un *id_ejecutable* para una sentencia dinámica, esta función de tabla devolverá los detalles de la sentencia dinámica aunque *tipo_sección* esté especificado como estático ("S").

argumentos_búsqueda

Parámetro de entrada de tipo CLOB(1K), actualmente reservado para usos futuros.

miembro

Argumento de entrada de tipo INTEGER que especifica un miembro válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique -1 para el miembro de base de datos actual o -2 para todos los miembros de base de datos. Si se especifica el valor nulo, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE sobre la función MON_GET_PKG_CACHE_STMT.

Ejemplo

Enumerar todas las sentencias de SQL dinámicas de la antememoria del paquete de bases de datos ordenadas por el tiempo medio de CPU.

```
db2 SELECT MEMBER,
        SECTION_TYPE ,
        TOTAL_CPU_TIME/NUM_EXEC_WITH_METRICS as
        AVG_CPU_TIME,EXECUTABLE_ID
FROM TABLE(MON_GET_PKG_CACHE_STMT ( 'D', NULL, NULL, -2)) as T
WHERE T.NUM_EXEC_WITH_METRICS <> 0 ORDER BY AVG_CPU_TIME
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| MEMBER | SECTION_TYPE | AVG_CPU_TIME | EXECUTABLE_ID |
|--------|--------------|--------------|--|
| 0 D | | 754 | x'01000000000000007A0000000000000000000000000000000020020081126171554951791' |
| 0 D | | 2964 | x'0100000000000000790000000000000000000000000000000020020081126171533551120' |
| 0 D | | 5664 | x'01000000000000007C0000000000000000000000000000000020020081126171720728997' |
| 0 D | | 5723 | x'01000000000000007B0000000000000000000000000000000020020081126171657272914' |
| 0 D | | 9762 | x'01000000000000007D0000000000000000000000000000000020020081126172409987719' |

5 registro(s) seleccionado(s).

Nota: Se tarda más tiempo en crear el entorno de compilación y transferir el texto de la sentencia (que puede ocupar hasta 2 MB) a los miembros. Para mejorar el rendimiento al recuperar una lista de todas las sentencias desde la antememoria de paquete, no seleccione las columnas STMT_TEXT y COMP_ENV_DESC.

Con la salida anterior, se puede utilizar *id_ejecutable* para obtener los detalles acerca de la sentencia que va a consumir más tiempo de CPU:

```
db2 SELECT STMT_TEXT FROM TABLE(MON_GET_PKG_CACHE_STMT)
      (null, x'0100000000000007D0000000000000000000000000000000020020081126172409987719', null, -2))
```

```
STMT_TEXT-----
SELECT * FROM EMPLOYEE
```

Como otro ejemplo, supongamos que un usuario llamado Alex tiene una conexión asociada a la carga de trabajo A que tiene establecido COLLECT ACTIVITY METRICS. Otro usuario, Brent, está asociado a la carga de trabajo B que tiene COLLECT ACTIVITY METRICS establecido en NONE. Además, el parámetro de configuración **mon_act_metrics** de base de datos está establecido en NONE. Cuando Brent ejecuta la consulta:

```
SELECT count(*) FROM syscat.tables
```

todas las métricas se devuelven como 0 y el valor de **num_exec_with_metrics** también es 0. Luego, Alex ejecuta la misma sentencia posteriormente, pero esta vez las métricas se recopilan para la ejecución de la sentencia y **num_exec_with_metrics** incrementa. De esta forma, después de que Brent y Alex ejecuten esta sentencia, el resultado de esta consulta:

```
SELECT num_executions, num_exec_with_metrics, SUBSTR(stmt_text,1,50) AS stmt_text
      FROM TABLE (MON_GET_PKG_CACHE_STMT('d', null, null, -1)) AS tf
      WHERE stmt_text LIKE 'SELECT count%'
```

muestra que la sentencia SELECT se ejecutó dos veces y uno de los tiempos de ejecución recopiló las métricas de actividad.

```
NUM_EXECUTIONS NUM_EXEC_WITH_METRICS STMT_TEXT
-----
                2                      1 SELECT count(*) FROM syscat.tables
```

1 registro(s) seleccionado(s).

Notas de uso

La función de tabla MON_GET_PKG_CACHE_STMT devuelve una vista de punto en el tiempo de sentencias de SQL tanto estáticas como dinámicas en la antememoria del paquete de bases de datos. Esto permite examinar la métrica agregada para una sentencia de SQL concreta y así se puede determinar rápidamente las razones de un bajo rendimiento de las consultas. Las métricas devueltas son agregados de las métricas recopiladas durante cada ejecución de la sentencia.

También permite comparar el comportamiento de una sección individual almacenada en la antememoria, con respecto a las demás sentencias, para ayudar a identificar la sección o las sentencias más costosas (en cuanto a costes de ejecución).

Las métricas de actividad que ha notificado esta función aparecen en la antememoria de la base de datos cuando finaliza la ejecución de la actividad.

La recopilación de métricas para la ejecución de cualquier sentencia se controlan mediante la cláusula COLLECT ACTIVITY METRICS en las cargas de trabajo, o

mediante el parámetro de configuración de base de datos **mon_act_metrics** en el nivel de la base de datos. Las métricas sólo se recopilan para las ejecuciones de la sentencia si la conexión que envía la actividad está asociada con una carga de trabajo o una base de datos para la que están habilitadas las métricas de actividad. El elemento **num_exec_with_metrics** devuelto por la función **MON_GET_PKG_CACHE_STMT** indica cuántas ejecuciones de la sentencia han recopilado las métricas y han contribuido a la notificación de métricas agregadas. Si no se han recopilado métricas para ninguna ejecución de la sentencia, el elemento **num_exec_with_metrics** es 0 y todos los valores de las métricas se devuelven como 0.

Información devuelta

Tabla 102. Información devuelta para **MON_GET_PKG_CACHE_STMT**

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------|--------------------------|---|
| MEMBER | SMALLINT | member - Miembro de base de datos |
| SECTION_TYPE | CHAR(1) | section_type - Indicador de tipo de sección. |
| INSERT_TIMESTAMP | TIMESTAMP | insert_timestamp - Indicación de fecha y hora de la inserción de sentencias |
| EXECUTABLE_ID | VARCHAR(32) FOR BIT DATA | executable_id - ID ejecutable. |
| PACKAGE_NAME | VARCHAR(128) | package_name - Nombre de paquete. Esta salida es válida sólo para sentencias de SQL estáticas. Se devuelve un valor NULL si la sentencia es dinámica. |
| PACKAGE_SCHEMA | VARCHAR(128) | package_schema - Esquema de paquete. Esta salida es válida sólo para sentencias de SQL estáticas. Se devuelve un valor NULL si la sentencia es dinámica. |
| PACKAGE_VERSION_ID | VARCHAR(64) | package_version_id - Versión de paquete. Esta salida es válida sólo para sentencias de SQL estáticas. Se devuelve un valor NULL si la sentencia es dinámica o si no se especifica la versión del paquete para la sentencia estática. Se devolverá una serie vacía para las sentencias estáticas si el identificador de versión del paquete no se ha especificado al crear el paquete. |
| SECTION_NUMBER | BIGINT | section_number - Número de sección. Se devuelve un valor NULL si la sentencia es dinámica. |
| EFFECTIVE_ISOLATION | CHAR(2) | effective_isolation - Aislamiento efectivo. Valor de aislamiento en vigor para la sección. Puede ser distinto del que se solicitó inicialmente en el momento de la compilación. |
| NUM_EXECUTIONS | BIGINT | num_executions - Ejecuciones de sentencia |
| NUM_EXEC_WITH_METRICS | BIGINT | num_exec_with_metrics - Número de ejecuciones con métricas recopiladas. |
| PREP_TIME | BIGINT | prep_time - Tiempo de preparación. Tenga en cuenta que PREP_TIME sólo es válido para las sentencias de SQL dinámicas. PREP_TIME se notifica como 0 para sentencias de SQL estáticas. |
| TOTAL_ACT_TIME | BIGINT | total_act_time - Tiempo total de actividad |

Tabla 102. Información devuelta para MON_GET_PKG_CACHE_STMT (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------------|---------------|--|
| TOTAL_ACT_WAIT_TIME | BIGINT | total_act_wait_time - Tiempo de espera total de actividad |
| TOTAL_CPU_TIME | BIGINT | total_cpu_time - Tiempo total de CPU |
| POOL_READ_TIME | BIGINT | pool_read_time - Tiempo total de lectura física de la agrupación de almacenamientos intermedios |
| POOL_WRITE_TIME | BIGINT | pool_write_time - Tiempo total de grabación física de agrupación de almacenamientos intermedios |
| DIRECT_READ_TIME | BIGINT | direct_read_time - Tiempo de lectura directa |
| DIRECT_WRITE_TIME | BIGINT | direct_write_time - Tiempo de grabación directa |
| LOCK_WAIT_TIME | BIGINT | lock_wait_time - Tiempo esperado en bloqueos |
| TOTAL_SECTION_SORT_TIME | BIGINT | total_section_sort_time - Tiempo total de clasificación de sección |
| TOTAL_SECTION_SORT_PROC_TIME | BIGINT | total_section_sort_proc_time - Tiempo total de proceso de clasificación de sección |
| TOTAL_SECTION_SORTS | BIGINT | total_section_sorts - Total de clasificaciones de sección |
| LOCK_ESCALS | BIGINT | lock_escals - Número de escalamientos de bloqueo |
| LOCK_WAITS | BIGINT | lock_waits - Esperas de bloqueo |
| ROWS_MODIFIED | BIGINT | rows_modified - Filas modificadas |
| ROWS_READ | BIGINT | rows_read - Filas leídas |
| ROWS_RETURNED | BIGINT | rows_returned - Filas devueltas |
| DIRECT_READS | BIGINT | direct_reads - Lecturas directas de la base de datos |
| DIRECT_READ_REQS | BIGINT | direct_read_reqs - Peticiones de lectura directa |
| DIRECT_WRITES | BIGINT | direct_writes - Grabaciones directas en la base de datos |
| DIRECT_WRITE_REQS | BIGINT | direct_write_reqs - Peticiones de grabación directa |
| POOL_DATA_L_READS | BIGINT | pool_data_l_reads - Lecturas lógicas de datos de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_L_READS | BIGINT | pool_temp_data_l_reads - Lecturas lógicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_XDA_L_READS | BIGINT | pool_xda_l_reads - Lecturas lógicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_L_READS | BIGINT | pool_temp_xda_l_reads - Lecturas lógicas de datos XDA temporales de agrupación de almacenamientos intermedios |
| POOL_INDEX_L_READS | BIGINT | pool_index_l_reads - Lecturas lógicas de índices de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_L_READS | BIGINT | pool_temp_index_l_reads - Lecturas lógicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_DATA_P_READS | BIGINT | pool_data_p_reads - Lecturas físicas de datos de la agrupación de almacenamientos intermedios |

Tabla 102. Información devuelta para MON_GET_PKG_CACHE_STMT (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------------|---------------|--|
| POOL_TEMP_DATA_P_READS | BIGINT | pool_temp_data_p_reads - Lecturas físicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_XDA_P_READS | BIGINT | pool_xda_p_reads - Lecturas físicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_P_READS | BIGINT | pool_temp_xda_p_reads - Lecturas físicas de datos XDA temporales de agrupación de almacenamientos intermedios |
| POOL_INDEX_P_READS | BIGINT | pool_index_p_reads - Lecturas físicas de índices de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_P_READS | BIGINT | pool_temp_index_p_reads - Lecturas físicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_DATA_WRITES | BIGINT | pool_data_writes - Grabaciones de datos de la agrupación de almacenamientos intermedios |
| POOL_XDA_WRITES | BIGINT | pool_xda_writes - Grabaciones de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_INDEX_WRITES | BIGINT | pool_index_writes - Grabaciones de índices de la agrupación de almacenamientos intermedios |
| TOTAL_SORTS | BIGINT | total_sorts - Número total de clasificaciones |
| POST_THRESHOLD_SORTS | BIGINT | post_threshold_sorts - Clasificaciones de umbral posteriores |
| POST_SHRTHRESHOLD_SORTS | BIGINT | post_shrthreshold_sorts - Clasificaciones de umbral compartidas posteriores |
| SORT_OVERFLOWS | BIGINT | sort_overflows - Desbordamientos de clasificación |
| WLM_QUEUE_TIME_TOTAL | BIGINT | wlm_queue_time_total - Tiempo de cola total del gestor de cargas de trabajo |
| WLM_QUEUE_ASSIGNMENTS_TOTAL | BIGINT | wlm_queue_assignments_total - Total de asignaciones de cola del gestor de cargas de trabajo |
| DEADLOCKS | BIGINT | deadlocks - Puntos muertos detectados |
| FCM_RECV_VOLUME | BIGINT | fcm_recv_volume - Volumen de recepción de FCM |
| FCM_RECVS_TOTAL | BIGINT | fcm_recv_total - Total de recepciones de FCM |
| FCM_SEND_VOLUME | BIGINT | fcm_send_volume - Volumen de envío de FCM |
| FCM_SENDS_TOTAL | BIGINT | fcm_sends_total - Total de envíos de FCM |
| FCM_RECV_WAIT_TIME | BIGINT | fcm_recv_wait_time - Tiempo de espera de recepción de FCM |
| FCM_SEND_WAIT_TIME | BIGINT | fcm_send_wait_time - Tiempo de espera de envío de FCM |
| LOCK_TIMEOUTS | BIGINT | lock_timeouts - Número de tiempos de espera de bloqueo |
| LOG_BUFFER_WAIT_TIME | BIGINT | log_buffer_wait_time - Tiempo de espera de almacenamiento intermedio de anotaciones cronológicas |
| NUM_LOG_BUFFER_FULL | BIGINT | num_log_buffer_full - Número de almacenamientos intermedios de anotaciones cronológicas llenos |

Tabla 102. Información devuelta para MON_GET_PKG_CACHE_STMT (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|----------------------------------|---------------|---|
| LOG_DISK_WAIT_TIME | BIGINT | log_disk_wait_time - Tiempo de espera de disco de anotaciones cronológicas |
| LOG_DISK_WAITS_TOTAL | BIGINT | log_disk_waits_total - Total de esperas de disco de anotaciones cronológicas |
| LAST_METRICS_UPDATE | TIMESTAMP | Indicación de fecha y hora para la última actualización de la métrica. Reservado para su utilización en el futuro. |
| NUM_COORD_EXEC | BIGINT | Reservado para una utilización futura. |
| NUM_COORD_EXEC_WITH_METRICS | BIGINT | Reservado para una utilización futura. |
| VALID | CHAR(1) | valid - Indicador de validez de sección. |
| TOTAL_ROUTINE_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_ROUTINE_INVOCATIONS | BIGINT | Reservado para una utilización futura. |
| ROUTINE_ID | BIGINT | Reservado para una utilización futura. |
| STMT_TYPE_ID | VARCHAR(32) | Reservado para una utilización futura. |
| QUERY_COST_ESTIMATE | BIGINT | Reservado para una utilización futura. |
| STMT_PKG_CACHE_ID | BIGINT | Reservado para una utilización futura. |
| COORD_STMT_EXEC_TIME | BIGINT | Reservado para una utilización futura. |
| STMT_EXEC_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_SECTION_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_SECTION_PROC_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_ROUTINE_NON_SECT_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_ROUTINE_NON_SECT_PROC_TIME | BIGINT | Reservado para una utilización futura. |
| STMT_TEXT | CLOB(2MB) | stmt_text - Texto de sentencia de SQL |
| COMP_ENV_DESC | BLOB(10K) | comp_env_desc - Descriptor de contexto de entorno de compilación. Si es necesario, se puede utilizar la función de tabla COMPILATION_ENV existente para obtener el entorno de compilación detallado de la sentencia específica. |
| ADDITIONAL_DETAILS | BLOB(100K) | Reservado para métricas adicionales en el futuro. |

Función de tabla MON_GET_SERVICE_SUBCLASS - Obtener métrica de subclase de servicio

La función de tabla MON_GET_SERVICE_SUBCLASS devuelve métricas para una o varias subclases de servicio.

Sintaxis

```

▶▶MON_GET_SERVICE_SUBCLASS(—nombre_superclase_servicio—, —————▶
▶—nombre_subclase_servicio—, —miembro—)————▶▶
    
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombre_superclase_servicio

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de superclase de servicio válido de la base de datos conectada actualmente cuando se llama a esta función. Si el argumento es nulo o una serie vacía, se recuperan métricas para todas las superclases de la base de datos.

nombre_subclase_servicio

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de subclase de servicio válido de la base de datos conectada actualmente cuando se llama a esta función. Si el argumento es nulo o una serie vacía, se recuperan métricas para todas las subclases de la base de datos.

miembro

Argumento de entrada de tipo INTEGER que especifica un miembro válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique -1 para el miembro de base de datos actual o -2 para todos los miembros de base de datos. Si se especifica el valor nulo, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE sobre la función MON_GET_SERVICE_SUBCLASS.

Ejemplo

Mostrar el tiempo total de CPU utilizado y el número total de peticiones procesadas para cada clase de servicio, ordenadas por uso de la CPU.

```
SELECT varchar(service_superclass_name,30) as service_superclass,  
       varchar(service_subclass_name,30) as service_subclass,  
       sum(total_cpu_time) as total_cpu,  
       sum(app_rqsts_completed_total) as total_rqsts  
FROM TABLE(MON_GET_SERVICE_SUBCLASS('', '', -2)) AS t  
GROUP BY service_superclass_name, service_subclass_name  
ORDER BY total_cpu desc
```

A continuación se muestra un ejemplo de salida de esta consulta.

| SERVICE_SUPERCLASS | SERVICE_SUBCLASS | ... |
|----------------------------|--------------------|-----|
| ----- | ----- | ... |
| SYSDEFAULTUSERCLASS | SYSDEFAULTSUBCLASS | ... |
| SYSDEFAULTMAINTENANCECLASS | SYSDEFAULTSUBCLASS | ... |
| SYSDEFAULTSYSTEMCLASS | SYSDEFAULTSUBCLASS | ... |

3 registro(s) seleccionado(s).

Salida de la consulta (continuación).

| ... | TOTAL_CPU | TOTAL_RQSTS |
|-----|-----------|-------------|
| ... | 967673 | 100 |
| ... | 0 | 0 |
| ... | 0 | 0 |

Notas de uso

Las métricas devueltas por la función de tabla MON_GET_SERVICE_SUBCLASS representan la suma de todas las métricas correspondientes a las peticiones que se han ejecutado en la subclase de servicio indicada. Las métricas se avanzan a una

clase de servicio cuando llegan a los límites de la unidad de trabajo, y periódicamente durante la ejecución de las peticiones. Por consiguiente, los valores notificados por esta función de tabla reflejan el estado actual del sistema en el momento del avance más reciente. Las métricas son estrictamente de valor creciente. Para determinar el valor de una métrica dada durante un intervalo de tiempo, utilice la función de tabla MON_GET_SERVICE_SUBCLASS para consultar la métrica al principio y al final del intervalo y calcular la diferencia.

Las métricas de petición se controlan mediante la cláusula COLLECT REQUEST METRICS en las superclases de servicio y el parámetro de configuración de base de datos *mon_req_metrics* en el nivel de la base de datos. Las métricas únicamente se recopilan para una petición si la petición se procesa a través de un agente en una subclase de servicio cuya superclase de servicio padre tiene habilitadas las métricas de petición o si la recopilación de métricas de petición está habilitada para toda la base de datos. Por omisión, las métricas de petición están habilitadas en el nivel de la base de datos. Si las métricas de petición están inhabilitadas en el nivel de la base de datos y para una superclase de servicio, las métricas notificadas para cada conexión correlacionada con dicha superclase de servicio dejan de aumentar (o permanecen en 0 si las métricas de petición estaban inhabilitadas en el momento de activarse la base de datos).

La función de tabla MON_GET_SERVICE_SUBCLASS devuelve una fila de datos por subclase de servicio y por miembro. No se realiza ninguna agregación entre clases de servicio (de un miembro) ni entre miembros (para una o varias clases de servicio). Sin embargo, se puede lograr la agregación mediante consultas de SQL, tal como se muestra en el ejemplo. Los parámetros de entrada tienen el efecto de sumarse (operación AND). Por consiguiente, si se especifican parámetros de entrada conflictivos (por ejemplo, un nombre de superclase SUPA y un nombre de subclase SUBB que no es una subclase de SUPA), no se devuelve ninguna fila.

Consejo: una petición podría ejecutarse en más de una subclase de servicio. Por ejemplo, podría producirse esta situación si se correlacionara una petición de una subclase de servicio con otra mediante un umbral de WLM (Workload Manager, gestor de cargas de trabajo) con una acción REMAP ACTIVITY. Aunque las métricas de tiempo transcurrido se actualizan para cada subclase de servicio bajo la que se ejecuta la petición, los contadores de petición aumentan para la subclase de servicio en la que se completó la petición. Por consiguiente, no debería analizar los promedios de tiempos de petición para una única subclase. Todas las subclases con las que se puede correlacionar una actividad se deben analizar juntas. Por ejemplo, si existe un umbral que puede correlacionar actividades de la subclase de servicio A con actividades de la subclase de servicio B, cuando vaya a calcular los promedios de las solicitudes, debería sumar los contadores y las métricas de las subclases de servicio A y B, y calcular los promedios utilizando las sumas.

Información devuelta

Tabla 103. Información devuelta para MON_GET_SERVICE_SUBCLASS

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|--|
| SERVICE_SUPERCLASS_NAME | VARCHAR(128) | service_superclass_name - Nombre de superclase de servicio |
| SERVICE_SUBCLASS_NAME | VARCHAR(128) | service_subclass_name - Nombre de subclase de servicio |
| SERVICE_CLASS_ID | INTEGER | service_class_id - ID de clase de servicio |
| MEMBER | SMALLINT | member - Miembro de base de datos |

Tabla 103. Información devuelta para MON_GET_SERVICE_SUBCLASS (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|---|
| ACT_ABORTED_TOTAL | BIGINT | act_aborted_total - Total de actividades abortadas |
| ACT_COMPLETED_TOTAL | BIGINT | act_completed_total - Total de actividades completadas |
| ACT_REJECTED_TOTAL | BIGINT | act_rejected_total - Total de actividades rechazadas |
| AGENT_WAIT_TIME | BIGINT | agent_wait_time - Tiempo de espera del agente |
| AGENT_WAITS_TOTAL | BIGINT | agent_waits_total - Total de esperas del agente |
| POOL_DATA_L_READS | BIGINT | pool_data_l_reads - Lecturas lógicas de datos de la agrupación de almacenamientos intermedios |
| POOL_INDEX_L_READS | BIGINT | pool_index_l_reads - Lecturas lógicas de índices de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_L_READS | BIGINT | pool_temp_data_l_reads - Lecturas lógicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_L_READS | BIGINT | pool_temp_index_l_reads - Lecturas lógicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_L_READS | BIGINT | pool_temp_xda_l_reads - Lecturas lógicas de datos XDA temporales de agrupación de almacenamientos intermedios |
| POOL_XDA_L_READS | BIGINT | pool_temp_xda_l_reads - Lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |
| POOL_DATA_P_READS | BIGINT | pool_data_p_reads - Lecturas físicas de datos de la agrupación de almacenamientos intermedios |
| POOL_INDEX_P_READS | BIGINT | pool_index_p_reads - Lecturas físicas de índices de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_P_READS | BIGINT | pool_temp_data_p_reads - Lecturas físicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_P_READS | BIGINT | pool_temp_index_p_reads - Lecturas físicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_P_READS | BIGINT | pool_temp_xda_p_reads - Lecturas físicas de datos XDA temporales de agrupación de almacenamientos intermedios |
| POOL_XDA_P_READS | BIGINT | pool_xda_p_reads - Lecturas físicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_DATA_WRITES | BIGINT | pool_data_writes - Grabaciones de datos de la agrupación de almacenamientos intermedios |

Tabla 103. Información devuelta para MON_GET_SERVICE_SUBCLASS (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------|---------------|---|
| POOL_INDEX_WRITES | BIGINT | pool_index_writes - Grabaciones de índices de la agrupación de almacenamientos intermedios |
| POOL_XDA_WRITES | BIGINT | pool_xda_writes - Grabaciones de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_READ_TIME | BIGINT | pool_read_time - Tiempo total de lectura física de la agrupación de almacenamientos intermedios |
| POOL_WRITE_TIME | BIGINT | pool_write_time - Tiempo total de grabación física de agrupación de almacenamientos intermedios |
| CLIENT_IDLE_WAIT_TIME | BIGINT | client_idle_wait_time - Tiempo de espera de inactividad del cliente |
| DEADLOCKS | BIGINT | deadlocks - Puntos muertos detectados |
| DIRECT_READS | BIGINT | direct_reads - Lecturas directas de la base de datos |
| DIRECT_READ_TIME | BIGINT | direct_read_time - Tiempo de lectura directa |
| DIRECT_WRITES | BIGINT | direct_writes - Grabaciones directas en la base de datos |
| DIRECT_WRITE_TIME | BIGINT | direct_write_time - Tiempo de grabación directa |
| DIRECT_READ_REQS | BIGINT | direct_read_reqs - Peticiones de lectura directa |
| DIRECT_WRITE_REQS | BIGINT | direct_write_reqs - Peticiones de grabación directa |
| FCM_RECV_VOLUME | BIGINT | fcm_recv_volume - Volumen de recepción de FCM |
| FCM_RECVS_TOTAL | BIGINT | fcm_recvs_total - Total de recepciones de FCM |
| FCM_SEND_VOLUME | BIGINT | fcm_send_volume - Volumen de envío de FCM |
| FCM_SENDS_TOTAL | BIGINT | fcm_sends_total - Total de envíos de FCM |
| FCM_RECV_WAIT_TIME | BIGINT | fcm_recv_wait_time - Tiempo de espera de recepción de FCM |
| FCM_SEND_WAIT_TIME | BIGINT | fcm_send_wait_time - Tiempo de espera de envío de FCM |
| IPC_RECV_VOLUME | BIGINT | ipc_recv_volume - Volumen de recepción de comunicación entre procesos |
| IPC_RECV_WAIT_TIME | BIGINT | ipc_recv_wait_time - Tiempo de espera de recepción de comunicación entre procesos |
| IPC_RECVS_TOTAL | BIGINT | ipc_recvs_total - Total de recepciones de comunicación entre procesos |
| IPC_SEND_VOLUME | BIGINT | ipc_send_volume - Volumen de envío de comunicación entre procesos |
| IPC_SEND_WAIT_TIME | BIGINT | ipc_send_wait_time - Tiempo de espera de envío de comunicación entre procesos |

Tabla 103. Información devuelta para MON_GET_SERVICE_SUBCLASS (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------|---------------|--|
| IPC_SENDS_TOTAL | BIGINT | ipc_sends_total - Total de envíos de comunicación entre procesos |
| LOCK_ESCALAS | BIGINT | lock_escals - Número de escalamientos de bloqueo |
| LOCK_TIMEOUTS | BIGINT | lock_timeouts - Número de tiempos de espera de bloqueo |
| LOCK_WAIT_TIME | BIGINT | lock_wait_time - Tiempo esperado en bloqueos |
| LOCK_WAITS | BIGINT | lock_waits - Esperas de bloqueo |
| LOG_BUFFER_WAIT_TIME | BIGINT | log_buffer_wait_time - Tiempo de espera de almacenamiento intermedio de anotaciones cronológicas |
| NUM_LOG_BUFFER_FULL | BIGINT | num_log_buffer_full - Número de almacenamientos intermedios de anotaciones cronológicas llenos |
| LOG_DISK_WAIT_TIME | BIGINT | log_disk_wait_time - Tiempo de espera de disco de anotaciones cronológicas |
| LOG_DISK_WAITS_TOTAL | BIGINT | log_disk_waits_total - Total de esperas de disco de anotaciones cronológicas |
| RQSTS_COMPLETED_TOTAL | BIGINT | rqsts_completed_total - Total de peticiones completadas |
| ROWS_MODIFIED | BIGINT | rows_modified - Filas modificadas |
| ROWS_READ | BIGINT | rows_read - Filas leídas |
| ROWS_RETURNED | BIGINT | rows_returned - Filas devueltas |
| TCPIP_RECV_VOLUME | BIGINT | tcpip_recv_volume - Volumen recibido TCP/IP |
| TCPIP_SEND_VOLUME | BIGINT | tcpip_send_volume - Volumen de envío TCP/IP |
| TCPIP_RECV_WAIT_TIME | BIGINT | tcpip_recv_wait_time - Tiempo de espera de recepción TCP/IP |
| TCPIP_RECVS_TOTAL | BIGINT | tcpip_recvs_total - Total de recepciones TCP/IP |
| TCPIP_SEND_WAIT_TIME | BIGINT | tcpip_send_wait_time - Tiempo de espera de envío TCP/IP |
| TCPIP_SENDS_TOTAL | BIGINT | tcpip_sends_total - Total de envíos TCP/IP |
| TOTAL_APP_RQST_TIME | BIGINT | total_app_rqst_time - Total de tiempo de petición de aplicaciones |
| TOTAL_RQST_TIME | BIGINT | total_rqst_time - Tiempo total de petición |
| WLM_QUEUE_TIME_TOTAL | BIGINT | wlm_queue_time_total - Tiempo de cola total del gestor de cargas de trabajo |
| _TOTAL | BIGINT | wlm_queue_assignments_total - Total de asignaciones de cola del gestor de cargas de trabajo |
| TOTAL_RQST_MAPPED_IN | BIGINT | total_rqst_mapped_in - Total de peticiones correlacionadas incluidas |

Tabla 103. Información devuelta para MON_GET_SERVICE_SUBCLASS (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|----------------------------------|---------------|--|
| TOTAL_RQST_MAPPED_OUT | BIGINT | total_rqst_mapped_out - Total de peticiones correlacionadas excluidas |
| TOTAL_CPU_TIME | BIGINT | total_cpu_time - Tiempo total de CPU |
| TOTAL_WAIT_TIME | BIGINT | total_wait_time - Tiempo total de espera |
| APP_RQSTS_COMPLETED_TOTAL | BIGINT | app_rqsts_completed_total - Total de peticiones de aplicación completadas |
| TOTAL_SECTION_SORT_TIME | BIGINT | total_section_sort_time - Tiempo total de clasificación de sección |
| TOTAL_SECTION_SORT_PROC_TIME | BIGINT | total_section_sort_proc_time - Tiempo total de proceso de clasificación de sección |
| TOTAL_SECTION_SORTS | BIGINT | total_section_sorts - Total de clasificaciones de sección |
| TOTAL_SORTS | BIGINT | total_sorts - Número total de clasificaciones |
| POST_THRESHOLD_SORTS | BIGINT | post_threshold_sorts - Clasificaciones de umbral posteriores |
| POST_SHRTHRESHOLD_SORTS | BIGINT | post_shrthreshold_sorts - Clasificaciones de umbral compartidas posteriores |
| SORT_OVERFLOWS | BIGINT | sort_overflows - Desbordamientos de clasificación |
| TOTAL_COMPILE_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_COMPILE_PROC_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_COMPILATIONS | BIGINT | Reservado para una utilización futura. |
| TOTAL_IMPLICIT_COMPILE_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_IMPLICIT_COMPILE_PROC_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_IMPLICIT_COMPILATIONS | BIGINT | Reservado para una utilización futura. |
| TOTAL_SECTION_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_SECTION_PROC_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_APP_SECTION_EXECUTIONS | BIGINT | Reservado para una utilización futura. |
| TOTAL_ACT_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_ACT_WAIT_TIME | BIGINT | Reservado para una utilización futura. |
| ACT_RQSTS_TOTAL | BIGINT | Reservado para una utilización futura. |
| TOTAL_ROUTINE_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_ROUTINE_INVOCATIONS | BIGINT | Reservado para una utilización futura. |
| TOTAL_COMMIT_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_COMMIT_PROC_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_APP_COMMITS | BIGINT | Reservado para una utilización futura. |
| INT_COMMITS | BIGINT | Reservado para una utilización futura. |
| TOTAL_ROLLBACK_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_ROLLBACK_PROC_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_APP_ROLLBACKS | BIGINT | Reservado para una utilización futura. |
| INT_ROLLBACKS | BIGINT | Reservado para una utilización futura. |

Tabla 103. Información devuelta para MON_GET_SERVICE_SUBCLASS (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------------|---------------|---|
| TOTAL_RUNSTATS_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_RUNSTATS_PROC_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_RUNSTATS | BIGINT | Reservado para una utilización futura. |
| TOTAL_REORG_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_REORG_PROC_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_REORGS | BIGINT | Reservado para una utilización futura. |
| TOTAL_LOAD_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_LOAD_PROC_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_LOADS | BIGINT | Reservado para una utilización futura. |
| CAT_CACHE_INSERTS | BIGINT | Reservado para una utilización futura. |
| CAT_CACHE_LOOKUPS | BIGINT | Reservado para una utilización futura. |
| PKG_CACHE_INSERTS | BIGINT | Reservado para una utilización futura. |
| PKG_CACHE_LOOKUPS | BIGINT | Reservado para una utilización futura. |
| THRESH_VIOLATIONS | BIGINT | Reservado para una utilización futura. |
| NUM_LW_THRESH_EXCEEDED | BIGINT | Reservado para una utilización futura. |
| ADDITIONAL_DETAILS | BLOB(100K) | Reservado para uso en el futuro |

Función de tabla MON_GET_SERVICE_SUBCLASS_DETAILS - Obtener métrica detallada de subclase de servicio

La función de tabla MON_GET_SERVICE_SUBCLASS_DETAILS devuelve métricas detalladas para una o varias subclases de servicio.

Sintaxis

```

▶▶—MON_GET_SERVICE_SUBCLASS_DETAILS—(—nombre_superclase_servicio—, —————▶
▶—nombre_subclase_servicio—, —miembro—) —————▶▶

```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombre_superclase_servicio

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de superclase de servicio válido de la base de datos conectada actualmente cuando se llama a esta función. Si el argumento es nulo o una serie vacía, se recuperan métricas para todas las superclases de la base de datos.

nombre_subclase_servicio

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de subclase de servicio válido de la base de datos conectada actualmente cuando se llama a esta función. Si el argumento es nulo o una serie vacía, se recuperan métricas para todas las subclases de la base de datos.

miembro

Argumento de entrada de tipo INTEGER que especifica un miembro válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique -1 para el miembro de base de datos actual o -2 para todos los miembros de base de datos. Si se especifica el valor nulo, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE sobre la función MON_GET_SERVICE_SUBCLASS_DETAILS.

Ejemplo

Mostrar el tiempo total de CPU utilizado y el número total de peticiones procesadas para cada superclase de servicio, ordenadas por uso de la CPU en formato relacional (mediante XMLTABLE).

```
SELECT varchar(scmetrics.service_superclass_name,30) as service_superclass,
       sum(detmetrics.total_cpu_time) as total_cpu,
       sum(detmetrics.app_rqsts_completed_total) as total_rqsts
FROM TABLE(MON_GET_SERVICE_SUBCLASS_DETAILS('','",-2)) AS SCMETRICS,
XMLTABLE (XMLNAMESPACES( DEFAULT 'http://www.ibm.com/xmlns/prod/db2/mon'),
         '$detmetric/db2_service_subclass'
         PASSING XMLPARSE(DOCUMENT SCMETRICS.DETAILS)
         as "detmetric"
COLUMNS "TOTAL_CPU_TIME" INTEGER PATH 'system_metrics/total_cpu_time',
         "APP_RQSTS_COMPLETED_TOTAL" INTEGER PATH
         'system_metrics/app_rqsts_completed_total')
AS DETMETRICS
GROUP BY service_superclass_name
ORDER BY total_cpu desc
```

A continuación se muestra un ejemplo de salida de esta consulta.

| SERVICE_SUPERCLASS | TOTAL_CPU | TOTAL_RQSTS |
|----------------------------|-----------|-------------|
| SYSDEFAULTUSERCLASS | 2428188 | 26 |
| SYSDEFAULTMAINTENANCECLASS | 0 | 0 |
| SYSDEFAULTSYSTEMCLASS | 0 | 0 |

3 registro(s) seleccionado(s).

Notas de uso

Las métricas devueltas por la función de tabla MON_GET_SERVICE_SUBCLASS_DETAILS representan la suma de todas las métricas correspondientes a las peticiones que se han ejecutado en la subclase de servicio indicada. Esta función es parecida a la función de tabla MON_GET_SERVICE_SUBCLASS:

- La función de tabla MON_GET_SERVICE_SUBCLASS devuelve las métricas utilizadas con más frecuencia en un formato basado en columnas, y es el método más eficiente, desde el punto de vista del rendimiento, de recuperar las métricas.
- La función de tabla MON_GET_SERVICE_SUBCLASS_DETAILS devuelve todo el conjunto de métricas disponibles en formato de documento XML, lo que proporciona la máxima flexibilidad para dar formato a la salida. La salida basada en XML puede analizarse directamente mediante un analizador XML, o puede convertirse a formato relacional mediante la función XMLTABLE (véase el ejemplo).

Las métricas se avanzan a una clase de servicio cuando llegan a los límites de la unidad de trabajo, y periódicamente durante la ejecución de las peticiones. Por consiguiente, los valores notificados por esta función de tabla reflejan el estado actual del sistema en el momento del avance más reciente. Las métricas son estrictamente de valor creciente. Para determinar el valor de una métrica dada durante un intervalo de tiempo, utilice la función de tabla `MON_GET_SERVICE_SUBCLASS_DETAILS` para consultar la métrica al principio y al final del intervalo y calcular la diferencia.

Las métricas de petición se controlan mediante la cláusula `COLLECT REQUEST METRICS` en las superclases de servicio y el parámetro de configuración de base de datos `mon_req_metrics` en el nivel de la base de datos. Las métricas únicamente se recopilan para una petición si la petición se procesa a través de un agente en una subclase de servicio cuya superclase de servicio padre tiene habilitadas las métricas de petición o si la recopilación de métricas de petición está habilitada para toda la base de datos. Por omisión, las métricas de petición están habilitadas en el nivel de la base de datos. Si las métricas de petición están inhabilitadas en el nivel de la base de datos para una superclase de servicio, las métricas notificadas para cada conexión correlacionada con dicha superclase de servicio dejan de aumentar (o permanecen en 0 si las métricas de petición estaban inhabilitadas en el momento de activarse la base de datos).

La función de tabla `MON_GET_SERVICE_SUBCLASS_DETAILS` devuelve una fila de datos por subclase de servicio y por miembro. No se realiza ninguna agregación entre clases de servicio (de un miembro) ni entre miembros (para una o varias clases de servicio). Sin embargo, se puede lograr la agregación mediante consultas de SQL (véase el ejemplo). Los parámetros de entrada tienen el efecto de sumarse (operación AND). Por consiguiente, si se especifican parámetros de entrada conflictivos (por ejemplo, un nombre de superclase SUPA y un nombre de subclase SUBB que no es una subclase de SUPA), no se devuelve ninguna fila.

Consejo: una petición podría ejecutarse en más de una subclase de servicio. Por ejemplo, podría producirse esta situación si se correlacionara una petición de una subclase de servicio con otra mediante un umbral de WLM (Workload Manager, gestor de cargas de trabajo) con una acción `REMAP ACTIVITY`. Aunque las métricas de tiempo transcurrido se actualizan para cada subclase de servicio bajo la que se ejecuta la petición, los contadores de petición aumentan para la subclase de servicio en la que se completó la petición. Por consiguiente, no debería analizar los promedios de tiempos de petición para una única subclase. Todas las subclases con las que se puede correlacionar una actividad se deben analizar juntas. Por ejemplo, si existe un umbral que puede correlacionar actividades de la subclase de servicio A con actividades de la subclase de servicio B, cuando vaya a calcular los promedios de las solicitudes, debería sumar los contadores y las métricas de las subclases de servicio A y B, y calcular los promedios utilizando las sumas.

El esquema para el documento XML devuelto en la columna `DETAILS` está disponible en el archivo `sql11b/misc/DB2MonRoutines.xsd`. Para obtener más información, consulte el archivo `sql11b/misc/DB2MonCommon.xsd`.

Información devuelta

Tabla 104. Información devuelta para `MON_GET_SERVICE_SUBCLASS_DETAILS`

| Nombre de columna | Tipo de datos | Descripción |
|--------------------------------------|---------------------------|---|
| <code>SERVICE_SUPERCLASS_NAME</code> | <code>VARCHAR(128)</code> | <code>service_superclass_name</code> - Nombre de superclase de servicio |

Tabla 104. Información devuelta para `MON_GET_SERVICE_SUBCLASS_DETAILS` (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|------------------------------------|---------------------------|--|
| <code>SERVICE_SUBCLASS_NAME</code> | <code>VARCHAR(128)</code> | <code>service_subclass_name</code> - Nombre de subclase de servicio |
| <code>SERVICE_CLASS_ID</code> | <code>INTEGER</code> | <code>service_class_id</code> - ID de clase de servicio |
| <code>MEMBER</code> | <code>SMALLINT</code> | <code>member</code> - Miembro de base de datos |
| <code>DETAILS</code> | <code>BLOB(1M)</code> | Documento XML que contiene las métricas detalladas de la clase de servicio. Consulte la Tabla 105 para ver una descripción de los elementos de este documento. |

El ejemplo siguiente muestra la estructura del documento XML que se devuelve en la columna `DETAILS`.

```
<db2_service_subclass xmlns="http://www.ibm.com/xmlns/prod/db2/mon" release="90700000">
  <service_superclass_name>SYSDEFAULTSYSTEMCLASS</service_superclass_name>
  <service_subclass_name>SYSDEFAULTSUBCLASS</service_subclass_name>
  <service_subclass_id>11</service_subclass_id>
  <member>0</member>
  <system_metrics release="90700000">
    <act_aborted_total>5</act_aborted_total>
    ...
    <wlm_queue_assignments_total>3</wlm_queue_assignments_total>
  </system_metrics>
</db2_service_subclass>
```

Para ver el esquema completo, consulte `sqllib/misc/DB2MonRoutines.xsd`.

Tabla 105. Métricas detalladas devueltas para `MON_GET_SERVICE_SUBCLASS_DETAILS`

| Nombre de elemento | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--|------------------------------------|--|
| <code>service_superclass_name</code> | <code>xs:string(128)</code> | <code>service_superclass_name</code> - Nombre de superclase de servicio |
| <code>service_subclass_name</code> | <code>xs:string(128)</code> | <code>service_subclass_name</code> - Nombre de subclase de servicio |
| <code>service_class_id</code> | <code>xs:nonNegativeInteger</code> | <code>service_class_id</code> - ID de clase de servicio |
| <code>member</code> | <code>xs:nonNegativeInteger</code> | <code>member</code> - Miembro de base de datos |
| <code>act_aborted_total</code> | <code>xs:nonNegativeInteger</code> | <code>act_aborted_total</code> - Total de actividades abortadas |
| <code>act_completed_total</code> | <code>xs:nonNegativeInteger</code> | <code>act_completed_total</code> - Total de actividades completadas |
| <code>act_rejected_total</code> | <code>xs:nonNegativeInteger</code> | <code>act_rejected_total</code> - Total de actividades rechazadas |
| <code>agent_wait_time</code> | <code>xs:nonNegativeInteger</code> | <code>agent_wait_time</code> - Tiempo de espera del agente |
| <code>agent_waits_total</code> | <code>xs:nonNegativeInteger</code> | <code>agent_waits_total</code> - Total de esperas del agente |
| <code>app_rqsts_completed_total</code> | <code>xs:nonNegativeInteger</code> | <code>app_rqsts_completed_total</code> - Total de peticiones de aplicación completadas |
| <code>audit_events_total</code> | <code>xs:nonNegativeInteger</code> | <code>audit_events_total</code> - Total de sucesos de auditoría |
| <code>audit_subsystem_wait_time</code> | <code>xs:nonNegativeInteger</code> | <code>audit_subsystem_wait_time</code> - Tiempo de espera del subsistema de auditoría |
| <code>audit_subsystem_waits_total</code> | <code>xs:nonNegativeInteger</code> | <code>audit_subsystem_waits_total</code> - Total de esperas del subsistema de auditoría |
| <code>audit_file_write_wait_time</code> | <code>xs:nonNegativeInteger</code> | <code>audit_file_write_wait_time</code> - Tiempo de espera de grabación del archivo de auditoría |
| <code>audit_file_writes_total</code> | <code>xs:nonNegativeInteger</code> | <code>audit_file_writes_total</code> - Total de archivos de auditoría grabados |
| <code>client_idle_wait_time</code> | <code>xs:nonNegativeInteger</code> | <code>client_idle_wait_time</code> - Tiempo de espera de inactividad del cliente |

Tabla 105. Métricas detalladas devueltas para MON_GET_SERVICE_SUBCLASS_DETAILS (continuación)

| Nombre de elemento | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|----------------------------|-----------------------|--|
| deadlocks | xs:nonNegativeInteger | deadlocks - Puntos muertos detectados |
| diaglog_writes_total | xs:nonNegativeInteger | diaglog_writes_total - Total de grabaciones de anotaciones cronológicas de diagnóstico |
| diaglog_write_wait_time | xs:nonNegativeInteger | diaglog_write_wait_time - Tiempo de grabación de anotaciones cronológicas de diagnóstico |
| direct_read_time | xs:nonNegativeInteger | direct_read_time - Tiempo de lectura directa |
| direct_write_time | xs:nonNegativeInteger | direct_write_time - Tiempo de grabación directa |
| direct_read_reqs | xs:nonNegativeInteger | direct_read_reqs - Peticiones de lectura directa |
| direct_reads | xs:nonNegativeInteger | direct_reads - Lecturas directas de la base de datos |
| direct_write_reqs | xs:nonNegativeInteger | direct_write_reqs - Peticiones de grabación directa |
| direct_writes | xs:nonNegativeInteger | direct_writes - Grabaciones directas en la base de datos |
| fcm_recv_volume | xs:nonNegativeInteger | fcm_recv_volume - Volumen de recepción de FCM |
| fcm_recv_wait_time | xs:nonNegativeInteger | fcm_recv_wait_time - Tiempo de espera de recepción de FCM |
| fcm_recvs_total | xs:nonNegativeInteger | fcm_recvs_total - Total de recepciones de FCM |
| fcm_message_recv_volume | xs:nonNegativeInteger | fcm_message_recv_volume - Volumen de recepción de mensaje de FCM |
| fcm_message_recvs_total | xs:nonNegativeInteger | fcm_message_recvs_total - Total de recepciones de mensaje de FCM |
| fcm_message_recv_wait_time | xs:nonNegativeInteger | fcm_message_recv_wait_time - Tiempo de espera de recepción de mensaje de FCM |
| fcm_message_send_volume | xs:nonNegativeInteger | fcm_message_send_volume - Volumen de envío de mensaje de FCM |
| fcm_message_send_wait_time | xs:nonNegativeInteger | fcm_message_send_wait_time - Tiempo de espera de envío de mensaje de FCM |
| fcm_message_sends_total | xs:nonNegativeInteger | fcm_message_sends_total - Total de envíos de mensaje de FCM |
| fcm_send_volume | xs:nonNegativeInteger | fcm_send_volume - Volumen de envío de FCM |
| fcm_send_wait_time | xs:nonNegativeInteger | fcm_send_wait_time - Tiempo de espera de envío de FCM |
| fcm_sends_total | xs:nonNegativeInteger | fcm_sends_total - Total de envíos de FCM |
| fcm_tq_recv_wait_time | xs:nonNegativeInteger | fcm_tq_recv_wait_time - Tiempo de espera de recepción de cola de tabla de FCM |
| fcm_tq_send_wait_time | xs:nonNegativeInteger | fcm_tq_send_wait_time - Tiempo de espera de envío de cola de tabla de FCM |
| fcm_tq_recv_volume | xs:nonNegativeInteger | fcm_tq_recv_volume - Volumen de recepción de cola de tabla de FCM |
| fcm_tq_recvs_total | xs:nonNegativeInteger | fcm_tq_recvs_total - Total de recepciones de cola de tabla de FCM |
| fcm_tq_send_volume | xs:nonNegativeInteger | fcm_tq_send_volume - Volumen de envíos de cola de tabla de FCM |
| fcm_tq_sends_total | xs:nonNegativeInteger | fcm_tq_sends_total - Total de envíos de cola de tabla de FCM |
| tq_tot_send_spills | xs:nonNegativeInteger | tq_tot_send_spills - Total de almacenamientos intermedios de cola de tabla desbordados |

Tabla 105. Métricas detalladas devueltas para MON_GET_SERVICE_SUBCLASS_DETAILS (continuación)

| Nombre de elemento | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|-----------------------|--|
| ipc_rcv_volume | xs:nonNegativeInteger | ipc_rcv_volume - Volumen de recepción de comunicación entre procesos |
| ipc_rcv_wait_time | xs:nonNegativeInteger | ipc_rcv_wait_time - Tiempo de espera de recepción de comunicación entre procesos |
| ipc_rcvs_total | xs:nonNegativeInteger | ipc_rcvs_total - Total de recepciones de comunicación entre procesos |
| ipc_send_volume | xs:nonNegativeInteger | ipc_send_volume - Volumen de envío de comunicación entre procesos |
| ipc_send_wait_time | xs:nonNegativeInteger | ipc_send_wait_time - Tiempo de espera de envío de comunicación entre procesos |
| ipc_sends_total | xs:nonNegativeInteger | ipc_sends_total - Total de envíos de comunicación entre procesos |
| lock_escals | xs:nonNegativeInteger | lock_escals - Número de escalamientos de bloqueo |
| lock_timeouts | xs:nonNegativeInteger | lock_timeouts - Número de tiempos de espera de bloqueo |
| lock_wait_time | xs:nonNegativeInteger | lock_wait_time - Tiempo esperado en bloqueos |
| lock_waits | xs:nonNegativeInteger | lock_waits - Esperas de bloqueo |
| log_buffer_wait_time | xs:nonNegativeInteger | log_buffer_wait_time - Tiempo de espera de almacenamiento intermedio de anotaciones cronológicas |
| log_disk_wait_time | xs:nonNegativeInteger | log_disk_wait_time - Tiempo de espera de disco de anotaciones cronológicas |
| log_disk_waits_total | xs:nonNegativeInteger | log_disk_waits_total - Total de esperas de disco de anotaciones cronológicas |
| pool_data_l_reads | xs:nonNegativeInteger | pool_data_l_reads - Lecturas lógicas de datos de la agrupación de almacenamientos intermedios |
| pool_data_p_reads | xs:nonNegativeInteger | pool_data_p_reads - Lecturas físicas de datos de la agrupación de almacenamientos intermedios |
| pool_data_writes | xs:nonNegativeInteger | pool_data_writes - Grabaciones de datos de la agrupación de almacenamientos intermedios |
| pool_index_l_reads | xs:nonNegativeInteger | pool_index_l_reads - Lecturas lógicas de índices de la agrupación de almacenamientos intermedios |
| pool_index_p_reads | xs:nonNegativeInteger | pool_index_p_reads - Lecturas físicas de índices de la agrupación de almacenamientos intermedios |
| pool_index_writes | xs:nonNegativeInteger | pool_index_writes - Grabaciones de índices de la agrupación de almacenamientos intermedios |
| pool_read_time | xs:nonNegativeInteger | pool_read_time - Tiempo total de lectura física de la agrupación de almacenamientos intermedios |
| pool_temp_data_l_reads | xs:nonNegativeInteger | pool_temp_data_l_reads - Lecturas lógicas de datos temporales de la agrupación de almacenamientos intermedios |
| pool_temp_data_p_reads | xs:nonNegativeInteger | pool_temp_data_p_reads - Lecturas físicas de datos temporales de la agrupación de almacenamientos intermedios |
| pool_temp_index_l_reads | xs:nonNegativeInteger | pool_temp_index_l_reads - Lecturas lógicas de índices temporales de la agrupación de almacenamientos intermedios |

Tabla 105. Métricas detalladas devueltas para MON_GET_SERVICE_SUBCLASS_DETAILS (continuación)

| Nombre de elemento | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------------|-----------------------|---|
| pool_temp_index_p_reads | xs:nonNegativeInteger | pool_temp_index_p_reads - Lecturas físicas de índices temporales de la agrupación de almacenamientos intermedios |
| pool_temp_xda_l_reads | xs:nonNegativeInteger | pool_temp_xda_l_reads - Lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |
| pool_temp_xda_p_reads | xs:nonNegativeInteger | pool_temp_xda_p_reads - Lecturas físicas de datos XDA temporales de agrupación de almacenamientos intermedios |
| pool_write_time | xs:nonNegativeInteger | pool_write_time - Tiempo total de grabación física de agrupación de almacenamientos intermedios |
| pool_xda_l_reads | xs:nonNegativeInteger | pool_xda_l_reads - Lecturas lógicas de datos de XDA de la agrupación de almacenamientos intermedios |
| pool_xda_p_reads | xs:nonNegativeInteger | pool_xda_p_reads - Lecturas físicas de datos de XDA de la agrupación de almacenamientos intermedios |
| pool_xda_writes | xs:nonNegativeInteger | pool_xda_writes - Grabaciones de datos de XDA de la agrupación de almacenamientos intermedios |
| num_log_buffer_full | xs:nonNegativeInteger | num_log_buffer_full - Número de almacenamientos intermedios de anotaciones cronológicas llenos |
| rqsts_completed_total | xs:nonNegativeInteger | rqsts_completed_total - Total de peticiones completadas |
| total_rqst_mapped_in | xs:nonNegativeInteger | total_rqst_mapped_in - Total de peticiones correlacionadas incluidas |
| total_rqst_mapped_out | xs:nonNegativeInteger | total_rqst_mapped_out - Total de peticiones correlacionadas excluidas |
| rows_modified | xs:nonNegativeInteger | rows_modified - Filas modificadas |
| rows_read | xs:nonNegativeInteger | rows_read - Filas leídas |
| rows_returned | xs:nonNegativeInteger | rows_returned - Filas devueltas |
| tcpip_rcv_volume | xs:nonNegativeInteger | tcpip_rcv_volume - Volumen recibido TCP/IP |
| tcpip_rcv_wait_time | xs:nonNegativeInteger | tcpip_rcv_wait_time - Tiempo de espera de recepción TCP/IP |
| tcpip_recvs_total | xs:nonNegativeInteger | tcpip_recvs_total - Total de recepciones TCP/IP |
| tcpip_send_volume | xs:nonNegativeInteger | tcpip_send_volume - Volumen de envío TCP/IP |
| tcpip_send_wait_time | xs:nonNegativeInteger | tcpip_send_wait_time - Tiempo de espera de envío TCP/IP |
| tcpip_sends_total | xs:nonNegativeInteger | tcpip_sends_total - Total de envíos TCP/IP |
| total_app_rqst_time | xs:nonNegativeInteger | total_app_rqst_time - Total de tiempo de petición de aplicaciones |
| total_cpu_time | xs:nonNegativeInteger | total_cpu_time - Tiempo total de CPU |
| total_rqst_time | xs:nonNegativeInteger | total_rqst_time - Tiempo total de petición |
| total_section_sort_time | xs:nonNegativeInteger | total_section_sort_time - Tiempo total de clasificación de sección |
| total_section_sort_proc_time | xs:nonNegativeInteger | total_section_sort_proc_time - Tiempo total de proceso de clasificación de sección |
| total_section_sorts | xs:nonNegativeInteger | total_section_sorts - Total de clasificaciones de sección |
| total_sorts | xs:nonNegativeInteger | total_sorts - Número total de clasificaciones |

Tabla 105. Métricas detalladas devueltas para MON_GET_SERVICE_SUBCLASS_DETAILS (continuación)

| Nombre de elemento | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------------|-----------------------|--|
| post_threshold_sorts | xs:nonNegativeInteger | post_shrthreshold_sorts - Clasificaciones de umbral compartidas posteriores |
| post_shrthreshold_sorts | xs:nonNegativeInteger | post_shrthreshold_sorts - Clasificaciones de umbral compartidas posteriores |
| sort_overflows | xs:nonNegativeInteger | sort_overflows - Desbordamientos de clasificación |
| tq_tot_send_spills | xs:nonNegativeInteger | tq_tot_send_spills - Número total de almacenamientos intermedios de colas de tabla desbordados |
| total_wait_time | xs:nonNegativeInteger | total_wait_time - Tiempo total de espera |
| wlm_queue_time_total | xs:nonNegativeInteger | wlm_queue_time_total - Tiempo de cola total del gestor de cargas de trabajo |
| wlm_queue_assignments_total | xs:nonNegativeInteger | wlm_queue_assignments_total - Total de asignaciones de cola del gestor de cargas de trabajo |

Función de tabla MON_GET_TABLE - Obtener métrica de tabla

La función de tabla MON_GET_TABLE devuelve métricas del supervisor para una o varias tablas.

Sintaxis

►►MON_GET_TABLE(—*esquematabla*—, —*nombretabla*—, —*miembro*—)◀◀

El esquema es SYSPROC.

Parámetros de la función de tabla

esquematabla

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de esquema de tablas válido de la base de datos conectada actualmente cuando se llama a esta función. Si el argumento es nulo o una serie vacía, se recuperan métricas correspondientes a todas las tablas de todos los esquemas de la base de datos. Si se especifica el argumento, solamente se devuelven métricas correspondientes a las tablas del esquema especificado.

nombretabla

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de tabla válido de la base de datos conectada actualmente cuando se llama a esta función. Si el argumento es nulo o una serie vacía, se recuperan métricas para todas las tablas de la base de datos.

miembro

Argumento de entrada de tipo INTEGER que especifica un miembro válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique -1 para el miembro de base de datos actual o -2 para todos los miembros de base de datos. Si se especifica el valor NULL, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE sobre la función MON_GET_TABLE.

Ejemplo

Enumerar las actividades en todas las tablas a las que se ha accedido desde que se ha activado la base de datos, agregadas en todos los miembros de base de datos, ordenadas por el mayor número de lecturas.

```
SELECT varchar(tabschema,20) as tabschema,
       varchar(tabname,20) as tabname,
       sum(rows_read) as total_rows_read,
       sum(rows_inserted) as total_rows_inserted,
       sum(rows_updated) as total_rows_updated,
       sum(rows_deleted) as total_rows_deleted
FROM TABLE(MON_GET_TABLE('', '-2')) AS t
GROUP BY tabschema, tabname
ORDER BY total_rows_read DESC
```

A continuación se muestra un ejemplo de salida de esta consulta.

| TABSCHEMA | TABNAME | TOTAL_ROWS_READ | ... |
|-----------|----------|-----------------|-----|
| ----- | ----- | ----- | ... |
| SYSIBM | SYSHISTO | 113 | ... |
| SYSIBM | SYSWORKL | 22 | ... |
| SYSIBM | SYSROUTI | 13 | ... |
| SYSIBM | SYSSERVI | 13 | ... |
| SYSIBM | SYSTHRES | 6 | ... |
| SYSIBM | SYSTABLE | 3 | ... |
| SYSIBM | SYSCONTE | 2 | ... |
| SYSIBM | SYSDBAUT | 2 | ... |
| SYSIBM | SYSEVENT | 2 | ... |
| SYSIBM | SYSPLAN | 1 | ... |
| SYSIBM | SYSSURRO | 1 | ... |
| SYSIBM | SYSVERSI | 1 | ... |
| SYSIBM | SYSXMLST | 1 | ... |
| SYSIBM | SYSAUDIT | 0 | ... |
| SYSIBM | SYSROLEA | 0 | ... |
| SYSIBM | SYSROLES | 0 | ... |
| SYSIBM | SYSTASKS | 0 | ... |
| SYSIBM | SYSWORKA | 0 | ... |
| SYSIBM | SYSXMLPA | 0 | ... |

19 registros(s) seleccionado(s).

Salida de la consulta (continuación).

| ... | TOTAL_ROWS_INSERTED | TOTAL_ROWS_UPDATED | TOTAL_ROWS_DELETED |
|-------|---------------------|--------------------|--------------------|
| ----- | ----- | ----- | ----- |
| ... | 0 | 0 | 0 |
| ... | 0 | 0 | 0 |
| ... | 0 | 0 | 0 |
| ... | 0 | 0 | 0 |
| ... | 0 | 0 | 0 |
| ... | 0 | 0 | 0 |
| ... | 0 | 0 | 0 |
| ... | 0 | 0 | 0 |
| ... | 0 | 0 | 0 |
| ... | 0 | 0 | 0 |
| ... | 0 | 0 | 0 |
| ... | 0 | 0 | 0 |
| ... | 0 | 0 | 0 |
| ... | 0 | 0 | 0 |
| ... | 0 | 0 | 0 |
| ... | 0 | 0 | 0 |
| ... | 0 | 0 | 0 |
| ... | 0 | 0 | 0 |
| ... | 0 | 0 | 0 |
| ... | 0 | 0 | 0 |
| ... | 0 | 0 | 0 |

Notas de uso

La función de tabla `MON_GET_TABLE` devuelve una fila de datos por tabla de base de datos y por miembro de base de datos. Si se utilizan tablas particionadas por rangos, se devuelve una fila para cada partición de tabla por miembro de base de datos. No se realiza una agregación en los miembros de base de datos. Sin embargo, se puede lograr la agregación mediante consultas de SQL, tal como se muestra en el ejemplo.

Sólo se devolverán métricas correspondientes a las tablas a las que se ha accedido desde que se activó la base de datos.

Las métricas siempre están habilitadas. No es necesario activar ningún conmutador del supervisor del sistema para acceder a las métricas de tabla mediante esta función.

Información devuelta

Tabla 106. Información devuelta para `MON_GET_TABLE`

| Nombre de columna | Tipo de datos | Descripción |
|--------------------|---------------|---|
| TABSCHEMA | VARCHAR(128) | table_schema - Nombre de esquema de tabla |
| TABNAME | VARCHAR(128) | table_name - Nombre de tabla |
| MEMBER | SMALLINT | member - Miembro de base de datos |
| TAB_TYPE | VARCHAR(14) | table_type - Tipo de tabla. Esta interfaz devuelve un identificador de texto basado en las sentencias define de <code>sqlmon.h</code> y es uno de los siguientes: <ul style="list-style-type: none">• USER_TABLE• DROPPED_TABLE• TEMP_TABLE• CATALOG_TABLE• REORG_TABLE |
| TAB_FILE_ID | BIGINT | table_file_id - ID de archivo de tabla |
| DATA_PARTITION_ID | INTEGER | data_partition_id - Identificador de partición de datos |
| TBSP_ID | BIGINT | tablespace_id - Identificación del espacio de tablas |
| INDEX_TBSP_ID | BIGINT | index_tbsp_id - ID de espacio de tablas de índice |
| LONG_TBSP_ID | BIGINT | long_tbsp_id - ID de espacio de tablas largo |
| TABLE_SCANS | BIGINT | table_scans - Exploraciones de tabla |
| ROWS_READ | BIGINT | rows_read - Filas leídas |
| ROWS_INSERTED | BIGINT | rows_inserted - Filas insertadas |
| ROWS_UPDATED | BIGINT | rows_updated - Filas actualizadas |
| ROWS_DELETED | BIGINT | rows_deleted - Filas suprimidas |
| OVERFLOW_ACCESSES | BIGINT | overflow_accesses - Accesos a los registros desbordados |
| OVERFLOW_CREATES | BIGINT | overflow_creates - Creaciones de desbordamiento |
| PAGE_REORGS | BIGINT | Reservado para una utilización futura. |
| ADDITIONAL_DETAILS | BLOB(100K) | Reservado para una utilización futura. |

Función de tabla MON_GET_TABLESPACE - Obtener métrica de espacio de tablas

La función de tabla MON_GET_TABLESPACE devuelve métricas del supervisor para uno o varios espacios de tablas.

Sintaxis

►►—MON_GET_TABLESPACE—(—nombre_espaciotablas—,—miembro—)—————►►

El esquema es SYSPROC.

Parámetros de la función de tabla

nombre_espaciotablas

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de espacio de tablas válido de la base de datos conectada actualmente cuando se llama a esta función. Si el argumento es nulo o una serie vacía, se recuperan métricas para todos los espacios de tablas de la base de datos.

miembro

Argumento de entrada de tipo INTEGER que especifica un miembro válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique -1 para el miembro de base de datos actual o -2 para todos los miembros de base de datos. Si se especifica el valor NULL, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE sobre la función MON_GET_TABLESPACE.

Ejemplo

Enumerar espacios de tablas ordenados por el número de lecturas físicas desde los contenedores de espacios de tablas.

```
SELECT varchar(tbsp_name, 30) as tbsp_name,
       member,
       tbsp_type,
       pool_data_p_reads
FROM TABLE(MON_GET_TABLESPACE('','-2)) AS t
ORDER BY pool_data_p_reads DESC
```

A continuación se muestra un ejemplo de salida de esta consulta.

| TBSP_NAME | MEMBER | TBSP_TYPE | POOL_DATA_P_READS |
|-------------|--------|-----------|-------------------|
| SYSCATSPACE | | 0 DMS | 79 |
| USERSPACE1 | | 0 DMS | 34 |
| TEMPSPACE1 | | 0 SMS | 0 |

3 registro(s) seleccionado(s).

Notas de uso

La función de tabla MON_GET_TABLESPACE devuelve una fila de datos por espacio de tablas de base de datos y por miembro de base de datos. No se realiza una agregación en los miembros de base de datos. Sin embargo, se puede lograr la agregación mediante consultas de SQL.

La métrica recopilada por esta función se controla a nivel de base de datos mediante el parámetro de configuración `mon_obj_metrics`. Por omisión, la recopilación de métrica está habilitada.

Información devuelta

Tabla 107. Información devuelta para `MON_GET_TABLESPACE`

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|--|
| TBSP_NAME | VARCHAR(128) | tablespace_name - Nombre de espacio de tablas |
| TBSP_ID | BIGINT | tablespace_id - Identificación del espacio de tablas |
| MEMBER | SMALLINT | member - Miembro de base de datos |
| TBSP_TYPE | VARCHAR(10) | tablespace_type - Tipo de espacio de tablas. Esta interfaz devuelve un identificador de texto basado en las sentencias define de <code>sqlutil.h</code> , y es uno de los siguientes: <ul style="list-style-type: none"> • DMS • SMS |
| TBSP_CONTENT_TYPE | VARCHAR(10) | tablespace_content_type - Tipo de contenido de espacio de tablas. Esta interfaz devuelve un identificador de texto basado en las sentencias define de <code>sqlmon.h</code> y es uno de los siguientes: <ul style="list-style-type: none"> • ANY • LARGE • SYSTEMP • USRTEMP |
| TBSP_PAGE_SIZE | BIGINT | tablespace_page_size - Tamaño de página de espacio de tablas |
| TBSP_EXTENT_SIZE | BIGINT | tablespace_extent_size - Tamaño de extensión de espacio de tablas |
| TBSP_PREFETCH_SIZE | BIGINT | tablespace_prefetch_size - Tamaño de captación previa de espacio de tablas |
| TBSP_CUR_POOL_ID | BIGINT | tablespace_cur_pool_id - Agrupación de almacenamientos intermedios que se utiliza en la actualidad |
| TBSP_NEXT_POOL_ID | BIGINT | tablespace_next_pool_id - Agrupación de almacenamientos intermedios que se utilizará en el siguiente arranque |
| FS_CACHING | SMALLINT | fs_caching - Antememoria del sistema de archivos |
| TBSP_REBALANCER_MODE | VARCHAR(30) | tablespace_rebalancer_mode - Modalidad de reequilibrador. Esta interfaz devuelve un identificador de texto basado en las sentencias define de <code>sqlmon.h</code> y es uno de los siguientes: <ul style="list-style-type: none"> • NO_REBAL • FWD_REBAL • REV_REBAL • FWD_REBAL_OF_2PASS • REV_REBAL_OF_2PASS |
| TBSP_USING_AUTO_STORAGE | SMALLINT | tablespace_using_auto_storage - Espacio de tablas habilitado para el almacenamiento automático |

Tabla 107. Información devuelta para MON_GET_TABLESPACE (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------------|---------------|---|
| TBSP_AUTO_RESIZE_ENABLED | SMALLINT | tablespace_auto_resize_enabled - Espacio de tablas habilitado para la modificación de tamaño automática |
| DIRECT_READS | BIGINT | direct_reads - Lecturas directas de la base de datos |
| DIRECT_READ_REQS | BIGINT | direct_read_reqs - Peticiones de lectura directa |
| DIRECT_WRITES | BIGINT | direct_writes - Grabaciones directas en la base de datos |
| DIRECT_WRITE_REQS | BIGINT | direct_write_reqs - Peticiones de grabación directa |
| POOL_DATA_L_READS | BIGINT | pool_data_l_reads - Lecturas lógicas de datos de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_L_READS | BIGINT | pool_temp_data_l_reads - Lecturas lógicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_XDA_L_READS | BIGINT | pool_xda_l_reads - Lecturas lógicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_L_READS | BIGINT | pool_temp_xda_l_reads - Lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |
| POOL_INDEX_L_READS | BIGINT | pool_index_l_reads - Lecturas lógicas de índices de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_L_READS | BIGINT | pool_temp_index_l_reads - Lecturas lógicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_DATA_P_READS | BIGINT | pool_data_p_reads - Lecturas físicas de datos de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_P_READS | BIGINT | pool_temp_data_p_reads - Lecturas físicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_XDA_P_READS | BIGINT | pool_xda_p_reads - Lecturas físicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_P_READS | BIGINT | pool_temp_xda_p_reads - Lecturas físicas de datos XDA temporales de agrupación de almacenamientos intermedios |
| POOL_INDEX_P_READS | BIGINT | pool_index_p_reads - Lecturas físicas de índices de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_P_READS | BIGINT | pool_temp_index_p_reads - Lecturas físicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_DATA_WRITES | BIGINT | pool_data_writes - Grabaciones de datos de la agrupación de almacenamientos intermedios |
| POOL_XDA_WRITES | BIGINT | pool_xda_writes - Grabaciones de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_INDEX_WRITES | BIGINT | pool_index_writes - Grabaciones de índices de la agrupación de almacenamientos intermedios |
| DIRECT_READ_TIME | BIGINT | direct_read_time - Tiempo de lectura directa |

Tabla 107. Información devuelta para MON_GET_TABLESPACE (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|----------------------------|---------------|---|
| DIRECT_WRITE_TIME | BIGINT | direct_write_time - Tiempo de grabación directa |
| POOL_READ_TIME | BIGINT | pool_read_time - Tiempo total de lectura física de la agrupación de almacenamientos intermedios |
| POOL_WRITE_TIME | BIGINT | pool_write_time - Tiempo total de grabación física de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_READS | BIGINT | pool_async_data_reads - Lecturas de datos asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_READ_REQS | BIGINT | pool_async_data_read_reqs - Peticiones de lectura asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_WRITES | BIGINT | pool_async_data_writes - Grabaciones de datos asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_READS | BIGINT | pool_async_index_reads - Lecturas de índice asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_READ_REQS | BIGINT | pool_async_index_read_reqs - Peticiones de lectura de índice asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_WRITES | BIGINT | pool_async_index_writes - Grabaciones de índice asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_XDA_READS | BIGINT | pool_async_xda_reads - Lecturas de datos XDA asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_XDA_READ_REQS | BIGINT | pool_async_xda_read_reqs - Peticiones de lectura XDA asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_XDA_WRITES | BIGINT | pool_async_xda_writes - Grabaciones de datos XDA asíncronas de agrupación de almacenamientos intermedios |
| VECTORED_IOS | BIGINT | vectored_ios - Número de peticiones de E/S de vector |
| PAGES_FROM_VECTORED_IOS | BIGINT | pages_from_vectored_ios - Número total de páginas leídas por E/S de vector |
| BLOCK_IOS | BIGINT | block_ios - Número de peticiones de E/S de bloque |
| PAGES_FROM_BLOCK_IOS | BIGINT | pages_from_block_ios - Número total de páginas leídas por E/S de bloque |
| UNREAD_PREFETCH_PAGES | BIGINT | unread_prefetch_pages - Páginas de captación previa no leídas |
| FILES_CLOSED | BIGINT | files_closed - Archivos de base de datos cerrados |
| TBSP_STATE | VARCHAR(256) | tablespace_state - Estado de espacio de tablas |
| TBSP_USED_PAGES | BIGINT | tablespace_used_pages - Páginas utilizadas en espacio de tablas |
| TBSP_FREE_PAGES | BIGINT | tablespace_free_pages - Páginas libres en espacio de tablas |

Tabla 107. Información devuelta para MON_GET_TABLESPACE (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------------|---------------|---|
| TBSP_USABLE_PAGES | BIGINT | tablespace_usable_pages - Páginas utilizables en espacio de tablas |
| TBSP_TOTAL_PAGES | BIGINT | tablespace_total_pages - Total de páginas en espacio de tablas |
| TBSP_PENDING_FREE_PAGES | BIGINT | tablespace_pending_free_pages - Páginas libres pendientes en espacio de tablas |
| TBSP_PAGE_TOP | BIGINT | tablespace_page_top - Nivel límite de espacio de tablas |
| TBSP_MAX_PAGE_TOP | BIGINT | tablespace_max_size - Tamaño máximo de espacio de tablas |
| RECLAIMABLE_SPACE_ENABLED | SMALLINT | reclaimable_space_enabled - Indicador habilitado de espacio reclamable |
| AUTO_STORAGE_HYBRID | SMALLINT | auto_storage_hybrid - Indicador de espacio de tablas de almacenamiento automático híbrido |
| TBSP_PATHS_DROPPED | SMALLINT | tablespace_paths_dropped - Espacio de tablas utilizando vía de acceso descartada |
| ADDITIONAL_DETAILS | BLOB(100K) | Reservado para una utilización futura. |

Función de tabla MON_GET_UNIT_OF_WORK - Obtener métrica de unidad de trabajo

La función de tabla MON_GET_UNIT_OF_WORK devuelve métricas para una o varias unidades de trabajo.

Sintaxis

►►—MON_GET_UNIT_OF_WORK—(—descriptorcontexto_aplicación—,—miembro—)——►►

El esquema es SYSPROC.

Parámetros de la función de tabla

descriptorcontexto_aplicación

Argumento de entrada opcional de tipo BIGINT que especifica un descriptor de contexto de aplicación válido en la misma base de datos que la conectada actualmente al llamar a esta función. Si el argumento es nulo, se recuperan métricas correspondientes a las unidades de trabajo en ejecución en todas las superclases de la base de datos.

miembro

Argumento de entrada opcional de tipo INTEGER que especifica un miembro válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique -1 para el miembro de base de datos actual o -2 para todos los miembros de base de datos. Si se especifica el valor NULL, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE sobre la función MON_GET_UNIT_OF_WORK.

Ejemplo

Identificar las unidades de trabajo que consumen el mayor tiempo de CPU en el sistema.

```
SELECT application_handle,
       uow_id,
       total_cpu_time,
       app_rqsts_completed_total,
       rqsts_completed_total
FROM TABLE(MON_GET_UNIT_OF_WORK(NULL,-1)) AS t
ORDER BY total_cpu_time DESC
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| APPLICATION_HANDLE | UOW_ID | TOTAL_CPU_TIME | ... |
|--------------------|--------|----------------|-----------|
| ----- | ----- | ----- | ... |
| | 46 | 5 | 27959 ... |

1 registro(s) seleccionado(s).

Salida de la consulta (continuación).

| | | |
|-----|---------------------------|-----------------------|
| ... | APP_RQSTS_COMPLETED_TOTAL | RQSTS_COMPLETED_TOTAL |
| ... | ----- | ----- |
| ... | 72 | 48 |

Notas de uso

Las métricas devueltas por la función de tabla `MON_GET_UNIT_OF_WORK` representan la suma de todas las métricas para las peticiones enviadas durante una unidad de trabajo. Las métricas se avanzan de forma no periódica durante la unidad de trabajo. Por consiguiente, los valores notificados por esta función de tabla reflejan el estado actual del sistema en el momento del avance más reciente. Las métricas son estrictamente de valor creciente. Para determinar el valor de una métrica dada durante un intervalo de tiempo, utilice la función para consultar la métrica al principio y al final del intervalo y calcular la diferencia.

Las métricas de petición se controlan mediante la cláusula `COLLECT REQUEST METRICS` en las superclases de servicio y el parámetro de configuración de base de datos `mon_req_metrics` en el nivel de la base de datos. Las métricas únicamente se recopilan para una petición si la petición se procesa a través de un agente en una subclase de servicio cuya superclase de servicio padre tiene habilitadas las métricas de petición o si la recopilación de métricas de petición está habilitada para toda la base de datos. Por omisión, las métricas de petición están habilitadas en el nivel de la base de datos. Si se han inhabilitado las métricas de petición en el nivel de la base de datos y para una superclase de servicio, las métricas notificadas para cada unidad de trabajo correlacionada con dicha superclase de servicio dejan de aumentar (o permanecen en 0 si las métricas de petición estaban inhabilitadas en el momento de la activación de la base de datos).

La función de tabla `MON_GET_UNIT_OF_WORK` devuelve una fila de datos por unidad de trabajo y por miembro. No se realiza ninguna agregación entre unidades de trabajo (en un miembro) ni entre miembros (correspondientes a una o varias clases de servicio). Sin embargo, se puede lograr la agregación mediante consultas de SQL. Los parámetros de entrada tienen el efecto de sumarse (operación AND). Por consiguiente, si se especifican parámetros de entrada conflictivos (por ejemplo, un nombre de superclase `SUPA` y un nombre de subclase `SUBB` que no es una subclase de `SUPA`), no se devuelve ninguna fila.

Información devuelta

Tabla 108. Información devuelta para MON_GET_UNIT_OF_WORK

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------------|---------------|--|
| SERVICE_SUPERCLASS_NAME | VARCHAR(128) | service_superclass_name - Nombre de superclase de servicio |
| SERVICE_SUBCLASS_NAME | VARCHAR(128) | service_subclass_name - Nombre de subclase de servicio |
| SERVICE_CLASS_ID | INTEGER | service_class_id - ID de clase de servicio |
| MEMBER | SMALLINT | member - Miembro de base de datos |
| COORD_MEMBER | SMALLINT | coord_member - Miembro del coordinador |
| APPLICATION_HANDLE | BIGINT | application_handle - Descriptor de contexto de aplicación |
| APPLICATION_ID | VARCHAR(128) | Reservado para una utilización futura. |
| WORKLOAD_NAME | VARCHAR(128) | workload_name - Nombre de carga de trabajo |
| WORKLOAD_OCCURRENCE_ID | INTEGER | workload_occurrence_id - Identificador de la aparición de carga de trabajo. Este ID no identifica la aparición de carga de trabajo de forma exclusiva a no ser que esté combinado con el miembro del coordinador y con el nombre de la carga de trabajo. |
| UOW_ID | INTEGER | uow_id - ID de unidad de trabajo |
| WORKLOAD_OCCURRENCE_STATE | VARCHAR(32) | workload_occurrence_state - Estado de la aparición de carga de trabajo |
| CLIENT_WRKSTNNAME | VARCHAR(255) | Registro especial CURRENT CLIENT_WRKSTNNAME |
| CLIENT_ACCTNG | VARCHAR(255) | Registro especial CURRENT CLIENT_ACCTNG |
| CLIENT_USERID | VARCHAR(255) | Registro especial CURRENT CLIENT_USERID |
| CLIENT_APPLNAME | VARCHAR(255) | Registro especial CURRENT CLIENT_APPLNAME |
| UOW_START_TIME | TIMESTAMP | Reservado para una utilización futura. |
| SESSION_AUTH_ID | VARCHAR(128) | Reservado para una utilización futura. |
| ACT_ABORTED_TOTAL | BIGINT | act_aborted_total - Total de actividades abortadas |
| ACT_COMPLETED_TOTAL | BIGINT | act_completed_total - Total de actividades completadas |
| ACT_REJECTED_TOTAL | BIGINT | act_rejected_total - Total de actividades rechazadas |
| AGENT_WAIT_TIME | BIGINT | agent_wait_time - Tiempo de espera del agente |
| AGENT_WAITS_TOTAL | BIGINT | agent_waits_total - Total de esperas del agente |
| POOL_DATA_L_READS | BIGINT | pool_data_l_reads - Lecturas lógicas de datos de la agrupación de almacenamientos intermedios |
| POOL_INDEX_L_READS | BIGINT | pool_index_l_reads - Lecturas lógicas de índice de agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_L_READS | BIGINT | pool_temp_data_l_reads - Lecturas lógicas de datos temporales de agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_L_READS | BIGINT | pool_temp_index_l_reads - Lecturas lógicas de índice temporal de agrupación de almacenamientos intermedios |

Tabla 108. Información devuelta para MON_GET_UNIT_OF_WORK (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|--|
| POOL_TEMP_XDA_L_READS | BIGINT | pool_temp_xda_l_reads - Lecturas lógicas de datos XDA temporales de agrupación de almacenamientos intermedios |
| POOL_XDA_L_READS | BIGINT | pool_xda_l_reads - Lecturas lógicas de datos XDA de agrupación de almacenamientos intermedios |
| POOL_DATA_P_READS | BIGINT | pool_data_p_reads - Lecturas físicas de datos de la agrupación de almacenamientos intermedios |
| POOL_INDEX_P_READS | BIGINT | pool_index_p_reads - Lecturas físicas de índice de agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_P_READS | BIGINT | pool_temp_data_p_reads - Lecturas físicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_P_READS | BIGINT | pool_temp_index_p_reads - Lecturas físicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_P_READS | BIGINT | pool_temp_xda_p_reads - Lecturas físicas de datos XDA temporales de agrupación de almacenamientos intermedios |
| POOL_XDA_P_READS | BIGINT | pool_xda_p_reads - Lecturas físicas de datos XDA de agrupación de almacenamientos intermedios |
| POOL_DATA_WRITES | BIGINT | pool_data_writes - Grabaciones de datos de agrupación de almacenamientos intermedios |
| POOL_INDEX_WRITES | BIGINT | pool_index_writes - Grabaciones de índice de agrupación de almacenamientos intermedios |
| POOL_XDA_WRITES | BIGINT | pool_xda_writes - Grabaciones de datos XDA de agrupación de almacenamientos intermedios |
| POOL_READ_TIME | BIGINT | pool_read_time - Tiempo total de lectura física de agrupación de almacenamientos intermedios |
| POOL_WRITE_TIME | BIGINT | pool_write_time - Tiempo total de grabación física de agrupación de almacenamientos intermedios |
| CLIENT_IDLE_WAIT_TIME | BIGINT | client_idle_wait_time - Tiempo de espera de inactividad del cliente |
| DEADLOCKS | BIGINT | deadlocks - Puntos muertos detectados |
| DIRECT_READS | BIGINT | direct_reads - Lecturas directas desde base de datos |
| DIRECT_READ_TIME | BIGINT | direct_read_time - Tiempo de lectura directa |
| DIRECT_WRITES | BIGINT | direct_writes - Grabaciones directas en base de datos |
| DIRECT_WRITE_TIME | BIGINT | direct_write_time - Tiempo de grabación directa |
| DIRECT_READ_REQS | BIGINT | direct_read_reqs - Peticiones de lectura directa |
| DIRECT_WRITE_REQS | BIGINT | direct_write_reqs - Solicitudes de grabación directa |
| FCM_RECV_VOLUME | BIGINT | fcm_recv_volume - Volumen de recepción de FCM |
| FCM_RECVS_TOTAL | BIGINT | fcm_recvs_total - Total de recepciones de FCM |
| FCM_SEND_VOLUME | BIGINT | fcm_send_volume - Volumen de envío de FCM |
| FCM_SENDS_TOTAL | BIGINT | fcm_sends_total - Total de envíos de FCM |

Tabla 108. Información devuelta para MON_GET_UNIT_OF_WORK (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------|---------------|--|
| FCM_RECV_WAIT_TIME | BIGINT | fcm_recv_wait_time - Tiempo de espera de recepción de FCM |
| FCM_SEND_WAIT_TIME | BIGINT | fcm_send_wait_time - Tiempo de espera de envío de FCM |
| IPC_RECV_VOLUME | BIGINT | ipc_recv_volume - Volumen de recepción de comunicación entre procesos |
| IPC_RECV_WAIT_TIME | BIGINT | ipc_recv_wait_time - Tiempo de espera de recepción de comunicación entre procesos |
| IPC_RECVS_TOTAL | BIGINT | ipc_recvs_total - Total de recepciones de comunicación entre procesos |
| IPC_SEND_VOLUME | BIGINT | ipc_send_volume - Volumen de envío de comunicación entre procesos |
| IPC_SEND_WAIT_TIME | BIGINT | ipc_send_wait_time - Tiempo de espera de envío de comunicación entre procesos |
| IPC_SENDS_TOTAL | BIGINT | ipc_sends_total - Total de envíos de comunicación entre procesos |
| LOCK_ESCALS | BIGINT | lock_escals - Número de escalamientos de bloqueo |
| LOCK_TIMEOUTS | BIGINT | lock_timeouts - Número de tiempos de espera de bloqueo |
| LOCK_WAIT_TIME | BIGINT | lock_wait_time - Tiempo esperado en bloqueos |
| LOCK_WAITS | BIGINT | lock_waits - Esperas de bloqueo |
| LOG_BUFFER_WAIT_TIME | BIGINT | log_buffer_wait_time - Tiempo de espera de almacenamiento intermedio de anotaciones cronológicas |
| NUM_LOG_BUFFER_FULL | BIGINT | num_log_buffer_full - Número de almacenamientos intermedios de anotaciones cronológicas llenos |
| LOG_DISK_WAIT_TIME | BIGINT | log_disk_wait_time - Tiempo de espera de disco de anotaciones cronológicas |
| LOG_DISK_WAITS_TOTAL | BIGINT | log_disk_waits_total - Total de esperas de disco de anotaciones cronológicas |
| NUM_LOCKS_HELD | BIGINT | Reservado para una utilización futura. |
| RQSTS_COMPLETED_TOTAL | BIGINT | rqsts_completed_total - Total de peticiones completadas |
| ROWS_MODIFIED | BIGINT | rows_modified - Filas modificadas |
| ROWS_READ | BIGINT | rows_read - Filas leídas |
| ROWS_RETURNED | BIGINT | rows_returned - Filas devueltas |
| TCPIP_RECV_VOLUME | BIGINT | tcPIP_recv_volume - Volumen recibido TCP/IP |
| TCPIP_SEND_VOLUME | BIGINT | tcPIP_send_volume - Volumen de envío TCP/IP |
| TCPIP_RECV_WAIT_TIME | BIGINT | tcPIP_recv_wait_time - Tiempo de espera de recepción TCP/IP |
| TCPIP_RECVS_TOTAL | BIGINT | tcPIP_recvs_total - Total de recepciones TCP/IP |
| TCPIP_SEND_WAIT_TIME | BIGINT | tcPIP_send_wait_time - Tiempo de espera de envío TCP/IP |
| TCPIP_SENDS_TOTAL | BIGINT | tcPIP_sends_total - Total de envíos TCP/IP |

Tabla 108. Información devuelta para MON_GET_UNIT_OF_WORK (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|----------------------------------|---------------|---|
| TOTAL_APP_RQST_TIME | BIGINT | total_app_rqst_time - Total de tiempo de petición de aplicaciones |
| TOTAL_RQST_TIME | BIGINT | total_rqst_time - Tiempo total de petición |
| WLM_QUEUE_TIME_TOTAL | BIGINT | wlm_queue_time_total - Tiempo de cola total del gestor de cargas de trabajo |
| WLM_QUEUE_ASSIGNMENTS_TOTAL | BIGINT | wlm_queue_assignments_total - Total de asignaciones de cola del gestor de cargas de trabajo |
| TOTAL_CPU_TIME | BIGINT | total_cpu_time - Tiempo total de CPU |
| TOTAL_WAIT_TIME | BIGINT | total_wait_time - Tiempo total de espera |
| APP_RQSTS_COMPLETED_TOTAL | BIGINT | app_rqsts_completed_total - Total de peticiones de aplicación completadas |
| TOTAL_SECTION_SORT_TIME | BIGINT | total_section_sort_time - Tiempo total de clasificación de sección |
| TOTAL_SECTION_SORT_PROC_TIME | BIGINT | total_section_sort_proc_time - Tiempo total de proceso de clasificación de sección |
| TOTAL_SECTION_SORTS | BIGINT | total_section_sorts - Total de clasificaciones de sección |
| TOTAL_SORTS | BIGINT | total_sorts - Número total de clasificaciones |
| POST_THRESHOLD_SORTS | BIGINT | post_threshold_sorts - Clasificaciones de umbral posteriores |
| POST_SHRTHRESHOLD_SORTS | BIGINT | post_shrthreshold_sorts - Clasificaciones de umbral compartidas posteriores |
| SORT_OVERFLOWS | BIGINT | sort_overflows - Desbordamientos de clasificación |
| TOTAL_COMPILE_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_COMPILE_PROC_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_COMPILATIONS | BIGINT | Reservado para una utilización futura. |
| TOTAL_IMPLICIT_COMPILE_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_IMPLICIT_COMPILE_PROC_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_IMPLICIT_COMPILATIONS | BIGINT | Reservado para una utilización futura. |
| TOTAL_SECTION_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_SECTION_PROC_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_APP_SECTION_EXECUTIONS | BIGINT | Reservado para una utilización futura. |
| TOTAL_ACT_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_ACT_WAIT_TIME | BIGINT | Reservado para una utilización futura. |
| ACT_RQSTS_TOTAL | BIGINT | Reservado para una utilización futura. |
| TOTAL_ROUTINE_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_ROUTINE_INVOCATIONS | BIGINT | Reservado para una utilización futura. |
| TOTAL_COMMIT_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_COMMIT_PROC_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_APP_COMMITS | BIGINT | Reservado para una utilización futura. |

Tabla 108. Información devuelta para MON_GET_UNIT_OF_WORK (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------------|---------------|---|
| INT_COMMITS | BIGINT | Reservado para una utilización futura. |
| TOTAL_ROLLBACK_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_ROLLBACK_PROC_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_APP_ROLLBACKS | BIGINT | Reservado para una utilización futura. |
| INT_ROLLBACKS | BIGINT | Reservado para una utilización futura. |
| TOTAL_RUNSTATS_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_RUNSTATS_PROC_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_RUNSTATS | BIGINT | Reservado para una utilización futura. |
| TOTAL_REORG_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_REORG_PROC_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_REORGS | BIGINT | Reservado para una utilización futura. |
| TOTAL_LOAD_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_LOAD_PROC_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_LOADS | BIGINT | Reservado para una utilización futura. |
| CAT_CACHE_INSERTS | BIGINT | Reservado para una utilización futura. |
| CAT_CACHE_LOOKUPS | BIGINT | Reservado para una utilización futura. |
| PKG_CACHE_INSERTS | BIGINT | Reservado para una utilización futura. |
| PKG_CACHE_LOOKUPS | BIGINT | Reservado para una utilización futura. |
| THRESH_VIOLATIONS | BIGINT | Reservado para una utilización futura. |
| NUM_LW_THRESH_EXCEEDED | BIGINT | Reservado para una utilización futura. |
| UOW_LOG_SPACE_USED | BIGINT | Reservado para una utilización futura. |
| ADDITIONAL_DETAILS | BLOB(100K) | Reservado para una utilización futura. |

Función de tabla MON_GET_UNIT_OF_WORK_DETAILS - Obtener métrica detallada de unidad de trabajo

La función de tabla MON_GET_UNIT_OF_WORK_DETAILS devuelve métricas detalladas para una o varias unidades de trabajo.

Sintaxis

```

▶▶—MON_GET_UNIT_OF_WORK_DETAILS—(—————▶
▶—descriptorcontexto_aplicación—,—miembro—)————▶▶

```

El esquema es SYSPROC.

Parámetros de la función de tabla

descriptorcontexto_aplicación

Argumento de entrada de tipo BIGINT que especifica un descriptor de contexto de aplicación válido en la misma base de datos que la conectada

actualmente al llamar a esta función. Si el argumento es nulo, se recuperan métricas correspondientes a las unidades de trabajo en ejecución en todas las superclases de la base de datos.

miembro

Argumento de entrada de tipo INTEGER que especifica un miembro válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique -1 para el miembro de base de datos actual o -2 para todos los miembros de base de datos. Si se especifica el valor NULL, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE sobre la función MON_GET_UNIT_OF_WORK_DETAILS.

Ejemplo

Identificar las unidades de trabajo que consumen el mayor tiempo de CPU en el sistema.

```
SELECT detmetrics.application_handle,
       detmetrics.uow_id,
       detmetrics.total_cpu_time,
       detmetrics.app_rqsts_completed_total,
       detmetrics.rqsts_completed_total
FROM TABLE(MON_GET_UNIT_OF_WORK_DETAILS(NULL,-2)) AS UOWMETRICS,
XMLTABLE (XMLNAMESPACES( DEFAULT 'http://www.ibm.com/xmlns/prod/db2/mon'),
'$detmetric/db2_unit_of_work' PASSING
XMLPARSE(DOCUMENT UOWMETRICS.DETAILS)
as "detmetric"
COLUMNNS ("APPLICATION_HANDLE" INTEGER PATH 'application_handle',
"UOW_ID" INTEGER PATH 'uow_id',
"TOTAL_CPU_TIME" INTEGER PATH 'system_metrics/total_cpu_time',
"APP_RQSTS_COMPLETED_TOTAL" INTEGER PATH 'system_metrics/app_rqsts_completed_total',
"RQSTS_COMPLETED_TOTAL" INTEGER PATH 'system_metrics/rqsts_completed_total'
) AS DETMETRICS
ORDER BY total_cpu_time DESC
```

A continuación se muestra un ejemplo de salida de esta consulta.

| APPLICATION_HANDLE | UOW_ID | TOTAL_CPU_TIME | ... |
|--------------------|--------|----------------|-----------|
| ----- | ----- | ----- | ... |
| | 46 | 5 | 27959 ... |

1 registro(s) seleccionado(s).

Salida de la consulta (continuación).

| ... | APP_RQSTS_COMPLETED_TOTAL | RQSTS_COMPLETED_TOTAL |
|-----|---------------------------|-----------------------|
| ... | ----- | ----- |
| ... | | 72 |
| ... | | 48 |

Notas de uso

Las métricas devueltas por la función MON_GET_UNIT_OF_WORK_DETAILS representan la suma de todas las métricas para las peticiones enviadas durante una unidad de trabajo. Esta función es parecida a la función de tabla MON_GET_UNIT_OF_WORK:

- La función de tabla MON_GET_UNIT_OF_WORK devuelve las métricas utilizadas con más frecuencia en un formato basado en columnas, y es el método más eficiente, desde el punto de vista del rendimiento, de recuperar las métricas.

- La función de tabla `MON_GET_UNIT_OF_WORK_DETAILS` devuelve todo el conjunto de métricas disponibles en formato de documento XML, lo que proporciona la máxima flexibilidad para dar formato a la salida. La salida basada en XML puede analizarse directamente mediante un analizador XML, o puede convertirse a formato relacional mediante la función `XMLTABLE` (véase el ejemplo).

Las métricas se avanzan de forma no periódica durante la unidad de trabajo. Por consiguiente, los valores notificados por esta función de tabla reflejan el estado actual del sistema en el momento del avance más reciente. Las métricas son estrictamente de valor creciente. Para determinar el valor de una métrica dada durante un intervalo de tiempo, utilice la función de tabla `MON_GET_UNIT_OF_WORK_DETAILS` para consultar la métrica al principio y al final del intervalo y calcular la diferencia.

Las métricas de petición se controlan mediante la cláusula `COLLECT REQUEST METRICS` en las superclases de servicio y el parámetro de configuración de base de datos `mon_req_metrics` en el nivel de la base de datos. Las métricas únicamente se recopilan para una petición si la petición se procesa a través de un agente en una subclase de servicio cuya superclase de servicio padre tiene habilitadas las métricas de petición o si la recopilación de métricas de petición está habilitada para toda la base de datos. Por omisión, las métricas de petición están habilitadas en el nivel de la base de datos. Si se han inhabilitado las métricas de petición en el nivel de la base de datos y para una superclase de servicio, las métricas notificadas para cada unidad de trabajo correlacionada con dicha superclase de servicio dejan de aumentar (o permanecen en 0 si las métricas de petición estaban inhabilitadas en el momento de la activación de la base de datos).

La función de tabla `MON_GET_UNIT_OF_WORK_DETAILS` devuelve una fila de datos por unidad de trabajo y por miembro. No se realiza ninguna agregación entre unidades de trabajo (en un miembro) ni entre miembros (correspondientes a una o varias clases de servicio). Sin embargo, se puede lograr la agregación mediante consultas de SQL. Los parámetros de entrada tienen el efecto de sumarse (operación AND). Por consiguiente, si se especifican parámetros de entrada conflictivos (por ejemplo, un nombre de superclase SUPA y un nombre de subclase SUBB que no es una subclase de SUPA), no se devuelve ninguna fila.

El esquema para el documento XML devuelto en la columna `DETAILS` está disponible en el archivo `sql11b/misc/DB2MonRoutines.xsd`. Para obtener más información, consulte el archivo `sql11b/misc/DB2MonCommon.xsd`.

Información devuelta

Tabla 109. Información devuelta para `MON_GET_UNIT_OF_WORK_DETAILS`

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------------------------|---------------------------|---|
| <code>SERVICE_SUPERCLASS_NAME</code> | <code>VARCHAR(128)</code> | <code>service_superclass_name</code> - Nombre de superclase de servicio |
| <code>SERVICE_SUBCLASS_NAME</code> | <code>VARCHAR(128)</code> | <code>service_subclass_name</code> - Nombre de subclase de servicio |
| <code>SERVICE_CLASS_ID</code> | <code>INTEGER</code> | <code>service_class_id</code> - ID de clase de servicio |
| <code>MEMBER</code> | <code>SMALLINT</code> | <code>member</code> - Miembro de base de datos |

Tabla 109. Información devuelta para MON_GET_UNIT_OF_WORK_DETAILS (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------|---------------|---|
| COORD_MEMBER | SMALLINT | coord_partition_num - Miembro del coordinador Miembro de base de datos para la partición de coordinador de la unidad de trabajo dada. |
| APPLICATION_HANDLE | BIGINT | application_handle - Descriptor de contexto de aplicación |
| WORKLOAD_NAME | VARCHAR(128) | workload_name - Nombre de carga de trabajo |
| WORKLOAD_OCCURRENCE_ID | INTEGER | workload_occurrence_id - Identificador de la aparición de carga de trabajo. Este ID no identifica la aparición de carga de trabajo de forma exclusiva a no ser que esté combinado con el número de partición de base de datos de coordinación y con el nombre de la carga de trabajo. |
| UOW_ID | INTEGER | uow_id - ID de unidad de trabajo |
| DETAILS | BLOB(1M) | Documento XML que contiene métricas detalladas para la unidad de trabajo. Consulte la Tabla 110 para ver una descripción de los elementos de este documento. |

El ejemplo siguiente muestra la estructura del documento XML que se devuelve en la columna DETAILS.

```
<db2_unit_of_work xmlns="http://www.ibm.com/xmlns/prod/db2/mon" release="90700000">
  <service_superclass_name>SYSDEFAULTUSERCLASS</service_superclass_name>
  <service_subclass_name>SYSDEFAULTSUBCLASS</service_subclass_name>
  <service_class_id>13</service_class_id>
  <workload_name>SYSDEFAULTUSERWORKLOAD</workload_name>
  <member>0</member>
  <coord_member>0</coord_member>
  <application_handle>21</application_handle>
  <workload_occurrence_id>1</workload_occurrence_id>
  <uow_id>2</uow_id>
  <workload_occurrence_state>UOWEXEC</workload_occurrence_state>
  <system_metrics>
    <act_aborted_total>5</act_aborted_total>
    ...
    <wlm_queue_assignments_total>3</wlm_queue_assignments_total>
  </system_metrics>
</db2_unit_of_work_metrics>
```

Para ver el esquema completo, consulte `sqllib/misc/DB2MonRoutines.xsd`.

Tabla 110. Métricas detalladas devueltas para MON_GET_UNIT_OF_WORK_DETAILS

| Nombre de elemento | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|-----------------------|--|
| service_superclass_name | xs:string (128) | service_superclass_name - Nombre de superclase de servicio |
| service_subclass_name | xs:string (128) | service_subclass_name - Nombre de subclase de servicio |
| service_class_id | xs:nonNegativeInteger | service_class_id - ID de clase de servicio |
| workload_name | xs:string (128) | workload_name - Nombre de carga de trabajo |
| member | xs:nonNegativeInteger | member - Miembro de base de datos |

Tabla 110. Métricas detalladas devueltas para MON_GET_UNIT_OF_WORK_DETAILS (continuación)

| Nombre de elemento | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------------|-----------------------|---|
| coord_member | xs:nonNegativeInteger | coord_member - Miembro del coordinador |
| application_handle | xs:nonNegativeInteger | application_handle - Descriptor de contexto de aplicación |
| workload_occurrence_id | xs:nonNegativeInteger | workload_occurrence_id - Identificador de aparición de carga de trabajo Este ID no identifica de forma exclusiva la aparición de carga de trabajo a menos que esté acompañado del nombre de carga de trabajo y miembro del coordinador. |
| uow_id | xs:nonNegativeInteger | uow_id - ID de unidad de trabajo |
| workload_occurrence_state | xs:string | workload_occurrence_state - Estado de la aparición de carga de trabajo |
| client_userid | xs:string | Registro especial CURRENT CLIENT_USERID |
| client_wrkstnname | xs:string | Registro especial CURRENT CLIENT_WRKSTNNAME |
| client_applname | xs:string | Registro especial CURRENT CLIENT_APPLNAME |
| client_acctng | xs:string | Registro especial CURRENT CLIENT_ACCTNG |
| act_aborted_total | xs:nonNegativeInteger | act_aborted_total - Total de actividades abortadas |
| act_completed_total | xs:nonNegativeInteger | act_completed_total - Total de actividades completadas |
| act_rejected_total | xs:nonNegativeInteger | act_rejected_total - Total de actividades rechazadas |
| agent_wait_time | xs:nonNegativeInteger | agent_wait_time - Tiempo de espera del agente |
| agent_waits_total | xs:nonNegativeInteger | agent_waits_total - Total de esperas del agente |
| app_rqsts_completed_total | xs:nonNegativeInteger | app_rqsts_completed_total - Total de peticiones de aplicación completadas |
| audit_events_total | xs:nonNegativeInteger | audit_events_total - Total de sucesos de auditoría |
| audit_subsystem_wait_time | xs:nonNegativeInteger | audit_subsystem_wait_time - Tiempo de espera del subsistema de auditoría |
| audit_subsystem_waits_total | xs:nonNegativeInteger | audit_subsystem_waits_total - Total de esperas del subsistema de auditoría |
| audit_file_write_wait_time | xs:nonNegativeInteger | audit_file_write_wait_time - Tiempo de espera de grabación del archivo de auditoría |
| audit_file_writes_total | xs:nonNegativeInteger | audit_file_writes_total - Total de archivos de auditoría grabados |
| client_idle_wait_time | xs:nonNegativeInteger | client_idle_wait_time - Tiempo de espera de inactividad del cliente |
| deadlocks | xs:nonNegativeInteger | deadlocks - Puntos muertos detectados |
| diaglog_writes_total | xs:nonNegativeInteger | diaglog_writes_total - Total de grabaciones de anotaciones cronológicas de diagnóstico |
| diaglog_write_wait_time | xs:nonNegativeInteger | diaglog_write_wait_time - Tiempo de grabación de anotaciones cronológicas de diagnóstico |
| direct_read_time | xs:nonNegativeInteger | direct_read_time - Tiempo de lectura directa |
| direct_write_time | xs:nonNegativeInteger | direct_write_time - Tiempo de grabación directa |
| direct_read_reqs | xs:nonNegativeInteger | direct_read_reqs - Peticiones de lectura directa |
| direct_reads | xs:nonNegativeInteger | direct_reads - Lecturas directas de la base de datos |

Tabla 110. Métricas detalladas devueltas para MON_GET_UNIT_OF_WORK_DETAILS (continuación)

| Nombre de elemento | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|----------------------------|-----------------------|--|
| direct_write_reqs | xs:nonNegativeInteger | direct_write_reqs - Peticiones de grabación directa |
| direct_writes | xs:nonNegativeInteger | direct_writes - Grabaciones directas en la base de datos |
| fcm_recv_volume | xs:nonNegativeInteger | fcm_recv_volume - Volumen de recepción de FCM |
| fcm_recv_wait_time | xs:nonNegativeInteger | fcm_recv_wait_time - Tiempo de espera de recepción de FCM |
| fcm_recvs_total | xs:nonNegativeInteger | fcm_recvs_total - Total de recepciones de FCM |
| fcm_message_recv_volume | xs:nonNegativeInteger | fcm_message_recv_volume - Volumen de recepción de mensaje de FCM |
| fcm_message_recvs_total | xs:nonNegativeInteger | fcm_message_recvs_total - Total de recepciones de mensaje de FCM |
| fcm_message_recv_wait_time | xs:nonNegativeInteger | fcm_message_recv_wait_time - Tiempo de espera de recepción de mensaje de FCM |
| fcm_message_send_volume | xs:nonNegativeInteger | fcm_message_send_volume - Volumen de envío de mensaje de FCM |
| fcm_message_send_wait_time | xs:nonNegativeInteger | fcm_message_send_wait_time - Tiempo de espera de envío de mensaje de FCM |
| fcm_message_sends_total | xs:nonNegativeInteger | fcm_message_sends_total - Total de envíos de mensaje de FCM |
| fcm_send_volume | xs:nonNegativeInteger | fcm_send_volume - Volumen de envío de FCM |
| fcm_send_wait_time | xs:nonNegativeInteger | fcm_send_wait_time - Tiempo de espera de envío de FCM |
| fcm_sends_total | xs:nonNegativeInteger | fcm_sends_total - Total de envíos de FCM |
| fcm_tq_recv_wait_time | xs:nonNegativeInteger | fcm_tq_recv_wait_time - Tiempo de espera de recepción de cola de tabla de FCM |
| fcm_tq_send_wait_time | xs:nonNegativeInteger | fcm_tq_send_wait_time - Tiempo de espera de envío de cola de tabla de FCM |
| fcm_tq_recv_volume | xs:nonNegativeInteger | fcm_tq_recv_volume - Volumen de recepción de cola de tabla de FCM |
| fcm_tq_recvs_total | xs:nonNegativeInteger | fcm_tq_recvs_total - Total de recepciones de cola de tabla de FCM |
| fcm_tq_send_volume | xs:nonNegativeInteger | fcm_tq_send_volume - Volumen de envíos de cola de tabla de FCM |
| fcm_tq_sends_total | xs:nonNegativeInteger | fcm_tq_sends_total - Total de envíos de cola de tabla de FCM |
| tq_tot_send_spills | xs:nonNegativeInteger | tq_tot_send_spills - Total de almacenamientos intermedios de cola de tabla desbordados |
| ipc_recv_volume | xs:nonNegativeInteger | ipc_recv_volume - Volumen de recepción de comunicación entre procesos |
| ipc_recv_wait_time | xs:nonNegativeInteger | ipc_recv_wait_time - Tiempo de espera de recepción de comunicación entre procesos |
| ipc_recvs_total | xs:nonNegativeInteger | ipc_recvs_total - Total de recepciones de comunicación entre procesos |
| ipc_send_volume | xs:nonNegativeInteger | ipc_send_volume - Volumen de envío de comunicación entre procesos |

Tabla 110. Métricas detalladas devueltas para MON_GET_UNIT_OF_WORK_DETAILS (continuación)

| Nombre de elemento | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|-----------------------|---|
| ipc_send_wait_time | xs:nonNegativeInteger | ipc_send_wait_time - Tiempo de espera de envío de comunicación entre procesos |
| ipc_sends_total | xs:nonNegativeInteger | ipc_sends_total - Total de envíos de comunicación entre procesos |
| lock_escals | xs:nonNegativeInteger | lock_escals - Número de escalamientos de bloqueo |
| lock_timeouts | xs:nonNegativeInteger | lock_timeouts - Número de tiempos de espera de bloqueo |
| lock_wait_time | xs:nonNegativeInteger | lock_wait_time - Tiempo esperado en bloqueos |
| lock_waits | xs:nonNegativeInteger | lock_waits - Esperas de bloqueo |
| log_buffer_wait_time | xs:nonNegativeInteger | log_buffer_wait_time - Tiempo de espera de almacenamiento intermedio de anotaciones cronológicas |
| log_disk_wait_time | xs:nonNegativeInteger | log_disk_wait_time - Tiempo de espera de disco de anotaciones cronológicas |
| log_disk_waits_total | xs:nonNegativeInteger | log_disk_waits_total - Total de esperas de disco de anotaciones cronológicas |
| pool_data_l_reads | xs:nonNegativeInteger | pool_data_l_reads - Lecturas lógicas de datos de la agrupación de almacenamientos intermedios |
| pool_data_p_reads | xs:nonNegativeInteger | pool_data_p_reads - Lecturas físicas de datos de la agrupación de almacenamientos intermedios |
| pool_data_writes | xs:nonNegativeInteger | pool_data_writes - Grabaciones de datos de la agrupación de almacenamientos intermedios |
| pool_index_l_reads | xs:nonNegativeInteger | pool_index_l_reads - Lecturas lógicas de índices de la agrupación de almacenamientos intermedios |
| pool_index_p_reads | xs:nonNegativeInteger | pool_index_p_reads - Lecturas físicas de índices de la agrupación de almacenamientos intermedios |
| pool_index_writes | xs:nonNegativeInteger | pool_index_writes - Grabaciones de índices de la agrupación de almacenamientos intermedios |
| pool_read_time | xs:nonNegativeInteger | pool_read_time - Tiempo total de lectura física de la agrupación de almacenamientos intermedios |
| pool_temp_data_l_reads | xs:nonNegativeInteger | pool_temp_data_l_reads - Lecturas lógicas de datos temporales de la agrupación de almacenamientos intermedios |
| pool_temp_data_p_reads | xs:nonNegativeInteger | pool_temp_data_p_reads - Lecturas físicas de datos temporales de la agrupación de almacenamientos intermedios |
| pool_temp_index_l_reads | xs:nonNegativeInteger | pool_temp_index_l_reads - Lecturas lógicas de índices temporales de la agrupación de almacenamientos intermedios |
| pool_temp_index_p_reads | xs:nonNegativeInteger | pool_temp_index_p_reads - Lecturas físicas de índices temporales de la agrupación de almacenamientos intermedios |
| pool_temp_xda_l_reads | xs:nonNegativeInteger | pool_temp_xda_l_reads - Lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |

Tabla 110. Métricas detalladas devueltas para MON_GET_UNIT_OF_WORK_DETAILS (continuación)

| Nombre de elemento | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------------|-----------------------|---|
| pool_temp_xda_p_reads | xs:nonNegativeInteger | pool_temp_xda_p_reads - Lecturas físicas de datos XDA temporales de agrupación de almacenamientos intermedios |
| pool_write_time | xs:nonNegativeInteger | pool_write_time - Tiempo total de grabación física de agrupación de almacenamientos intermedios |
| pool_xda_l_reads | xs:nonNegativeInteger | pool_xda_l_reads - Lecturas lógicas de datos de XDA de la agrupación de almacenamientos intermedios |
| pool_xda_p_reads | xs:nonNegativeInteger | pool_xda_p_reads - Lecturas físicas de datos de XDA de la agrupación de almacenamientos intermedios |
| pool_xda_writes | xs:nonNegativeInteger | pool_xda_writes - Grabaciones de datos de XDA de la agrupación de almacenamientos intermedios |
| num_log_buffer_full | xs:nonNegativeInteger | num_log_buffer_full - Número de almacenamientos intermedios de anotaciones cronológicas llenos |
| rqsts_completed_total | xs:nonNegativeInteger | rqsts_completed_total - Total de peticiones completadas |
| rows_modified | xs:nonNegativeInteger | rows_modified - Filas modificadas |
| rows_read | xs:nonNegativeInteger | rows_read - Filas leídas |
| rows_returned | xs:nonNegativeInteger | rows_returned - Filas devueltas |
| tcpip_recv_volume | xs:nonNegativeInteger | tcpip_recv_volume - Volumen recibido TCP/IP |
| tcpip_recv_wait_time | xs:nonNegativeInteger | tcpip_recv_wait_time - Tiempo de espera de recepción TCP/IP |
| tcpip_recvs_total | xs:nonNegativeInteger | tcpip_recvs_total - Total de recepciones TCP/IP |
| tcpip_send_volume | xs:nonNegativeInteger | tcpip_send_volume - Volumen de envío TCP/IP |
| tcpip_send_wait_time | xs:nonNegativeInteger | tcpip_send_wait_time - Tiempo de espera de envío TCP/IP |
| tcpip_sends_total | xs:nonNegativeInteger | tcpip_sends_total - Total de envíos TCP/IP |
| total_app_rqst_time | xs:nonNegativeInteger | total_app_rqst_time - Total de tiempo de petición de aplicaciones |
| total_cpu_time | xs:nonNegativeInteger | total_cpu_time - Tiempo total de CPU |
| total_rqst_time | xs:nonNegativeInteger | total_rqst_time - Tiempo total de petición |
| total_section_sort_time | xs:nonNegativeInteger | total_section_sort_time - Tiempo total de clasificación de sección |
| total_section_sort_proc_time | xs:nonNegativeInteger | total_section_sort_proc_time - Tiempo total de proceso de clasificación de sección |
| total_section_sorts | xs:nonNegativeInteger | total_section_sorts - Total de clasificaciones de sección |
| total_sorts | xs:nonNegativeInteger | total_sorts - Número total de clasificaciones |
| post_threshold_sorts | xs:nonNegativeInteger | post_threshold_sorts - Clasificaciones de umbral posteriores |
| post_shrthreshold_sorts | xs:nonNegativeInteger | post_shrthreshold_sorts - Clasificaciones de umbral compartidas posteriores |
| sort_overflows | xs:nonNegativeInteger | sort_overflows - Desbordamientos de clasificación |
| tq_tot_send_spills | xs:nonNegativeInteger | tq_tot_send_spills - Número total de almacenamientos intermedios de colas de tabla desbordados |

Tabla 110. Métricas detalladas devueltas para MON_GET_UNIT_OF_WORK_DETAILS (continuación)

| Nombre de elemento | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------------|-----------------------|---|
| total_wait_time | xs:nonNegativeInteger | total_wait_time - Tiempo total de espera |
| wlm_queue_time_total | xs:nonNegativeInteger | wlm_queue_time_total - Tiempo de cola total del gestor de cargas de trabajo |
| wlm_queue_assignments_total | xs:nonNegativeInteger | wlm_queue_assignments_total - Total de asignaciones de cola del gestor de cargas de trabajo |

Función de tabla MON_GET_WORKLOAD - Obtener métrica de carga de trabajo

La función de tabla MON_GET_WORKLOAD devuelve métricas para una o varias cargas de trabajo.

Sintaxis

►►—MON_GET_WORKLOAD—(—nombre_cargatrabajo—,—miembro—)—————►►

El esquema es SYSPROC.

Parámetros de la función de tabla

nombre_cargatrabajo

Un argumento de entrada de tipo VARCHAR(128) que especifica una carga de trabajo determinada para la que se tienen que devolver métricas. Si el argumento es NULL o una serie vacía, se devuelven métricas para todas las cargas de trabajo.

miembro

Argumento de entrada de tipo INTEGER que especifica un miembro válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique -1 para el miembro de base de datos actual o -2 para todos los miembros de base de datos. Si se especifica el valor NULL, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE sobre la función MON_GET_WORKLOAD.

Ejemplo

Visualizar información de bloqueo para cada carga de trabajo, agregada en todos los miembros, ordenada por tiempo de espera de bloqueo más alto.

```
SELECT varchar(workload_name,30) as workload_name,
       sum(lock_wait_time) as total_lock_wait_time,
       sum(lock_waits) as total_lock_waits,
       sum(lock_timeouts) as total_lock_timeouts,
       sum(lock_escals) as total_lock_escals
FROM TABLE(MON_GET_WORKLOAD(',-2)) AS t
GROUP BY workload_name
ORDER BY total_lock_wait_time DESC
```

A continuación se muestra un ejemplo de salida de esta consulta.

```

WORKLOAD_NAME          TOTAL_LOCK_WAIT_TIME TOTAL_LOCK_WAITS    ...
-----
SYSDEFAULTADMWORKLOAD          0                0 ...
SYSDEFAULTUSERWORKLOAD        0                0 ...

```

2 registro(s) seleccionado(s).

Salida de la consulta (continuación).

```

... TOTAL_LOCK_TIMEOUTS TOTAL_LOCK_ESCALA
... -----
...                0                0
...                0                0

```

Notas de uso

Las métricas devueltas por la función `MON_GET_WORKLOAD` representan la suma de todas las métricas correspondientes a las peticiones que enviaron las conexiones correlacionadas con el objeto de carga de trabajo identificado. Las métricas se avanzan a una carga de trabajo cuando llegan a los límites de la unidad de trabajo, y periódicamente durante la ejecución de las peticiones. Por consiguiente, los valores notificados por esta función de tabla reflejan el estado actual del sistema en el momento del avance más reciente. Las métricas son estrictamente de valor creciente. Para determinar el valor de una métrica dada durante un intervalo de tiempo, utilice la función de tabla `MON_GET_WORKLOAD` para consultar la métrica al principio y al final del intervalo y calcular la diferencia.

Las métricas de petición se controlan mediante la cláusula `COLLECT REQUEST METRICS` en las superclases de servicio y el parámetro de configuración de base de datos `mon_req_metrics` en el nivel de la base de datos. Las métricas únicamente se recopilan para una petición si la petición se procesa a través de un agente en una subclase de servicio cuya superclase de servicio padre tiene habilitadas las métricas de petición o si la recopilación de métricas de petición está habilitada para toda la base de datos. Por omisión, las métricas de petición están habilitadas en el nivel de la base de datos. Si se han inhabilitado las métricas de petición en el nivel de la base de datos y para una superclase de servicio, las métricas notificadas para cada carga de trabajo correlacionada con dicha superclase de servicio dejan de aumentar (o permanecen en 0 si las métricas de petición estaban inhabilitadas en el momento de la activación de la base de datos).

La función de tabla `MON_GET_WORKLOAD` devuelve una fila de datos por carga de trabajo y por miembro. No se realiza ninguna agregación entre cargas de trabajo (de un miembro) ni entre miembros (correspondientes a una o varias clases de servicio). Sin embargo, se puede lograr la agregación mediante consultas de SQL (véase el ejemplo).

Información devuelta

Tabla 111. Información devuelta para `MON_GET_WORKLOAD`

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------------------|---------------------------|---|
| <code>WORKLOAD_NAME</code> | <code>VARCHAR(128)</code> | <code>workload_name</code> - Nombre de carga de trabajo |
| <code>WORKLOAD_ID</code> | <code>INTEGER</code> | <code>workload_id</code> - ID de carga de trabajo |
| <code>MEMBER</code> | <code>SMALLINT</code> | <code>member</code> - Miembro de base de datos |
| <code>ACT_ABORTED_TOTAL</code> | <code>BIGINT</code> | <code>act_aborted_total</code> - Total de actividades abortadas |

Tabla 111. Información devuelta para MON_GET_WORKLOAD (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|--|
| ACT_COMPLETED_TOTAL | BIGINT | act_completed_total - Total de actividades completadas |
| ACT_REJECTED_TOTAL | BIGINT | act_rejected_total - Total de actividades rechazadas |
| AGENT_WAIT_TIME | BIGINT | agent_wait_time - Tiempo de espera del agente |
| AGENT_WAITS_TOTAL | BIGINT | agent_waits_total - Total de esperas del agente |
| POOL_DATA_L_READS | BIGINT | pool_data_l_reads - Lecturas lógicas de datos de la agrupación de almacenamientos intermedios |
| POOL_INDEX_L_READS | BIGINT | pool_index_l_reads - Lecturas lógicas de índices de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_L_READS | BIGINT | pool_temp_data_l_reads - Lecturas lógicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_L_READS | BIGINT | pool_temp_index_l_reads - Lecturas lógicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_L_READS | BIGINT | pool_temp_xda_l_reads - Lecturas lógicas de datos XDA temporales de agrupación de almacenamientos intermedios |
| POOL_XDA_L_READS | BIGINT | pool_xda_l_reads - Lecturas lógicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_DATA_P_READS | BIGINT | pool_data_p_reads - Lecturas físicas de datos de la agrupación de almacenamientos intermedios |
| POOL_INDEX_P_READS | BIGINT | pool_index_p_reads - Lecturas físicas de índices de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_P_READS | BIGINT | pool_temp_data_p_reads - Lecturas físicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_P_READS | BIGINT | pool_temp_index_p_reads - Lecturas físicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_P_READS | BIGINT | pool_temp_xda_p_reads - Lecturas físicas de datos XDA temporales de agrupación de almacenamientos intermedios |
| POOL_XDA_P_READS | BIGINT | pool_xda_p_reads - Lecturas físicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_DATA_WRITES | BIGINT | pool_data_writes - Grabaciones de datos de la agrupación de almacenamientos intermedios |
| POOL_INDEX_WRITES | BIGINT | pool_index_writes - Grabaciones de índices de la agrupación de almacenamientos intermedios |
| POOL_XDA_WRITES | BIGINT | pool_xda_writes - Grabaciones de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_READ_TIME | BIGINT | pool_read_time - Tiempo total de lectura física de la agrupación de almacenamientos intermedios |
| POOL_WRITE_TIME | BIGINT | pool_write_time - Tiempo total de grabación física de agrupación de almacenamientos intermedios |

Tabla 111. Información devuelta para MON_GET_WORKLOAD (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------|---------------|--|
| CLIENT_IDLE_WAIT_TIME | BIGINT | client_idle_wait_time - Tiempo de espera de inactividad del cliente |
| DEADLOCKS | BIGINT | deadlocks - Puntos muertos detectados |
| DIRECT_READS | BIGINT | direct_reads - Lecturas directas de la base de datos |
| DIRECT_READ_TIME | BIGINT | direct_read_time - Tiempo de lectura directa |
| DIRECT_WRITES | BIGINT | direct_writes - Grabaciones directas en la base de datos |
| DIRECT_WRITE_TIME | BIGINT | direct_write_time - Tiempo de grabación directa |
| DIRECT_READ_REQS | BIGINT | direct_read_reqs - Peticiones de lectura directa |
| DIRECT_WRITE_REQS | BIGINT | direct_write_reqs - Peticiones de grabación directa |
| FCM_RECV_VOLUME | BIGINT | fcm_recv_volume - Volumen de recepción de FCM |
| FCM_RECVS_TOTAL | BIGINT | fcm_recvs_total - Total de recepciones de FCM |
| FCM_SEND_VOLUME | BIGINT | fcm_send_volume - Volumen de envío de FCM |
| FCM_SENDS_TOTAL | BIGINT | fcm_sends_total - Total de envíos de FCM |
| FCM_RECV_WAIT_TIME | BIGINT | fcm_recv_wait_time - Tiempo de espera de recepción de FCM |
| FCM_SEND_WAIT_TIME | BIGINT | fcm_send_wait_time - Tiempo de espera de envío de FCM |
| IPC_RECV_VOLUME | BIGINT | ipc_recv_volume - Volumen de recepción de comunicación entre procesos |
| IPC_RECV_WAIT_TIME | BIGINT | ipc_recv_wait_time - Tiempo de espera de recepción de comunicación entre procesos |
| IPC_RECVS_TOTAL | BIGINT | ipc_recvs_total - Total de recepciones de comunicación entre procesos |
| IPC_SEND_VOLUME | BIGINT | ipc_send_volume - Volumen de envío de comunicación entre procesos |
| IPC_SEND_WAIT_TIME | BIGINT | ipc_send_wait_time - Tiempo de espera de envío de comunicación entre procesos |
| IPC_SENDS_TOTAL | BIGINT | ipc_sends_total - Total de envíos de comunicación entre procesos |
| LOCK_ESCALS | BIGINT | lock_escals - Número de escalamientos de bloqueo |
| LOCK_TIMEOUTS | BIGINT | lock_timeouts - Número de tiempos de espera de bloqueo |
| LOCK_WAIT_TIME | BIGINT | lock_wait_time - Tiempo esperado en bloqueos |
| LOCK_WAITS | BIGINT | lock_waits - Esperas de bloqueo |
| LOG_BUFFER_WAIT_TIME | BIGINT | log_buffer_wait_time - Tiempo de espera de almacenamiento intermedio de anotaciones cronológicas |
| NUM_LOG_BUFFER_FULL | BIGINT | num_log_buffer_full - Número de almacenamientos intermedios de anotaciones cronológicas llenos |
| LOG_DISK_WAIT_TIME | BIGINT | log_disk_wait_time - Tiempo de espera de disco de anotaciones cronológicas |
| LOG_DISK_WAITS_TOTAL | BIGINT | log_disk_waits_total - Total de esperas de disco de anotaciones cronológicas |

Tabla 111. Información devuelta para MON_GET_WORKLOAD (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|----------------------------------|---------------|---|
| RQSTS_COMPLETED_TOTAL | BIGINT | rqsts_completed_total - Total de peticiones completadas |
| ROWS_MODIFIED | BIGINT | rows_modified - Filas modificadas |
| ROWS_READ | BIGINT | rows_read - Filas leídas |
| ROWS_RETURNED | BIGINT | rows_returned - Filas devueltas |
| TCPIP_RECV_VOLUME | BIGINT | tcPIP_recv_volume - Volumen recibido TCP/IP |
| TCPIP_SEND_VOLUME | BIGINT | tcPIP_send_volume - Volumen de envío TCP/IP |
| TCPIP_RECV_WAIT_TIME | BIGINT | tcPIP_recv_wait_time - Tiempo de espera de recepción TCP/IP |
| TCPIP_RECVS_TOTAL | BIGINT | tcPIP_recvs_total - Total de recepciones TCP/IP |
| TCPIP_SEND_WAIT_TIME | BIGINT | tcPIP_send_wait_time - Tiempo de espera de envío TCP/IP |
| TCPIP_SENDS_TOTAL | BIGINT | tcPIP_sends_total - Total de envíos TCP/IP |
| TOTAL_APP_RQST_TIME | BIGINT | total_app_rqst_time - Total de tiempo de petición de aplicaciones |
| TOTAL_RQST_TIME | BIGINT | total_rqst_time - Tiempo total de petición |
| WLM_QUEUE_TIME_TOTAL | BIGINT | wlm_queue_time_total - Tiempo de cola total del gestor de cargas de trabajo |
| _TOTAL | BIGINT | wlm_queue_assignments_total - Total de asignaciones de cola del gestor de cargas de trabajo |
| TOTAL_CPU_TIME | BIGINT | total_cpu_time - Tiempo total de CPU |
| TOTAL_WAIT_TIME | BIGINT | total_wait_time - Tiempo total de espera |
| APP_RQSTS_COMPLETED_TOTAL | BIGINT | app_rqsts_completed_total - Total de peticiones de aplicación completadas |
| TOTAL_SECTION_SORT_TIME | BIGINT | total_section_sort_time - Tiempo total de clasificación de sección |
| TOTAL_SECTION_SORT_PROC_TIME | BIGINT | total_section_sort_proc_time - Tiempo total de proceso de clasificación de sección |
| TOTAL_SECTION_SORTS | BIGINT | total_section_sorts - Total de clasificaciones de sección |
| TOTAL_SORTS | BIGINT | total_sorts - Número total de clasificaciones |
| POST_THRESHOLD_SORTS | BIGINT | post_threshold_sorts - Clasificaciones de umbral posteriores |
| POST_SHRTHRESHOLD_SORTS | BIGINT | post_shrthreshold_sorts - Clasificaciones de umbral compartidas posteriores |
| SORT_OVERFLOWS | BIGINT | sort_overflows - Desbordamientos de clasificación |
| TOTAL_COMPILE_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_COMPILE_PROC_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_COMPILATIONS | BIGINT | Reservado para una utilización futura. |
| TOTAL_IMPLICIT_COMPILE_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_IMPLICIT_COMPILE_PROC_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_IMPLICIT_COMPILATIONS | BIGINT | Reservado para una utilización futura. |

Tabla 111. Información devuelta para MON_GET_WORKLOAD (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------------|---------------|---|
| TOTAL_SECTION_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_SECTION_PROC_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_APP_SECTION_EXECUTIONS | BIGINT | Reservado para una utilización futura. |
| TOTAL_ACT_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_ACT_WAIT_TIME | BIGINT | Reservado para una utilización futura. |
| ACT_RQSTS_TOTAL | BIGINT | Reservado para una utilización futura. |
| TOTAL_ROUTINE_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_ROUTINE_INVOCATIONS | BIGINT | Reservado para una utilización futura. |
| TOTAL_COMMIT_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_COMMIT_PROC_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_APP_COMMITS | BIGINT | Reservado para una utilización futura. |
| INT_COMMITS | BIGINT | Reservado para una utilización futura. |
| TOTAL_ROLLBACK_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_ROLLBACK_PROC_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_APP_ROLLBACKS | BIGINT | Reservado para una utilización futura. |
| INT_ROLLBACKS | BIGINT | Reservado para una utilización futura. |
| TOTAL_RUNSTATS_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_RUNSTATS_PROC_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_RUNSTATS | BIGINT | Reservado para una utilización futura. |
| TOTAL_REORG_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_REORG_PROC_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_REORGS | BIGINT | Reservado para una utilización futura. |
| TOTAL_LOAD_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_LOAD_PROC_TIME | BIGINT | Reservado para una utilización futura. |
| TOTAL_LOADS | BIGINT | Reservado para una utilización futura. |
| CAT_CACHE_INSERTS | BIGINT | Reservado para una utilización futura. |
| CAT_CACHE_LOOKUPS | BIGINT | Reservado para una utilización futura. |
| PKG_CACHE_INSERTS | BIGINT | Reservado para una utilización futura. |
| PKG_CACHE_LOOKUPS | BIGINT | Reservado para una utilización futura. |
| THRESH_VIOLATIONS | BIGINT | Reservado para una utilización futura. |
| NUM_LW_THRESH_EXCEEDED | BIGINT | Reservado para una utilización futura. |
| ADDITIONAL_DETAILS | BLOB(100K) | Reservado para una utilización futura. |

Función de tabla MON_GET_WORKLOAD_DETAILS - Obtener métrica detallada de carga de trabajo

La función de tabla MON_GET_WORKLOAD_DETAILS devuelve métricas detalladas para una o varias cargas de trabajo.

Sintaxis

►►—MON_GET_WORKLOAD_DETAILS—(—nombre_cargatrabajo—,—miembro—)————►►

El esquema es SYSPROC.

Parámetros de la función de tabla

nombre_cargatrabajo

Un argumento de entrada de tipo VARCHAR(128) que especifica una carga de trabajo determinada para la que se tienen que devolver métricas. Si el argumento es NULL o una serie vacía, se devuelven métricas para todas las cargas de trabajo.

miembro

Argumento de entrada de tipo INTEGER que especifica un miembro válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique -1 para el miembro de base de datos actual o -2 para todos los miembros de base de datos. Si se especifica el valor NULL, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE sobre la función MON_GET_WORKLOAD_DETAILS.

Ejemplo

Visualizar información de bloqueo para cada carga de trabajo, agregada en todos los miembros, ordenada por tiempo de espera de bloqueo más alto.

```
SELECT varchar(wlmetrics.workload_name,30) as workload_name,
       sum(detmetrics.lock_wait_time) as total_lock_wait_time,
       sum(detmetrics.lock_waits) as total_lock_waits,
       sum(detmetrics.lock_timeouts) as total_lock_timeouts,
       sum(detmetrics.lock_escals) as total_lock_escals
FROM TABLE(MON_GET_WORKLOAD_DETAILS('','-2')) AS WLMETRICS,
XMLTABLE (XMLNAMESPACES( DEFAULT 'http://www.ibm.com/xmlns/prod/db2/mon'),
          '$detmetric/db2_workload' PASSING
          XMLPARSE(DOCUMENT WLMETRICS.DETAILS)
          as "detmetric"
COLUMNS "LOCK_WAIT_TIME" INTEGER PATH 'system_metrics/lock_wait_time',
         "LOCK_WAITS" INTEGER PATH 'system_metrics/lock_waits',
         "LOCK_TIMEOUTS" INTEGER PATH 'system_metrics/lock_timeouts',
         "LOCK_ESCALS" INTEGER PATH 'system_metrics/lock_escals'
) AS DETMETRICS
GROUP BY workload_name
ORDER BY total_lock_wait_time desc;
```

A continuación se muestra un ejemplo de salida de esta consulta.

| WORKLOAD_NAME | TOTAL_LOCK_WAIT_TIME | TOTAL_LOCK_WAITS | ... |
|------------------------|----------------------|------------------|-----|
| SYSDEFAULTADMWORKLOAD | 0 | 0 | ... |
| SYSDEFAULTUSERWORKLOAD | 0 | 0 | ... |

2 registro(s) seleccionado(s).

Salida de la consulta (continuación).

```

... TOTAL_LOCK_TIMEOUTS TOTAL_LOCK_ESCALS
... -----
... 0 0
... 0 0

```

Notas de uso

Las métricas devueltas por la función `MON_GET_WORKLOAD_DETAILS` representan la suma de todas las métricas correspondientes a las peticiones que enviaron las conexiones correlacionadas con el objeto de carga de trabajo identificado. Esta función es parecida a la función de tabla `MON_GET_WORKLOAD`:

- La función de tabla `MON_GET_WORKLOAD` devuelve las métricas utilizadas con más frecuencia en un formato basado en columnas, y es el método más eficiente, desde el punto de vista del rendimiento, de recuperar las métricas.
- La función de tabla `MON_GET_WORKLOAD_DETAILS` devuelve todo el conjunto de métricas disponibles en formato de documento XML, lo que proporciona la máxima flexibilidad para dar formato a la salida. La salida basada en XML puede analizarse directamente mediante un analizador XML, o puede convertirse a formato relacional mediante la función `XMLTABLE` (véase el ejemplo).

Las métricas se avanzan a una carga de trabajo cuando llegan a los límites de la unidad de trabajo, y periódicamente durante la ejecución de las peticiones. Por consiguiente, los valores notificados por esta función de tabla reflejan el estado actual del sistema en el momento del avance más reciente. Las métricas son estrictamente de valor creciente. Para determinar el valor de una métrica dada durante un intervalo de tiempo, utilice la función de tabla `MON_GET_WORKLOAD_DETAILS` para consultar la métrica al principio y al final del intervalo y calcular la diferencia.

Las métricas de petición se controlan mediante la cláusula `COLLECT REQUEST METRICS` en las superclases de servicio y el parámetro de configuración de base de datos `mon_req_metrics` en el nivel de la base de datos. Las métricas únicamente se recopilan para una petición si la petición se procesa a través de un agente en una subclase de servicio cuya superclase de servicio padre tiene habilitadas las métricas de petición o si la recopilación de métricas de petición está habilitada para toda la base de datos. Por omisión, las métricas de petición están habilitadas en el nivel de la base de datos. Si se han inhabilitado las métricas de petición en el nivel de la base de datos y para una superclase de servicio, las métricas notificadas para cada carga de trabajo correlacionada con dicha superclase de servicio dejan de aumentar (o permanecen en 0 si las métricas de petición estaban inhabilitadas en el momento de la activación de la base de datos).

La función de tabla `MON_GET_WORKLOAD_DETAILS` devuelve una fila de datos por carga de trabajo y por miembro. No se realiza ninguna agregación entre cargas de trabajo (de un miembro) ni entre miembros (correspondientes a una o varias clases de servicio). Sin embargo, se puede lograr la agregación mediante consultas de SQL, tal como se muestra en el ejemplo.

El esquema para el documento XML devuelto en la columna `DETAILS` está disponible en el archivo `sqllib/misc/DB2MonRoutines.xsd`. Para obtener más información, consulte el archivo `sqllib/misc/DB2MonCommon.xsd`.

Información devuelta

Tabla 112. Información devuelta para MON_GET_WORKLOAD_DETAILS

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| WORKLOAD_NAME | VARCHAR(128) | workload_name - Nombre de carga de trabajo |
| WORKLOAD_ID | INTEGER | workload_id - ID de carga de trabajo |
| MEMBER | SMALLINT | member - Miembro de base de datos |
| DETAILS | BLOB(1M) | Documento XML que contiene las métricas detalladas de la carga de trabajo. Consulte la Tabla 113 para ver una descripción de los elementos de este documento. |

El ejemplo siguiente muestra la estructura del documento XML que se devuelve en la columna DETAILS.

```
<db2_workload xmlns="http://www.ibm.com/xmlns/prod/db2/mon" release="90700000">
  <workload_name>SYSDEFAULTADMWORKLOAD</workload_name>
  <workload_id>11</workload_id>
  <member>0</member>
  <system_metrics release="90700000">
    <act_aborted_total>5</act_aborted_total>
    ...
    <wlm_queue_assignments_total>3</wlm_queue_assignments_total>
  </system_metrics>
</db2_workload>
```

Para ver el esquema completo, consulte `sql1lib/misc/DB2MonRoutines.xsd`.

Tabla 113. Métricas detalladas devueltas para MON_GET_WORKLOAD_DETAILS

| Nombre de elemento | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------------|-----------------------|---|
| workload_name | xs:string (128) | workload_name - Nombre de carga de trabajo |
| workload_id | xs:nonNegativeInteger | workload_id - ID de carga de trabajo |
| member | xs:nonNegativeInteger | member - Miembro de base de datos |
| act_aborted_total | xs:nonNegativeInteger | act_aborted_total - Total de actividades abortadas |
| act_completed_total | xs:nonNegativeInteger | act_completed_total - Total de actividades completadas |
| act_rejected_total | xs:nonNegativeInteger | act_rejected_total - Total de actividades rechazadas |
| agent_wait_time | xs:nonNegativeInteger | agent_wait_time - Tiempo de espera del agente |
| agent_waits_total | xs:nonNegativeInteger | agent_waits_total - Total de esperas del agente |
| app_rqsts_completed_total | xs:nonNegativeInteger | app_rqsts_completed_total - Total de peticiones de aplicación completadas |
| audit_events_total | xs:nonNegativeInteger | audit_events_total - Total de sucesos de auditoría |
| audit_subsystem_wait_time | xs:nonNegativeInteger | audit_subsystem_wait_time - Tiempo de espera del subsistema de auditoría |
| audit_subsystem_waits_total | xs:nonNegativeInteger | audit_subsystem_waits_total - Total de esperas del subsistema de auditoría |
| audit_file_write_wait_time | xs:nonNegativeInteger | audit_file_write_wait_time - Tiempo de espera de grabación del archivo de auditoría |
| audit_file_writes_total | xs:nonNegativeInteger | audit_file_writes_total - Total de archivos de auditoría grabados |
| client_idle_wait_time | xs:nonNegativeInteger | client_idle_wait_time - Tiempo de espera de inactividad del cliente |
| deadlocks | xs:nonNegativeInteger | deadlocks - Puntos muertos detectados |

Tabla 113. Métricas detalladas devueltas para MON_GET_WORKLOAD_DETAILS (continuación)

| Nombre de elemento | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|----------------------------|-----------------------|--|
| diaglog_writes_total | xs:nonNegativeInteger | diaglog_writes_total - Total de grabaciones de anotaciones cronológicas de diagnóstico |
| diaglog_write_wait_time | xs:nonNegativeInteger | diaglog_write_wait_time - Tiempo de grabación de anotaciones cronológicas de diagnóstico |
| direct_read_time | xs:nonNegativeInteger | direct_read_time - Tiempo de lectura directa |
| direct_write_time | xs:nonNegativeInteger | direct_write_time - Tiempo de grabación directa |
| direct_read_reqs | xs:nonNegativeInteger | direct_read_reqs - Peticiones de lectura directa |
| direct_reads | xs:nonNegativeInteger | direct_reads - Lecturas directas de la base de datos |
| direct_write_reqs | xs:nonNegativeInteger | direct_write_reqs - Peticiones de grabación directa |
| direct_writes | xs:nonNegativeInteger | direct_writes - Grabaciones directas en la base de datos |
| fcm_recv_volume | xs:nonNegativeInteger | fcm_recv_volume - Volumen de recepción de FCM |
| fcm_recv_wait_time | xs:nonNegativeInteger | fcm_recv_wait_time - Tiempo de espera de recepción de FCM |
| fcm_recvs_total | xs:nonNegativeInteger | fcm_recvs_total - Total de recepciones de FCM |
| fcm_message_recv_volume | xs:nonNegativeInteger | fcm_message_recv_volume - Volumen de recepción de mensaje de FCM |
| fcm_message_recvs_total | xs:nonNegativeInteger | fcm_message_recvs_total - Total de recepciones de mensaje de FCM |
| fcm_message_recv_wait_time | xs:nonNegativeInteger | fcm_message_recv_wait_time - Tiempo de espera de recepción de mensaje de FCM |
| fcm_message_send_volume | xs:nonNegativeInteger | fcm_message_send_volume - Volumen de envío de mensaje de FCM |
| fcm_message_send_wait_time | xs:nonNegativeInteger | fcm_message_send_wait_time - Tiempo de espera de envío de mensaje de FCM |
| fcm_message_sends_total | xs:nonNegativeInteger | fcm_message_sends_total - Total de envíos de mensaje de FCM |
| fcm_send_volume | xs:nonNegativeInteger | fcm_send_volume - Volumen de envío de FCM |
| fcm_send_wait_time | xs:nonNegativeInteger | fcm_send_wait_time - Tiempo de espera de envío de FCM |
| fcm_sends_total | xs:nonNegativeInteger | fcm_sends_total - Total de envíos de FCM |
| fcm_tq_recv_wait_time | xs:nonNegativeInteger | fcm_tq_recv_wait_time - Tiempo de espera de recepción de cola de tabla de FCM |
| fcm_tq_send_wait_time | xs:nonNegativeInteger | fcm_tq_send_wait_time - Tiempo de espera de envío de cola de tabla de FCM |
| fcm_tq_recv_volume | xs:nonNegativeInteger | fcm_tq_recv_volume - Volumen de recepción de cola de tabla de FCM |
| fcm_tq_recvs_total | xs:nonNegativeInteger | fcm_tq_recvs_total - Total de recepciones de cola de tabla de FCM |
| fcm_tq_send_volume | xs:nonNegativeInteger | fcm_tq_send_volume - Volumen de envíos de cola de tabla de FCM |
| fcm_tq_sends_total | xs:nonNegativeInteger | fcm_tq_sends_total - Total de envíos de cola de tabla de FCM |
| tq_tot_send_spills | xs:nonNegativeInteger | tq_tot_send_spills - Total de almacenamientos intermedios de cola de tabla desbordados |

Tabla 113. Métricas detalladas devueltas para MON_GET_WORKLOAD_DETAILS (continuación)

| Nombre de elemento | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|-----------------------|---|
| ipc_rcv_volume | xs:nonNegativeInteger | ipc_rcv_volume - Volumen de recepción de comunicación entre procesos |
| ipc_rcv_wait_time | xs:nonNegativeInteger | ipc_rcv_wait_time - Tiempo de espera de recepción de comunicación entre procesos |
| ipc_rcvs_total | xs:nonNegativeInteger | ipc_rcvs_total - Total de recepciones de comunicación entre procesos |
| ipc_send_volume | xs:nonNegativeInteger | ipc_send_volume - Volumen de envío de comunicación entre procesos |
| ipc_send_wait_time | xs:nonNegativeInteger | ipc_send_wait_time - Tiempo de espera de envío de comunicación entre procesos |
| ipc_sends_total | xs:nonNegativeInteger | ipc_sends_total - Total de envíos de comunicación entre procesos |
| lock_escals | xs:nonNegativeInteger | lock_escals - Número de escalamientos de bloqueo |
| lock_timeouts | xs:nonNegativeInteger | lock_timeouts - Número de tiempos de espera de bloqueo |
| lock_wait_time | xs:nonNegativeInteger | lock_wait_time - Tiempo esperado en bloqueos |
| lock_waits | xs:nonNegativeInteger | lock_waits - Esperas de bloqueo |
| log_buffer_wait_time | xs:nonNegativeInteger | log_buffer_wait_time - Tiempo de espera de almacenamiento intermedio de anotaciones cronológicas |
| log_disk_wait_time | xs:nonNegativeInteger | log_disk_wait_time - Tiempo de espera de disco de anotaciones cronológicas |
| log_disk_waits_total | xs:nonNegativeInteger | log_disk_waits_total - Total de esperas de disco de anotaciones cronológicas |
| pool_data_l_reads | xs:nonNegativeInteger | pool_data_l_reads - Lecturas lógicas de datos de la agrupación de almacenamientos intermedios |
| pool_data_p_reads | xs:nonNegativeInteger | pool_data_p_reads - Lecturas físicas de datos de la agrupación de almacenamientos intermedios |
| pool_data_writes | xs:nonNegativeInteger | pool_data_writes - Grabaciones de datos de la agrupación de almacenamientos intermedios |
| pool_index_l_reads | xs:nonNegativeInteger | pool_index_l_reads - Lecturas lógicas de índice de agrupación de almacenamientos intermedios |
| pool_index_p_reads | xs:nonNegativeInteger | pool_index_p_reads - Lecturas físicas de índice de agrupación de almacenamientos intermedios |
| pool_index_writes | xs:nonNegativeInteger | pool_index_writes - Grabaciones de índice de agrupación de almacenamientos intermedios |
| pool_read_time | xs:nonNegativeInteger | pool_read_time - Tiempo total de lectura física de agrupación de almacenamientos intermedios |
| pool_temp_data_l_reads | xs:nonNegativeInteger | pool_temp_data_l_reads - Lecturas lógicas de datos temporales de agrupación de almacenamientos intermedios |
| pool_temp_data_p_reads | xs:nonNegativeInteger | pool_temp_data_p_reads - Lecturas físicas de datos temporales de la agrupación de almacenamientos intermedios |
| pool_temp_index_l_reads | xs:nonNegativeInteger | pool_temp_index_l_reads - Lecturas lógicas de índice temporal de agrupación de almacenamientos intermedios |

Tabla 113. Métricas detalladas devueltas para MON_GET_WORKLOAD_DETAILS (continuación)

| Nombre de elemento | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------------|-----------------------|---|
| pool_temp_index_p_reads | xs:nonNegativeInteger | pool_temp_index_p_reads - Lecturas físicas de índices temporales de la agrupación de almacenamientos intermedios |
| pool_temp_xda_l_reads | xs:nonNegativeInteger | pool_temp_xda_l_reads - Lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |
| pool_temp_xda_p_reads | xs:nonNegativeInteger | pool_temp_xda_p_reads - Lecturas físicas de datos XDA temporales de agrupación de almacenamientos intermedios |
| pool_write_time | xs:nonNegativeInteger | pool_write_time - Tiempo total de grabación física de agrupación de almacenamientos intermedios |
| pool_xda_l_reads | xs:nonNegativeInteger | pool_xda_l_reads - Lecturas lógicas de datos de XDA de la agrupación de almacenamientos intermedios |
| pool_xda_p_reads | xs:nonNegativeInteger | pool_xda_p_reads - Lecturas físicas de datos de XDA de la agrupación de almacenamientos intermedios |
| pool_xda_writes | xs:nonNegativeInteger | pool_xda_writes - Grabaciones de datos de XDA de la agrupación de almacenamientos intermedios |
| num_log_buffer_full | xs:nonNegativeInteger | num_log_buffer_full - Número de almacenamientos intermedios de anotaciones cronológicas llenos |
| rqsts_completed_total | xs:nonNegativeInteger | rqsts_completed_total - Total de peticiones completadas |
| rows_modified | xs:nonNegativeInteger | rows_modified - Filas modificadas |
| rows_read | xs:nonNegativeInteger | rows_read - Filas leídas |
| rows_returned | xs:nonNegativeInteger | rows_returned - Filas devueltas |
| tcpip_rcv_volume | xs:nonNegativeInteger | tcpip_rcv_volume - Volumen recibido TCP/IP |
| tcpip_rcv_wait_time | xs:nonNegativeInteger | tcpip_rcv_wait_time - Tiempo de espera de recepción TCP/IP |
| tcpip_rcvs_total | xs:nonNegativeInteger | tcpip_rcvs_total - Total de recepciones TCP/IP |
| tcpip_send_volume | xs:nonNegativeInteger | tcpip_send_volume - Volumen de envío TCP/IP |
| tcpip_send_wait_time | xs:nonNegativeInteger | tcpip_send_wait_time - Tiempo de espera de envío TCP/IP |
| tcpip_sends_total | xs:nonNegativeInteger | tcpip_sends_total - Total de envíos TCP/IP |
| total_app_rqst_time | xs:nonNegativeInteger | total_app_rqst_time - Total de tiempo de petición de aplicaciones |
| total_cpu_time | xs:nonNegativeInteger | total_cpu_time - Tiempo total de CPU |
| total_rqst_time | xs:nonNegativeInteger | total_rqst_time - Tiempo total de petición |
| total_section_sort_time | xs:nonNegativeInteger | total_section_sort_time - Tiempo total de clasificación de sección |
| total_section_sort_proc_time | xs:nonNegativeInteger | total_section_sort_proc_time - Tiempo total de proceso de clasificación de sección |
| total_section_sorts | xs:nonNegativeInteger | total_section_sorts - Total de clasificaciones de sección |
| total_sorts | xs:nonNegativeInteger | total_sorts - Número total de clasificaciones |
| post_threshold_sorts | xs:nonNegativeInteger | post_threshold_sorts - Clasificaciones de umbral posteriores |
| post_shrthreshold_sorts | xs:nonNegativeInteger | post_shrthreshold_sorts - Clasificaciones de umbral compartidas posteriores |

Tabla 113. Métricas detalladas devueltas para MON_GET_WORKLOAD_DETAILS (continuación)

| Nombre de elemento | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------------|-----------------------|--|
| sort_overflows | xs:nonNegativeInteger | sort_overflows - Desbordamientos de clasificación |
| tq_tot_send_spills | xs:nonNegativeInteger | tq_tot_send_spills - Número total de almacenamientos intermedios de colas de tabla desbordados |
| total_wait_time | xs:nonNegativeInteger | total_wait_time - Tiempo total de espera |
| wlm_queue_time_total | xs:nonNegativeInteger | wlm_queue_time_total - Tiempo de cola total del gestor de cargas de trabajo |
| wlm_queue_assignments_total | xs:nonNegativeInteger | wlm_queue_assignments_total - Total de asignaciones de cola del gestor de cargas de trabajo |

Capítulo 12. Rutinas de MQSeries

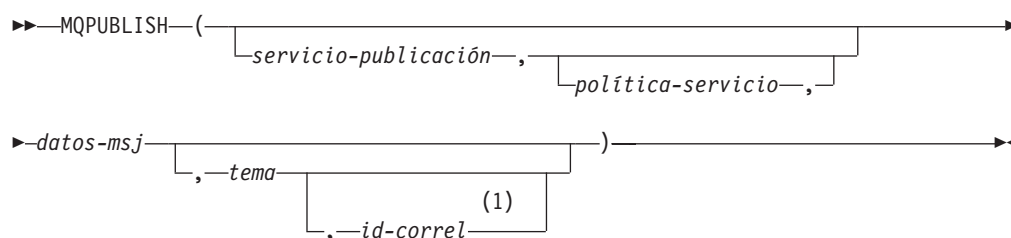
MQPUBLISH

La función MQPUBLISH publica datos en MQSeries. Para obtener más detalles, visite <http://www.ibm.com/software/MQSeries>.

La función MQPUBLISH publica los datos contenidos en *datos-msj* en el publicador de MQSeries especificado en *servicio-publicación*, utilizando la política de calidad de la política de servicio definida por *política-servicio*. Se puede especificar un tema opcional para el mensaje y también se puede especificar un identificador de correlación de mensaje opcional definido por el usuario.

El tipo de datos del resultado es VARCHAR(1). El resultado de la función es '1' si es satisfactorio o cero ('0') si no es satisfactorio.

Sintaxis



Notas:

- 1 El *id-correl* no se puede especificar a no ser que se haya definido un *servicio* y una *política*.

El esquema es DB2MQ para funciones de cola de mensajes de no transacción, y DB2MQ1C para funciones transaccionales de MQ de comprometer en una fase.

Parámetros de la función

servicio-publicación

Serie que contiene el destino MQSeries lógico donde se debe enviar el mensaje. Si se especifica, el *servicio-publicación* debe hacer referencia a un punto de servicio de publicación definido en la tabla DB2MQ.MQPUBSUB que tiene el valor de tipo 'P' para servicio de publicación. Si no se especifica *servicio-publicación*, se utilizará DB2.DEFAULT.PUBLISHER. El tamaño máximo de *servicio-publicación* es de 48 bytes.

política-servicio

Serie que contiene la política de servicio de MQSeries que se debe utilizar en el manejo de este mensaje. Si se especifica, la *política-servicio* debe hacer referencia a una política definida en la tabla DB2MQ.MQPOLICY. Una Política de servicio define un conjunto de opciones de calidad de servicio que deben aplicarse a esta operación de mensajería. Estas opciones incluyen la prioridad de mensaje y la permanencia de mensaje. Si no se especifica *política-servicio*, se utilizará el valor DB2.DEFAULT.POLICY por omisión. El tamaño máximo de *política-servicio* es de 48 bytes.

datos-msj

Expresión de serie que contiene los datos que se deben enviar a través de MQSeries. El tamaño máximo para una expresión de serie VARCHAR es de 32.000 bytes, mientras que para una expresión de serie CLOB es de 1 M bytes.

tema

Expresión de serie que contiene el tema para la publicación del mensaje. Si no se especifica ningún tema, no se asociará ninguno con el mensaje. El tamaño máximo de *tema* es de 40 bytes. Se pueden especificar múltiples temas en una serie (de una longitud máxima de 40). Cada tema debe estar separado por dos puntos. Por ejemplo, "t1:t2:el tercer tema" indica que el mensaje está asociado con los tres temas: t1, t2 y "el tercer tema".

id-correl

Expresión de serie opcional que contiene un identificador de correlación que se debe asociar con este mensaje. El *id-correl* se especifica normalmente en escenarios de petición y respuesta para asociar las peticiones con las respuestas. Si no se especifica, no se añadirá ningún ID de correlación al mensaje. El tamaño máximo de *id-correl* es de 24 bytes.

Ejemplos

Ejemplo 1: En este ejemplo se publica la serie "Testing 123" en el servicio de publicación por omisión (DB2.DEFAULT.PUBLISHER) utilizando la política por omisión (DB2.DEFAULT.POLICY). No se especifica ningún identificador de correlación ni tema para el mensaje.

```
VALUES MQPUBLISH('Testing 123')
```

Ejemplo 2: En este ejemplo se publica la serie "Testing 345" en el servicio de publicación "MYPUBLISHER" bajo el tema "TESTS". Se utiliza la política por omisión y no se especifica ningún identificador de correlación.

```
VALUES MQPUBLISH('MYPUBLISHER','Testing 345','TESTS')
```

Ejemplo 3: En este ejemplo se publica la serie "Testing 678" en el servicio de publicación "MYPUBLISHER" utilizando la política "MYPOLICY" con un identificador de correlación de "TEST1". El mensaje se publica con el tema "TESTS".

```
VALUES MQPUBLISH('MYPUBLISHER','MYPOLICY','Testing 678','TESTS','TEST1')
```

Ejemplo 4: En este ejemplo se publica la serie "Testing 901" en el servicio de publicación "MYPUBLISHER" bajo el tema "TESTS" utilizando la política por omisión (DB2.DEFAULT.POLICY) y ningún identificador de correlación.

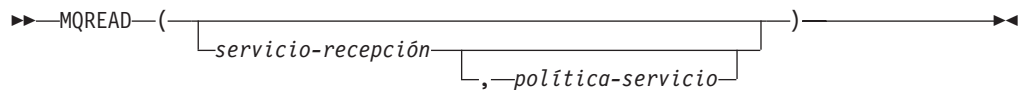
```
VALUES MQPUBLISH('Testing 901','TESTS')
```

MQREAD

La función MQREAD devuelve un mensaje de la ubicación MQSeries especificada por *servicio-recepción*, utilizando la política de calidad de servicio definida en *política-servicio*. Al ejecutar esta operación no se elimina el mensaje de la cola asociada con *servicio-recepción*, sino que, en lugar de ello, se devuelve el mensaje situado en la cabecera de la cola.

El tipo de datos del resultado es VARCHAR (32000). Si no hay mensajes disponibles para devolverse, el resultado es el valor nulo.

Sintaxis



El esquema es DB2MQ para funciones de cola de mensajes de no transacción, y DB2MQ1C para funciones transaccionales de MQ de comprometer en una fase.

Parámetros de la función

servicio-recepción

Serie que contiene el destino MQSeries lógico desde el que se debe recibir el mensaje. Si se especifica, el *servicio-recepción* debe hacer referencia a un punto de servicio definido en la tabla DB2MQ.MQSERVICE. Un punto de servicio es un punto final lógico desde el que se envía o se recibe un mensaje. Las definiciones de punto de servicio incluyen el nombre del gestor de colas y de la cola de MQSeries. Si no se especifica *servicio-recepción*, se utilizará DB2.DEFAULT.SERVICE. El tamaño máximo de *servicio-recepción* es de 48 bytes.

política-servicio

Serie que contiene la política de servicio de MQSeries utilizada en el manejo de este mensaje. Si se especifica, la *política-servicio* debe hacer referencia a una política definida en la tabla DB2MQ.MQPOLICY. Una Política de servicio define un conjunto de opciones de calidad de servicio que deben aplicarse a esta operación de mensajería. Estas opciones incluyen la prioridad de mensaje y la permanencia de mensaje. Si no se especifica *política-servicio*, se utilizará el valor DB2.DEFAULT.POLICY por omisión. El tamaño máximo de *política-servicio* es de 48 bytes.

Ejemplos

Ejemplo 1: En este ejemplo se lee el mensaje que se encuentra en la cabecera de la cola especificada por el servicio por omisión (DB2.DEFAULT.SERVICE), utilizando la política por omisión (DB2.DEFAULT.POLICY).

```
VALUES MQREAD()
```

Ejemplo 2: En este ejemplo se lee el mensaje que se encuentra en la cabecera de la cola especificada por el servicio "MYSERVICE" utilizando la política por omisión (DB2.DEFAULT.POLICY).

```
VALUES MQREAD('MYSERVICE')
```

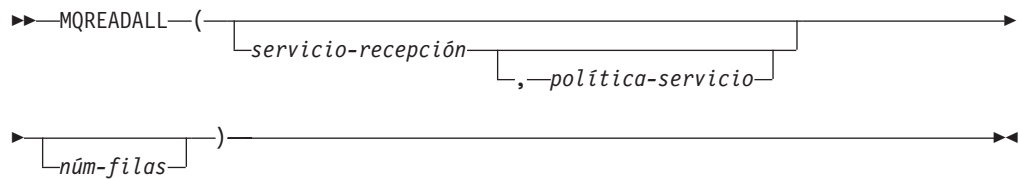
Ejemplo 3: En este ejemplo se lee el mensaje situado en la cabecera de la cola especificada por el servicio "MYSERVICE", utilizando la política "MYPOLICY".

```
VALUES MQREAD('MYSERVICE', 'MYPOLICY')
```

MQREADALL

La función de tabla MQREADALL devuelve una tabla que contiene los mensajes y los metadatos de mensaje de la ubicación MQSeries especificada por *servicio-recepción*, utilizando la política de calidad de servicio *política-servicio*. Al realizar esta operación no se eliminan los mensajes de la cola asociada con *servicio-recepción*.

Sintaxis



El esquema es DB2MQ para funciones de cola de mensajes de no transacción, y DB2MQ1C para funciones transaccionales de MQ de comprometer en una fase.

Parámetros de la función de tabla

servicio-recepción

Serie que contiene el destino MQSeries lógico en el cual se lee el mensaje. Si se especifica, el *servicio-recepción* debe hacer referencia a un punto de servicio definido en la tabla DB2MQ.MQSERVICE. Un punto de servicio es un punto final lógico desde el cual se envía o se recibe un mensaje. Las definiciones de punto de servicio incluyen el nombre del gestor de colas y de la cola de MQSeries. Si no se especifica *servicio-recepción*, se utilizará DB2.DEFAULT.SERVICE. El tamaño máximo de *servicio-recepción* es de 48 bytes.

política-servicio

Serie que contiene la política de servicio de MQSeries utilizada en el manejo de este mensaje. Si se especifica, la *política-servicio* hace referencia a una política definida en la tabla DB2MQ.MQPOLICY. Una política de servicio define un conjunto de opciones de calidad de servicio que deben aplicarse a esta operación de mensajería. Estas opciones incluyen la prioridad de mensaje y la permanencia de mensaje. Si no se especifica *política-servicio*, se utilizará el valor DB2.DEFAULT.POLICY por omisión. El tamaño máximo de *política-servicio* es de 48 bytes.

núm-filas

Entero positivo que contiene el número máximo de mensajes que debe devolver la función.

Si se especifica *núm-filas*, se devolverá un máximo de *núm-filas* mensajes. Si no se especifica *núm-filas*, se devolverán todos los mensajes disponibles.

Autorización

Privilegio EXECUTE para la función de tabla MQREADALL.

Ejemplos

Ejemplo 1: En este ejemplo se reciben todos los mensajes de la cola especificada por el servicio por omisión (DB2.DEFAULT.SERVICE), utilizando la política por omisión (DB2.DEFAULT.POLICY). Los mensajes y todos los metadatos se devuelven como una tabla.

```
SELECT * FROM table (MQREADALL()) AS T
```

Ejemplo 2: En este ejemplo se reciben todos los mensajes de la cabecera de la cola especificada por el servicio MYSERVICE, utilizando la política por omisión (DB2.DEFAULT.POLICY). Sólo se devuelven las columnas MSG y CORRELID.

```
SELECT T.MSG, T.CORRELID FROM table (MQREADALL('MYSERVICE')) AS T
```


Ejemplo 3: En este ejemplo se lee la cabecera de la cola especificada por el servicio por omisión (DB2.DEFAULT.SERVICE), utilizando la política por omisión (DB2.DEFAULT.POLICY). Sólo se devuelven los mensajes con un CORRELID de '1234'. Se devuelven todas las columnas.

```
SELECT * FROM table (MQREADALL()) AS T WHERE T.CORRELID = '1234'
```

Ejemplo 4: En este ejemplo se reciben los 10 primeros mensajes de la cabecera de la cola especificada por el servicio por omisión (DB2.DEFAULT.SERVICE), utilizando la política por omisión (DB2.DEFAULT.POLICY). Se devuelven todas las columnas.

```
SELECT * FROM table (MQREADALL(10)) AS T
```

Información devuelta

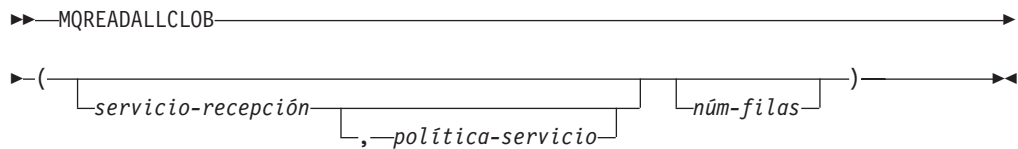
Tabla 114. Información devuelta por la función de tabla MQREADALL

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|----------------|--|
| MSG | VARCHAR(32000) | Incluye el contenido del mensaje MQSeries. |
| CORRELID | VARCHAR(24) | Contiene un ID de correlación que puede utilizarse para identificar mensajes. Puede seleccionar un mensaje de la cola utilizando este identificador. En el caso de un escenario de petición y respuesta, el ID de correlación le permite asociar una respuesta con una petición determinada. |
| TOPIC | VARCHAR(40) | Contiene el tema con el que se ha publicado el mensaje, si está disponible. |
| QNAME | VARCHAR(48) | Contiene el nombre de la cola donde se ha recibido el mensaje. |
| MSGID | CHAR(24) | Contiene el identificador de MQSeries exclusivo asignado para este mensaje. |
| MSGFORMAT | VARCHAR(8) | Contiene el formato del mensaje, definido por MQSeries. Las series típicas tienen un formato MQSTR. |

MQREADALLCLOB

La función de tabla MQREADALLCLOB devuelve una tabla que contiene los mensajes y los metadatos de mensaje de la ubicación MQSeries especificada por *servicio-recepción*, utilizando la política de calidad de servicio *política-servicio*. Al realizar esta operación no se eliminan los mensajes de la cola asociada con *servicio-recepción*.

Sintaxis



El esquema es DB2MQ.

Parámetros de la función de tabla

servicio-recepción

Serie que contiene el destino MQSeries lógico en el cual se lee el mensaje. Si se especifica, el *servicio-recepción* debe hacer referencia a un punto de servicio definido en la tabla DB2MQ.MQSERVICE. Un punto de servicio es un punto final lógico desde el cual se envía o se recibe un mensaje. Las definiciones de punto de servicio incluyen el nombre del gestor de colas y de la cola de MQSeries. Si no se especifica *servicio-recepción*, se utilizará DB2.DEFAULT.SERVICE. El tamaño máximo de *servicio-recepción* es de 48 bytes.

política-servicio

Serie que contiene la política de servicio de MQSeries utilizada en el manejo de este mensaje. Si se especifica, la *política-servicio* hace referencia a una política definida en la tabla DB2MQ.MQPOLICY. Una política de servicio define un conjunto de opciones de calidad de servicio que deben aplicarse a esta operación de mensajería. Estas opciones incluyen la prioridad de mensaje y la permanencia de mensaje. Si no se especifica *política-servicio*, se utilizará el valor DB2.DEFAULT.POLICY por omisión. El tamaño máximo de *política-servicio* es de 48 bytes.

núm-filas

Entero positivo que contiene el número máximo de mensajes que debe devolver la función.

Si se especifica *núm-filas*, se devolverá un máximo de *núm-filas* mensajes. Si no se especifica *núm-filas*, se devolverán todos los mensajes disponibles.

Autorización

Privilegio EXECUTE para la función de tabla MQREADALLCLOB.

Ejemplos

Ejemplo 1: En este ejemplo se reciben todos los mensajes de la cola especificada por el servicio por omisión (DB2.DEFAULT.SERVICE), utilizando la política por omisión (DB2.DEFAULT.POLICY). Los mensajes y todos los metadatos se devuelven como una tabla.

```
SELECT * FROM table (MQREADALLCLOB()) AS T
```

Ejemplo 2: En este ejemplo se reciben todos los mensajes de la cabecera de la cola especificada por el servicio MYSERVICE, utilizando la política por omisión (DB2.DEFAULT.POLICY). Sólo se devuelven las columnas MSG y CORRELID.

```
SELECT T.MSG, T.CORRELID FROM table (MQREADALLCLOB('MYSERVICE')) AS T
```

Ejemplo 3: En este ejemplo se lee la cabecera de la cola especificada por el servicio por omisión (DB2.DEFAULT.SERVICE), utilizando la política por omisión (DB2.DEFAULT.POLICY). Sólo se devuelven los mensajes con un CORRELID de '1234'. Se devuelven todas las columnas.

```
SELECT * FROM table (MQREADALLCLOB()) AS T WHERE T.CORRELID = '1234'
```

Ejemplo 4: En este ejemplo se reciben los 10 primeros mensajes de la cabecera de la cola especificada por el servicio por omisión (DB2.DEFAULT.SERVICE), utilizando la política por omisión (DB2.DEFAULT.POLICY). Se devuelven todas las columnas.

```
SELECT * FROM table (MQREADALLCLOB(10)) AS T
```

Información devuelta

Tabla 115. Información devuelta por la función de tabla MQREADALLCLOB

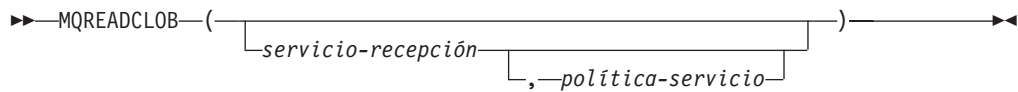
| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| MSG | CLOB(1M) | Incluye el contenido del mensaje MQSeries. |
| CORRELID | VARCHAR(24) | Contiene un ID de correlación que puede utilizarse para identificar mensajes. Puede seleccionar un mensaje de la cola utilizando este identificador. En el caso de un escenario de petición y respuesta, el ID de correlación le permite asociar una respuesta con una petición determinada. |
| TOPIC | VARCHAR(40) | Contiene el tema con el que se ha publicado el mensaje, si está disponible. |
| QNAME | VARCHAR(48) | Contiene el nombre de la cola donde se ha recibido el mensaje. |
| MSGID | CHAR(24) | Contiene el identificador de MQSeries exclusivo asignado para este mensaje. |
| MSGFORMAT | VARCHAR(8) | Contiene el formato del mensaje, definido por MQSeries. Las series típicas tienen un formato MQSTR. |

MQREADCLOB

La función MQREADCLOB devuelve un mensaje de la ubicación MQSeries especificada por *servicio-recepción*, utilizando la política de calidad de servicio definida en *política-servicio*. Al ejecutar esta operación no se elimina el mensaje de la cola asociada con *servicio-recepción*, sino que, en lugar de ello, se devuelve el mensaje situado en la cabecera de la cola.

El tipo de datos del resultado es CLOB(1M). Si no hay mensajes disponibles para devolverse, el resultado es el valor nulo.

Sintaxis



El esquema es DB2MQ.

Parámetros de la función

servicio-recepción

Serie que contiene el destino MQSeries lógico desde el que se debe recibir el mensaje. Si se especifica, el *servicio-recepción* debe hacer referencia a un punto de servicio definido en la tabla DB2MQ.MQSERVICE. Un punto de servicio es un punto final lógico desde el que se envía o se recibe un mensaje. Las definiciones de punto de servicio incluyen el nombre del gestor de colas y de la cola de MQSeries. Si no se especifica *servicio-recepción*, se utilizará DB2.DEFAULT.SERVICE. El tamaño máximo de *servicio-recepción* es de 48 bytes.

política-servicio

Serie que contiene la política de servicio de MQSeries utilizada en el manejo de este mensaje. Si se especifica, la *política-servicio* debe hacer referencia a una política definida en la tabla DB2MQ.MQPOLICY. Una Política de servicio define un conjunto de opciones de calidad de servicio que deben aplicarse a esta operación de mensajería. Estas opciones incluyen la prioridad de mensaje y la permanencia de mensaje. Si no se especifica *política-servicio*, se utilizará el valor DB2.DEFAULT.POLICY por omisión. El tamaño máximo de *política-servicio* es de 48 bytes.

Ejemplos

Ejemplo 1: En este ejemplo se lee el mensaje que se encuentra en la cabecera de la cola especificada por el servicio por omisión (DB2.DEFAULT.SERVICE), utilizando la política por omisión (DB2.DEFAULT.POLICY).

```
VALUES MQREADCLOB()
```

Ejemplo 2: En este ejemplo se lee el mensaje que se encuentra en la cabecera de la cola especificada por el servicio "MYSERVICE" utilizando la política por omisión (DB2.DEFAULT.POLICY).

```
VALUES MQREADCLOB('MYSERVICE')
```

Ejemplo 3: En este ejemplo se lee el mensaje situado en la cabecera de la cola especificada por el servicio "MYSERVICE", utilizando la política "MYPOLICY".

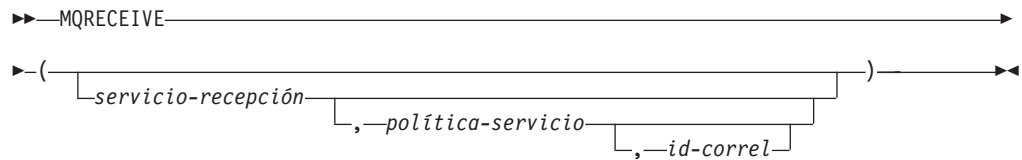
```
VALUES MQREADCLOB('MYSERVICE', 'MYPOLICY')
```

MQRECEIVE

La función MQRECEIVE devuelve un mensaje de la ubicación MQSeries especificada por *servicio-recepción*, utilizando la política de calidad de servicio *política-servicio*. Al realizar esta operación se elimina el mensaje de la cola asociada con *servicio-recepción*. Si se especifica el *id-correl*, se devolverá el primer mensaje con un identificador de correlación que coincida. Si no se especifica *id-correl*, se devolverá el mensaje situado en la cabecera de la cola.

El tipo de datos del resultado es VARCHAR (32000). Si no hay mensajes disponibles para devolverse, el resultado es el valor nulo.

Sintaxis



El esquema es DB2MQ para funciones de cola de mensajes de no transacción, y DB2MQ1C para funciones transaccionales de MQ de comprometer en una fase.

Parámetros de la función

servicio-recepción

Serie que contiene el destino MQSeries lógico desde el cual se recibe el mensaje. Si se especifica, el *servicio-recepción* debe hacer referencia a un punto de servicio definido en la tabla DB2MQ.MQSERVICE. Un punto de servicio es un punto final lógico desde el cual se envía o se recibe un mensaje. Las definiciones de punto de servicio incluyen el nombre del gestor de colas y de la cola de MQSeries. Si no se especifica *servicio-recepción*, se utiliza DB2.DEFAULT.SERVICE. El tamaño máximo de *servicio-recepción* es de 48 bytes.

política-servicio

Serie que contiene la política de servicio de MQSeries que se debe utilizar en el manejo de este mensaje. Si se especifica, la *política-servicio* debe hacer referencia a una política definida en la tabla DB2MQ.MQPOLICY. Una política de servicio define un conjunto de opciones de calidad de servicio que deben aplicarse a esta operación de mensajería. Estas opciones incluyen la prioridad de mensaje y la permanencia de mensaje. Si no se especifica *política-servicio*, se utiliza el valor DB2.DEFAULT.POLICY por omisión. El tamaño máximo de *política-servicio* es de 48 bytes.

id-correl

Serie que contiene una identificación de correlación opcional que se debe asociar con este mensaje. El *id-correl* se especifica normalmente en escenarios de petición y respuesta para asociar las peticiones con las respuestas. Si no se especifica, no se especificará ningún id de correlación. El tamaño máximo de *id-correl* es de 24 bytes.

Ejemplos

Ejemplo 1: En este ejemplo se recibe el mensaje que se encuentra en la cabecera de la cola especificada por el servicio por omisión (DB2.DEFAULT.SERVICE), utilizando la política por omisión (DB2.DEFAULT.POLICY).

```
VALUES MQRECEIVE()
```

Ejemplo 2: En este ejemplo se recibe el mensaje que se encuentra en la cabecera de la cola especificada por el servicio "MYSERVICE" utilizando la política por omisión (DB2.DEFAULT.POLICY).

```
VALUES MQRECEIVE('MYSERVICE')
```

Ejemplo 3: En este ejemplo se recibe el mensaje que se encuentra en la cabecera de la cola especificada por el servicio "MYSERVICE", utilizando la política "MYPOLICY".

```
VALUES MQRECEIVE('MYSERVICE', 'MYPOLICY')
```

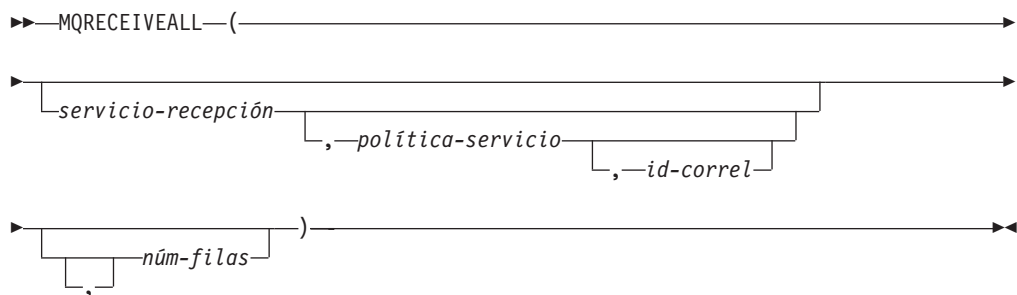
Ejemplo 4: En este ejemplo se recibe el primer mensaje con un id de correlación que coincide con '1234' de la cabecera de la cola especificada por el servicio "MYSERVICE", utilizando la política "MYPOLICY".

```
VALUES MQRECEIVE('MYSERVICE', 'MYPOLICY', '1234')
```

MQRECEIVEALL

La función de tabla MQRECEIVEALL devuelve una tabla que contiene los mensajes y los metadatos de mensaje de la ubicación MQSeries especificada por *servicio-recepción*, utilizando la política de calidad de servicio *política-servicio*. Al realizar esta operación se eliminan los mensajes de la cola asociada con *servicio-recepción*.

Sintaxis



El esquema es DB2MQ para funciones de cola de mensajes de no transacción, y DB2MQ1C para funciones transaccionales de MQ de comprometer en una fase.

Parámetros de la función de tabla

servicio-recepción

Serie que contiene el destino MQSeries lógico desde el cual se recibe el mensaje. Si se especifica, el *servicio-recepción* debe hacer referencia a un punto de servicio definido en la tabla DB2MQ.MQSERVICE. Un punto de servicio es un punto final lógico desde el cual se envía o se recibe un mensaje. Las definiciones de punto de servicio incluyen el nombre del gestor de colas y de la cola de MQSeries. Si no se especifica *servicio-recepción*, se utilizará DB2.DEFAULT.SERVICE. El tamaño máximo de *servicio-recepción* es de 48 bytes.

política-servicio

Serie que contiene la política de servicio de MQSeries utilizada en el manejo de este mensaje. Si se especifica, la *política-servicio* hace referencia a una política definida en la tabla DB2MQ.MQPOLICY. Una política de servicio define un conjunto de opciones de calidad de servicio que deben aplicarse a esta operación de mensajería. Estas opciones incluyen la prioridad de mensaje y la permanencia de mensaje. Si no se especifica *política-servicio*, se utilizará el valor DB2.DEFAULT.POLICY por omisión. El tamaño máximo de *política-servicio* es de 48 bytes.

id-correl

Serie opcional que contiene un identificador de correlación asociado con este mensaje. El *id-correl* se especifica normalmente en escenarios de petición y respuesta para asociar las peticiones con las respuestas. Si no se especifica, no se especifica ningún id de correlación. El tamaño máximo de *id-correl* es de 24 bytes.

Si se especifica *correl-id*, se devolverán y eliminarán de la cola todos los mensajes que tengan un identificador de correlación que coincida. Si no se especifica *correl-id*, se devolverá el mensaje situado en la cabecera de la cola.

núm-filas

Entero positivo que contiene el número máximo de mensajes que debe devolver la función.

Si se especifica *núm-filas*, se devolverá un máximo de *núm-filas* mensajes. Si no se especifica *núm-filas*, se devolverán todos los mensajes disponibles.

Autorización

Privilegio EXECUTE para la función de tabla MQRECEIVEALL.

Ejemplos

Ejemplo 1: En este ejemplo se reciben todos los mensajes de la cola especificada por el servicio por omisión (DB2.DEFAULT.SERVICE), utilizando la política por omisión (DB2.DEFAULT.POLICY). Los mensajes y todos los metadatos se devuelven como una tabla.

```
SELECT * FROM table (MQRECEIVEALL()) AS T
```

Ejemplo 2: En este ejemplo se reciben todos los mensajes de la cabecera de la cola especificada por el servicio MYSERVICE, utilizando la política por omisión (DB2.DEFAULT.POLICY). Sólo se devuelven las columnas MSG y CORRELID.

```
SELECT T.MSG, T.CORRELID FROM table (MQRECEIVEALL('MYSERVICE')) AS T
```

Ejemplo 3: En este ejemplo se reciben todos los mensajes de la cabecera de la cola especificada por el servicio "MYSERVICE", utilizando la política "MYPOLICY". Sólo se devuelven los mensajes con un CORRELID de '1234'. Sólo se devuelven las columnas MSG y CORRELID.

```
SELECT T.MSG, T.CORRELID FROM table  
(MQRECEIVEALL('MYSERVICE', 'MYPOLICY', '1234')) AS T
```

Ejemplo 4: En este ejemplo se reciben los 10 primeros mensajes de la cabecera de la cola especificada por el servicio por omisión (DB2.DEFAULT.SERVICE), utilizando la política por omisión (DB2.DEFAULT.POLICY). Se devuelven todas las columnas.

```
SELECT * FROM table (MQRECEIVEALL(10)) AS T
```

Información devuelta

Tabla 116. Información devuelta por la función de tabla MQRECEIVEALL

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|----------------|--|
| MSG | VARCHAR(32000) | Incluye el contenido del mensaje MQSeries. |

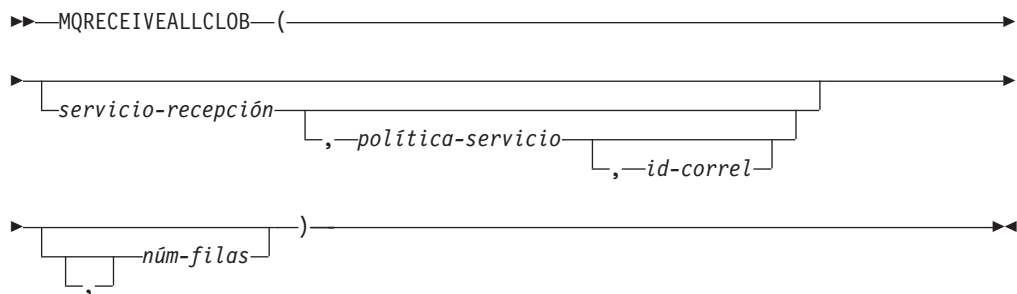
Tabla 116. Información devuelta por la función de tabla MQRECEIVEALL (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| CORRELID | VARCHAR(24) | Contiene un ID de correlación que puede utilizarse para identificar mensajes. Puede seleccionar un mensaje de la cola utilizando este identificador. En el caso de un escenario de petición y respuesta, el ID de correlación le permite asociar una respuesta con una petición determinada. |
| TOPIC | VARCHAR(40) | Contiene el tema con el que se ha publicado el mensaje, si está disponible. |
| QNAME | VARCHAR(48) | Contiene el nombre de la cola donde se ha recibido el mensaje. |
| MSGID | CHAR(24) | Contiene el identificador de MQSeries exclusivo asignado para este mensaje. |
| MSGFORMAT | VARCHAR(8) | Contiene el formato del mensaje, definido por MQSeries. Las series típicas tienen un formato MQSTR. |

MQRECEIVEALLCLOB

La función de tabla MQRECEIVEALLCLOB devuelve una tabla que contiene los mensajes y los metadatos de mensaje de la ubicación MQSeries especificada por *servicio-recepción*, utilizando la política de calidad de servicio *política-servicio*. Al realizar esta operación se eliminan los mensajes de la cola asociada con *servicio-recepción*.

Sintaxis



El esquema es DB2MQ.

Parámetros de la función de tabla

servicio-recepción

Serie que contiene el destino MQSeries lógico desde el cual se recibe el

mensaje. Si se especifica, el *servicio-recepción* debe hacer referencia a un punto de servicio definido en la tabla DB2MQ.MQSERVICE. Un punto de servicio es un punto final lógico desde el cual se envía o se recibe un mensaje. Las definiciones de punto de servicio incluyen el nombre del gestor de colas y de la cola de MQSeries. Si no se especifica *servicio-recepción*, se utilizará DB2.DEFAULT.SERVICE. El tamaño máximo de *servicio-recepción* es de 48 bytes.

política-servicio

Serie que contiene la política de servicio de MQSeries utilizada en el manejo de este mensaje. Si se especifica, la *política-servicio* hace referencia a una política definida en la tabla DB2MQ.MQPOLICY. Una política de servicio define un conjunto de opciones de calidad de servicio que se deben aplicar a esta operación de mensajería. Estas opciones incluyen la prioridad de mensaje y la permanencia de mensaje. Si no se especifica *política-servicio*, se utilizará el valor DB2.DEFAULT.POLICY por omisión. El tamaño máximo de *política-servicio* es de 48 bytes.

id-correl

Serie opcional que contiene un identificador de correlación asociado con este mensaje. El *id-correl* se especifica normalmente en escenarios de petición y respuesta para asociar las peticiones con las respuestas. Si no se especifica, no se especifica ningún id de correlación. El tamaño máximo de *id-correl* es de 24 bytes.

Si se especifica *id-correl*, sólo se devolverán los mensajes que tengan un identificador de correlación que coincida. Si no se especifica *id-correl*, se devolverá el mensaje situado en la cabecera de la cola.

núm-filas

Entero positivo que contiene el número máximo de mensajes que debe devolver la función.

Si se especifica *núm-filas*, se devolverá un máximo de *núm-filas* mensajes. Si no se especifica *núm-filas*, se devolverán todos los mensajes disponibles.

Autorización

Privilegio EXECUTE para la función de tabla MQRECEIVEALLCLOB.

Ejemplos

Ejemplo 1: En este ejemplo se reciben todos los mensajes de la cola especificada por el servicio por omisión (DB2.DEFAULT.SERVICE), utilizando la política por omisión (DB2.DEFAULT.POLICY). Los mensajes y todos los metadatos se devuelven como una tabla.

```
SELECT * FROM table (MQRECEIVEALLCLOB()) AS T
```

Ejemplo 2: En este ejemplo se reciben todos los mensajes de la cabecera de la cola especificada por el servicio MYSERVICE, utilizando la política por omisión (DB2.DEFAULT.POLICY). Sólo se devuelven las columnas MSG y CORRELID.

```
SELECT T.MSG, T.CORRELID  
FROM table (MQRECEIVEALLCLOB('MYSERVICE')) AS T
```

Ejemplo 3: En este ejemplo se reciben todos los mensajes de la cabecera de la cola especificada por el servicio "MYSERVICE", utilizando la política "MYPOLICY". Sólo se devuelven los mensajes con un CORRELID de '1234'. Sólo se devuelven las columnas MSG y CORRELID.

```
SELECT T.MSG, T.CORRELID
FROM table (MQRECEIVEALLCLOB('MYSERVICE','MYPOLICY','1234')) AS T
```

Ejemplo 4: En este ejemplo se reciben los 10 primeros mensajes de la cabecera de la cola especificada por el servicio por omisión (DB2.DEFAULT.SERVICE), utilizando la política por omisión (DB2.DEFAULT.POLICY). Se devuelven todas las columnas.

```
SELECT * FROM table (MQRECEIVEALLCLOB(10)) AS T
```

Información devuelta

Tabla 117. Información devuelta por la función de tabla MQRECEIVEALLCLOB

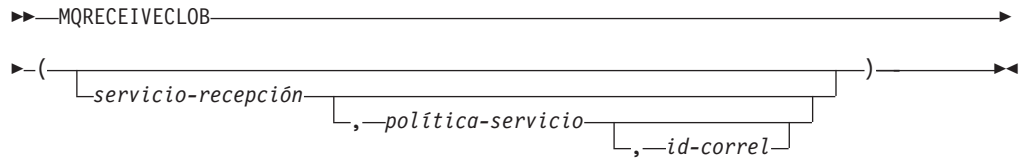
| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| MSG | CLOB(1M) | Incluye el contenido del mensaje MQSeries. |
| CORRELID | VARCHAR(24) | Contiene un ID de correlación que puede utilizarse para identificar mensajes. Puede seleccionar un mensaje de la cola utilizando este identificador. En el caso de un escenario de petición y respuesta, el ID de correlación le permite asociar una respuesta con una petición determinada. |
| TOPIC | VARCHAR(40) | Contiene el tema con el que se ha publicado el mensaje, si está disponible. |
| QNAME | VARCHAR(48) | Contiene el nombre de la cola donde se ha recibido el mensaje. |
| MSGID | CHAR(24) | Contiene el identificador de MQSeries exclusivo asignado para este mensaje. |
| MSGFORMAT | VARCHAR(8) | Contiene el formato del mensaje, definido por MQSeries. Las series típicas tienen un formato MQSTR. |

MQRECEIVECLOB

La función MQRECEIVECLOB devuelve un mensaje de la ubicación MQSeries especificada por *servicio-recepción*, utilizando la política de calidad de servicio *política-servicio*. Al realizar esta operación se elimina el mensaje de la cola asociada con *servicio-recepción*. Si se especifica el *id-correl*, se devolverá el primer mensaje con un identificador de correlación coincidente. Si no se especifica *id-correl*, se devolverá el mensaje situado en la cabecera de la cola.

El tipo de datos del resultado es CLOB(1M). Si no hay mensajes disponibles para devolverse, el resultado es el valor nulo.

Sintaxis



El esquema es DB2MQ.

Parámetros de la función

servicio-recepción

Serie que contiene el destino MQSeries lógico desde el cual se recibe el mensaje. Si se especifica, el *servicio-recepción* debe hacer referencia a un punto de servicio definido en la tabla DB2MQ.MQSERVICE. Un punto de servicio es un punto final lógico desde el cual se envía o se recibe un mensaje. Las definiciones de punto de servicio incluyen el nombre del gestor de colas y de la cola de MQSeries. Si no se especifica *servicio-recepción*, se utiliza DB2.DEFAULT.SERVICE. El tamaño máximo de *servicio-recepción* es de 48 bytes.

política-servicio

Serie que contiene la política de servicio de MQSeries que se debe utilizar en el manejo de este mensaje. Si se especifica, la *política-servicio* debe hacer referencia a una política definida en la tabla DB2MQ.MQPOLICY. Una política de servicio define un conjunto de opciones de calidad de servicio que deben aplicarse a esta operación de mensajería. Estas opciones incluyen la prioridad de mensaje y la permanencia de mensaje. Si no se especifica *política-servicio*, se utiliza el valor DB2.DEFAULT.POLICY por omisión. El tamaño máximo de *política-servicio* es de 48 bytes.

id-correl

Serie que contiene una identificación de correlación opcional que se debe asociar con este mensaje. El *id-correl* se especifica normalmente en escenarios de petición y respuesta para asociar las peticiones con las respuestas. Si no se especifica, no se utilizará ningún id de correlación. El tamaño máximo de *id-correl* es de 24 bytes.

Ejemplos

Ejemplo 1: En este ejemplo se recibe el mensaje que se encuentra en la cabecera de la cola especificada por el servicio por omisión (DB2.DEFAULT.SERVICE), utilizando la política por omisión (DB2.DEFAULT.POLICY).

```
VALUES MQRECEIVECLOB()
```

Ejemplo 2: En este ejemplo se recibe el mensaje que se encuentra en la cabecera de la cola especificada por el servicio "MYSERVICE" utilizando la política por omisión (DB2.DEFAULT.POLICY).

```
VALUES MQRECEIVECLOB('MYSERVICE')
```

Ejemplo 3: En este ejemplo se recibe el mensaje que se encuentra en la cabecera de la cola especificada por el servicio "MYSERVICE", utilizando la política "MYPOLICY".

```
VALUES MQRECEIVECLOB('MYSERVICE', 'MYPOLICY')
```

Ejemplo 4: En este ejemplo se recibe el primer mensaje con un ID de correlación que coincide con '1234' de la cabecera de la cola especificada por el servicio "MYSERVICE", utilizando la política "MYPOLICY".

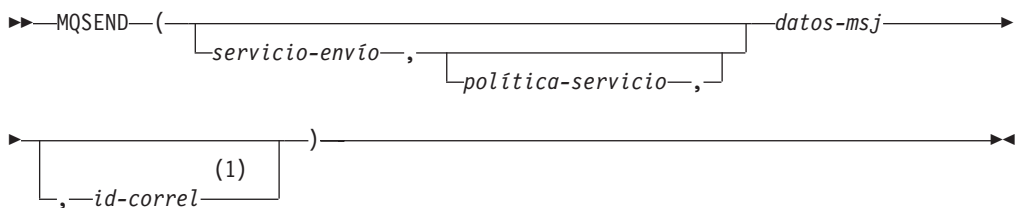
```
VALUES MQRECEIVECLOB('MYSERVICE',MYPOLICY','1234')
```

MQSEND

La función MQSEND envía los datos contenidos en *datos-msj* a la ubicación MQSeries especificada por *servicio-envío*, utilizando la calidad de la política de servicio definida por *política-servicio*. Puede especificarse un identificador de correlación de mensaje opcional definido por el usuario utilizando *id-correl*.

El tipo de datos del resultado es VARCHAR(1). El resultado de la función es '1' si es satisfactorio o cero ('0') si no es satisfactorio.

Sintaxis



Notas:

- 1 El *id-correl* no se puede especificar a no ser que se haya definido un *servicio* y una *política*.

El esquema es DB2MQ para funciones de cola de mensajes de no transacción, y DB2MQ1C para funciones transaccionales de MQ de comprometer en una fase.

Parámetros de la función

datos-msj

Expresión de serie que contiene los datos que se deben enviar a través de MQSeries. El tamaño máximo para una expresión de serie VARCHAR es de 32.000 bytes, mientras que para una expresión de serie CLOB es de 1 M bytes.

servicio-envío

Serie que contiene el destino MQSeries lógico donde se debe enviar el mensaje. Si se especifica, el *servicio-envío* hace referencia a un punto de servicio definido en la tabla DB2MQ.MQSERVICE. Un punto de servicio es un punto final lógico desde el cual se puede enviar o recibir un mensaje. Las definiciones de punto de servicio incluyen el nombre del gestor de colas y de la cola de MQSeries. Si no se especifica *servicio-envío*, se utiliza el valor de DB2.DEFAULT.SERVICE. El tamaño máximo de *servicio-envío* es de 48 bytes.

política-servicio

Serie que contiene la política de servicio de MQSeries utilizada en el manejo de este mensaje. Si se especifica, la *política-servicio* debe hacer referencia a una política de servicio definida en la tabla DB2MQ.MQPOLICY. Una Política de servicio define un conjunto de opciones de calidad de servicio que deben aplicarse a esta operación de mensajería. Estas opciones incluyen la prioridad

de mensaje y la permanencia de mensaje. Si no se especifica *política-servicio*, se utilizará un valor por omisión de DB2.DEFAULT.POLICY. El tamaño máximo de *política-servicio* es de 48 bytes.

id-correl

Serie opcional que contiene un identificador de correlación asociado con este mensaje. El *id-correl* se especifica normalmente en escenarios de petición y respuesta para asociar las peticiones con las respuestas. Si no se especifica, no se enviará ningún ID de correlación. El tamaño máximo de *id-correl* es de 24 bytes.

Ejemplos

Ejemplo 1: En este ejemplo se envía la serie "Testing 123" al servicio por omisión (DB2.DEFAULT.SERVICE), utilizando la política por omisión (DB2.DEFAULT.POLICY), sin identificador de correlación.

```
VALUES MQSEND('Testing 123')
```

Ejemplo 2: En este ejemplo se envía la serie "Testing 345" al servicio "MYSERVICE", utilizando la política "MYPOLICY", sin identificador de correlación.

```
VALUES MQSEND('MYSERVICE', 'MYPOLICY', 'Testing 345')
```

Ejemplo 3: En este ejemplo se envía la serie "Testing 678" al servicio "MYSERVICE", utilizando la política "MYPOLICY", con el identificador de correlación "TEST3".

```
VALUES MQSEND('MYSERVICE', 'MYPOLICY', 'Testing 678', 'TEST3')
```

Ejemplo 4: En este ejemplo se envía la serie "Testing 901" al servicio "MYSERVICE", utilizando la política por omisión (DB2.DEFAULT.POLICY) y ningún identificador de correlación.

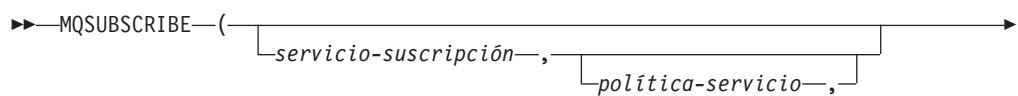
```
VALUES MQSEND('MYSERVICE', 'Testing 901')
```

MQSUBSCRIBE

La función MQSUBSCRIBE se utiliza para registrar interés en los mensajes MQSeries publicados sobre un tema especificado. La ejecución satisfactoria de esta función causa que el servidor de publicación y suscripción reenvíe los mensajes que coinciden con el tema al punto de servicio definido por *servicio-suscripción*. El *servicio-suscripción* especifica un destino lógico para los mensajes que coinciden con el tema especificado. Los mensajes que coinciden con el *tema* se colocan en la cola definida por *servicio-suscripción* y se pueden leer o recibir mediante una llamada subsiguiente a MQREAD, MQRECEIVE, MQREADALL o MQRECEIVEALL. Para obtener más detalles, visite <http://www.ibm.com/software/MQSeries>.

El tipo de datos del resultado es VARCHAR(1). El resultado de la función es '1' si es satisfactorio o cero ('0') si no es satisfactorio.

Sintaxis



El esquema es DB2MQ para funciones de cola de mensajes de no transacción, y DB2MQ1C para funciones transaccionales de MQ de comprometer en una fase.

Parámetros de la función

servicio-suscripción

Serie que contiene el punto de suscripción MQSeries lógico al que se enviarán los mensajes que coinciden con *tema*. Si se especifica, el *servicio-suscripción* debe hacer referencia a un punto de servicio de suscriptores definido en la tabla DB2MQ.MQPUBSUB que tiene el valor de tipo 'S' para servicio de publicación. Si no se especifica *servicio-suscripción*, se utilizará en su lugar DB2.DEFAULT.SUBSCRIBER. El tamaño máximo de *servicio-suscripción* es de 48 bytes.

política-servicio

Serie que contiene la política de servicio de MQSeries que se debe utilizar en el manejo del mensaje. Si se especifica, la *política-servicio* debe hacer referencia a una política definida en la tabla DB2MQ.MQPOLICY. Una Política de servicio define un conjunto de opciones de calidad de servicio que se deben aplicar a esta operación de mensajería. Estas opciones incluyen la prioridad de mensaje y la permanencia de mensaje. Si no se especifica *política-servicio*, se utilizará en su lugar el valor DB2.DEFAULT.POLICY por omisión. El tamaño máximo de *política-servicio* es de 48 bytes.

tema

Serie que define los tipos de mensajes que se deben recibir. Esta suscripción sólo recibirá los mensajes publicados con los temas especificados. Pueden coexistir múltiples suscripciones. El tamaño máximo es de 40 bytes. Se pueden especificar múltiples temas en una serie (de una longitud máxima de 40). Cada tema debe estar separado por dos puntos. Por ejemplo, "t1:t2:el tercer tema" indica que el mensaje está asociado con los tres temas: t1, t2 y "el tercer tema".

Ejemplos

Ejemplo 1: En este ejemplo se registra un interés en los mensajes que contienen el tema "Tiempo". El *servicio-suscripción* por omisión (DB2.DEFAULT.SUBSCRIBER) se registra como suscriptor y la *política-servicio* por omisión (DB2.DEFAULT.POLICY) especifica la calidad de servicio.

```
VALUES MQSUBSCRIBE('Tiempo')
```

Ejemplo 2: Este ejemplo muestra un interés de registro de suscriptor en los mensajes que contienen "Stocks". El suscriptor se registra como "PORTFOLIO-UPDATES" con la política "BASIC-POLICY".

```
VALUES MQSUBSCRIBE('PORTFOLIO-UPDATES', 'BASIC-POLICY', 'Stocks')
```

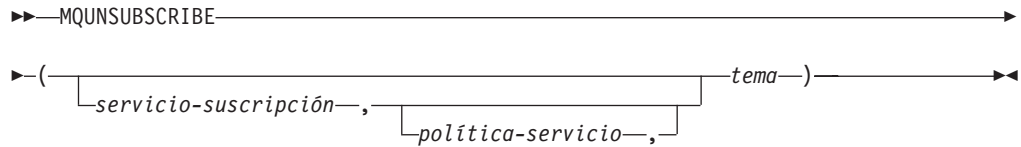
MQUNSUBSCRIBE

La función MQUNSUBSCRIBE se utiliza para eliminar el registro de una suscripción de mensajes existente. El *servicio-suscripción*, la *política-servicio* y el *tema* se utilizan para identificar la suscripción que se debe cancelar. La ejecución satisfactoria de esta función causa que el servidor de publicación y suscripción elimine la suscripción especificada. Los mensajes con el *tema* especificado no se

enviarán ya al destino lógico definido por *servicio-suscripción*. Para obtener más detalles, visite <http://www.ibm.com/software/MQSeries>.

El tipo de datos del resultado es VARCHAR(1). El resultado de la función es '1' si es satisfactorio o cero ('0') si no es satisfactorio.

Sintaxis



El esquema es DB2MQ para funciones de cola de mensajes de no transacción, y DB2MQ1C para funciones transaccionales de MQ de comprometer en una fase.

Parámetros de la función

servicio-suscripción

Si se especifica, el *servicio-suscripción* debe hacer referencia a un punto de servicio de suscriptores definido en la tabla DB2MQ.MQPUBSUB que tiene el valor de tipo 'S' para servicio de publicación. Si no se especifica *servicio-suscripción*, se utilizará en su lugar DB2.DEFAULT.SUBSCRIBER. El tamaño máximo de *servicio-suscripción* es de 48 bytes.

política-servicio

Si se especifica, la *política-servicio* debe hacer referencia a una política definida en la tabla DB2MQ.MQPOLICY. Una Política de servicio define un conjunto de opciones de calidad de servicio que se deben aplicar a esta operación de mensajería. Si no se especifica *política-servicio*, se utilizará el valor DB2.DEFAULT.POLICY por omisión. El tamaño máximo de *política-servicio* es de 48 bytes.

tema

Serie que especifica el tema de los mensajes que no se deben recibir. El tamaño máximo de *tema* es de 40 bytes. Se pueden especificar múltiples temas en una serie (de una longitud máxima de 40). Cada tema debe estar separado por dos puntos. Por ejemplo, "t1:t2:el tercer tema" indica que el mensaje está asociado con los tres temas: t1, t2 y "el tercer tema".

Ejemplos

Ejemplo 1: En este ejemplo se cancela un interés en los mensajes que contienen el tema "Tiempo". El servicio-suscripción por omisión (DB2.DEFAULT.SUBSCRIBER) se registra como servicio que cancela la suscripción y la política-servicio por omisión (DB2.DEFAULT.POLICY) especifica la calidad de servicio.

```
VALUES MQUNSUBSCRIBE('Tiempo')
```

Ejemplo 2: Este ejemplo muestra un suscriptor que cancela su interés por los mensajes que contienen "stocks". El suscriptor se registra como "PORTFOLIO-UPDATES" con la política "BASIC-POLICY".

```
VALUES MQUNSUBSCRIBE('PORTFOLIO-UPDATES', 'BASIC-POLICY', 'Stocks')
```

Capítulo 13. Rutinas y vistas de seguridad

AUTH_LIST_AUTHORITIES_FOR_AUTHID

La función de tabla AUTH_LIST_AUTHORITIES_FOR_AUTHID devuelve todas las autorizaciones de este ID de autorización encontradas en el archivo de configuración de la base de datos u otorgada a un ID de autorización directa o indirectamente mediante un grupo o un rol.

Sintaxis

►►—AUTH_LIST_AUTHORITIES_FOR_AUTHID—(—*idaut*—,—*tipout*—)————►►

El esquema es SYSPROC.

Parámetros de la función de tabla

idaut

Es un argumento de entrada de tipo VARCHAR(128) que especifica el ID de autorización que se está consultando. El ID de autorización puede ser un usuario, grupo o rol. Si *idaut* es una serie NULL o vacía, se devuelve una tabla de resultados vacía.

tipout

Es un argumento de entrada de tipo VARCHAR(1) que especifica el ID de autorización que se está consultando. Si *tipout* no existe, es una serie NULL o vacía, se devuelve una tabla de resultados vacía. Valores posibles para *tipout*:

- G: Grupo
- R: Rol
- U: Usuario

Autorización

Privilegio EXECUTE sobre la función de tabla AUTH_LIST_AUTHORITIES_FOR_AUTHID.

Información devuelta

Tabla 118. La información devuelta para AUTH_LIST_AUTHORITIES_FOR_AUTHID

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| AUTHORITY | VARCHAR(128) | Autorización que tiene el ID de autorización |
| D_USER | CHAR(1) | Autorización otorgada directamente al <i>idaut</i> , cuando el <i>tipoidaut</i> es un usuario (U). Si el <i>tipoidaut</i> es un grupo (G) o un rol (R), entonces el valor no se aplicará ('*'). <ul style="list-style-type: none">• N = No se mantiene• Y = Se mantiene• * = No se aplica |

Tabla 118. La información devuelta para AUTH_LIST_AUTHORITIES_FOR_AUTHID (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| D_GROUP | CHAR(1) | Autorización otorgada directamente al <i>idaut</i> cuando el <i>tipoidaut</i> es un grupo (G) o al grupo al que pertenece el <i>idaut</i> cuando el <i>tipoidaut</i> es un usuario (U). Si el <i>tipoidaut</i> es un rol (R), entonces el valor no se aplicará (*). <ul style="list-style-type: none"> • N = No se mantiene • Y = Se mantiene • * = No se aplica |
| D_PUBLIC | CHAR(1) | Autorización otorgada directamente al <i>idaut</i> , llamado PUBLIC cuando el <i>tipoidaut</i> es un usuario (U) o un grupo (G). Si el <i>tipoidaut</i> es un rol (R), entonces el valor no se aplicará (*). <ul style="list-style-type: none"> • N = No se mantiene • Y = Se mantiene • * = No se aplica |
| ROLE_USER | CHAR(1) | Autorización otorgada directamente a un rol con el <i>idaut</i> , cuando el <i>tipoidaut</i> es un usuario (U). Si el <i>tipoidaut</i> es un grupo (G) o un rol (R), entonces el valor no se aplicará (*). El rol podría formar parte de una jerarquía de roles. <ul style="list-style-type: none"> • N = No se mantiene • Y = Se mantiene • * = No se aplica |
| ROLE_GROUP | CHAR(1) | Autorización otorgada directamente a un rol con el <i>idaut</i> , cuando el <i>tipoidaut</i> es un grupo (G). Si el <i>tipoidaut</i> es un usuario (U) o una rol (R), entonces el valor no se aplicará (*). El rol podría formar parte de una jerarquía de roles. <ul style="list-style-type: none"> • N = No se mantiene • Y = Se mantiene • * = No se aplica |
| ROLE_PUBLIC | CHAR(1) | Autorización otorgada directamente a un rol con el <i>idaut</i> , llamado PUBLIC cuando el <i>tipoidaut</i> es un usuario (U) o un grupo (G). Si el <i>tipoidaut</i> es un rol (R), entonces el valor no se aplicará (*). El rol podría formar parte de una jerarquía de roles. <ul style="list-style-type: none"> • N = No se mantiene • Y = Se mantiene • * = No se aplica |
| D_ROLE | CHAR(1) | Autorización otorgada a un rol o a un rol otorgado al rol. Si el <i>tipoidaut</i> es un usuario (U) o un grupo (G), entonces el valor no se aplicará (*). El rol podría formar parte de una jerarquía de roles. <ul style="list-style-type: none"> • N = No se mantiene • Y = Se mantiene • * = No se aplica |

Ejemplo

El usuario ALICE dispone de los privilegios por omisión BIND, CONNECT, CREATETAB y IMPLICIT_SCHEMA a través del grupo especial PUBLIC. ALICE es un miembro de un grupo ADMIN1 que dispone de las siguientes autorizaciones de sistema: SYSADM, SYSCTRL y SYSMAINT. También es miembro del grupo ADMIN2 que tiene la autorización DBADM. Además, ALICE dispone de las autorizaciones de base de datos DBADM y SECADM. Se ha el rol R1 a ALICE. Se

ha otorgado la autorización LOAD al rol R1. Se ha otorgado el rol R2 al grupo ADMIN1. La autorización CREATE_NOT_FENCED_ROUTINE se ha otorgado al rol R2.

Ejemplo 1: Recuperar todas las autorizaciones otorgadas directamente al usuario ALICE o indirectamente a través de un grupo, PUBLIC o un rol.

```
SELECT AUTHORITY, D_USER, D_GROUP, D_PUBLIC, ROLE_USER, ROLE_GROUP, ROLE_PUBLIC, D_ROLE
FROM TABLE (SYSPROC.AUTH_LIST_AUTHORITIES_FOR_AUTHID ('ALICE', 'U') ) AS T
ORDER BY AUTHORITY
```

| AUTHORITY | D_USER | D_GROUP | D_PUBLIC | ROLE_USER | ROLE_GROUP | ROLE_PUBLIC | D_ROLE |
|---------------------------|--------|---------|----------|-----------|------------|-------------|--------|
| ACCESSCTRL | N | N | N | N | N | N | * |
| BINDADD | N | N | Y | N | N | N | * |
| CONNECT | N | N | Y | N | N | N | * |
| CREATE_EXTERNAL_ROUTINE | N | N | N | N | N | N | * |
| CREATE_NOT_FENCED_ROUTINE | N | N | N | N | Y | N | * |
| CREATETAB | N | N | Y | N | N | N | * |
| DATAACCESS | N | N | N | N | N | N | * |
| DBADM | Y | Y | N | N | N | N | * |
| EXPLAIN | N | N | N | N | N | N | * |
| IMPLICIT_SCHEMA | N | N | Y | N | N | N | * |
| LOAD | N | N | N | Y | N | N | * |
| QUIESCE_CONNECT | N | N | N | N | N | N | * |
| SECADM | Y | N | N | N | N | N | * |
| SQLADM | N | N | N | N | N | N | * |
| SYSADM | * | Y | * | * | * | * | * |
| SYSCTRL | * | Y | * | * | * | * | * |
| SYSMAINT | * | Y | * | * | * | * | * |
| SYSMON | * | N | * | * | * | * | * |
| WLMADM | N | N | N | N | N | N | * |

Ejemplo 2: Recuperar todas las autorizaciones otorgadas directamente al grupo ADMIN1 o indirectamente a través de PUBLIC o un rol.

```
SELECT AUTHORITY, D_USER, D_GROUP, D_PUBLIC, ROLE_USER, ROLE_GROUP, ROLE_PUBLIC, D_ROLE
FROM TABLE (SYSPROC.AUTH_LIST_AUTHORITIES_FOR_AUTHID ('ADMIN1', 'G') ) AS T
ORDER BY AUTHORITY
```

| AUTHORITY | D_USER | D_GROUP | D_PUBLIC | ROLE_USER | ROLE_GROUP | ROLE_PUBLIC | D_ROLE |
|---------------------------|--------|---------|----------|-----------|------------|-------------|--------|
| ACCESSCTRL | * | N | * | * | N | * | * |
| BINDADD | * | N | * | * | N | * | * |
| CONNECT | * | N | * | * | N | * | * |
| CREATE_EXTERNAL_ROUTINE | * | N | * | * | N | * | * |
| CREATE_NOT_FENCED_ROUTINE | * | N | * | * | Y | * | * |
| CREATETAB | * | N | * | * | N | * | * |
| DATAACCESS | * | N | * | * | N | * | * |
| DBADM | * | N | * | * | N | * | * |
| EXPLAIN | * | N | * | * | N | * | * |
| IMPLICIT_SCHEMA | * | N | * | * | N | * | * |
| LOAD | * | N | * | * | N | * | * |
| QUIESCE_CONNECT | * | N | * | * | N | * | * |
| SECADM | * | N | * | * | N | * | * |
| SQLADM | * | N | * | * | N | * | * |
| SYSADM | * | Y | * | * | * | * | * |
| SYSCTRL | * | Y | * | * | * | * | * |
| SYSMAINT | * | Y | * | * | * | * | * |
| SYSMON | * | N | * | * | * | * | * |
| WLMADM | * | N | * | * | N | * | * |

Ejemplo 3: Recuperar todas las autorizaciones otorgadas directamente al grupo especial PUBLIC o indirectamente a través de un rol.

```
SELECT AUTHORITY, D_USER, D_GROUP, D_PUBLIC, ROLE_USER, ROLE_GROUP, ROLE_PUBLIC, D_ROLE
FROM TABLE (SYSPROC.AUTH_LIST_AUTHORITIES_FOR_AUTHID ('PUBLIC', 'G') ) AS T
ORDER BY AUTHORITY
```

| 1 | D_USER | D_GROUP | D_PUBLIC | ROLE_USER | ROLE_GROUP | ROLE_PUBLIC | D_ROLE |
|-------------------------|--------|---------|----------|-----------|------------|-------------|--------|
| ACCESSCTRL | * | * | N | * | * | N | * |
| BINDADD | * | * | Y | * | * | N | * |
| CONNECT | * | * | Y | * | * | N | * |
| CREATE_EXTERNAL_ROUTINE | * | * | N | * | * | N | * |

| | | | | | | | |
|---------------------------|---|---|---|---|---|---|---|
| CREATE_NOT_FENCED_ROUTINE | * | * | N | * | * | N | * |
| CREATETAB | * | * | Y | * | * | N | * |
| DATAACCESS | * | * | N | * | * | N | * |
| DBADM | * | * | N | * | * | N | * |
| EXPLAIN | * | * | N | * | * | N | * |
| IMPLICIT_SCHEMA | * | * | Y | * | * | N | * |
| LOAD | * | * | N | * | * | N | * |
| QUIESCE_CONNECT | * | * | N | * | * | N | * |
| SECADM | * | * | N | * | * | N | * |
| SQLADM | * | * | N | * | * | N | * |
| SYSADM | * | * | * | * | * | * | * |
| SYSCTRL | * | * | * | * | * | * | * |
| SYSMAINT | * | * | * | * | * | * | * |
| SYSMON | * | * | * | * | * | * | * |
| WLMADM | * | * | N | * | * | N | * |

Ejemplo 4: Recuperar todas las autorizaciones otorgadas directamente al rol R1 o indirectamente a través de un rol. Tenga en cuenta que en este caso, también se ha otorgado el rol R2 al rol R1.

```
SELECT AUTHORITY, D_USER, D_GROUP, D_PUBLIC, ROLE_USER, ROLE_GROUP, ROLE_PUBLIC, D_ROLE
FROM TABLE (SYSPROC.AUTH_LIST_AUTHORITIES_FOR_AUTHID ('R1', 'R') ) AS T
ORDER BY AUTHORITY
```

| AUTHORITY | D_USER | D_GROUP | D_PUBLIC | ROLE_USER | ROLE_GROUP | ROLE_PUBLIC | D_ROLE |
|---------------------------|--------|---------|----------|-----------|------------|-------------|--------|
| ACCESSCTRL | * | * | * | * | * | * | N |
| BINDADD | * | * | * | * | * | * | N |
| CONNECT | * | * | * | * | * | * | N |
| CREATE_EXTERNAL_ROUTINE | * | * | * | * | * | * | N |
| CREATE_NOT_FENCED_ROUTINE | * | * | * | * | * | * | Y |
| CREATETAB | * | * | * | * | * | * | N |
| DATAACCESS | * | * | * | * | * | * | N |
| DBADM | * | * | * | * | * | * | N |
| EXPLAIN | * | * | * | * | * | * | N |
| IMPLICIT_SCHEMA | * | * | * | * | * | * | N |
| LOAD | * | * | * | * | * | * | Y |
| QUIESCE_CONNECT | * | * | * | * | * | * | N |
| SECADM | * | * | * | * | * | * | N |
| SYSADM | * | * | * | * | * | * | * |
| SQLADM | * | * | * | * | * | * | N |
| SYSCTRL | * | * | * | * | * | * | * |
| SYSMAINT | * | * | * | * | * | * | * |
| SYSMON | * | * | * | * | * | * | * |
| WLMADM | * | * | * | * | * | * | N |

Notas de uso

La salida de la función de tabla AUTH_LIST_AUTHORITIES_FOR_AUTHID depende de *tipoidaut*. Por ejemplo, para un *tipoidaut* del USER, devuelve todas las autorizaciones de que dispone *idaut* por cualquier medio:

- otorgada directamente a *idaut*
- otorgada a cualquier grupo (o roles otorgados al grupo) a quien pertenece el *idaut*
- otorgada a cualquier rol (o roles otorgados al grupo) otorgado a *idaut*
- otorgada a PUBLIC (o roles otorgados a PUBLIC)

Función de tabla AUTH_LIST_GROUPS_FOR_AUTHID – Recuperación lista de pertenencia al grupo para un ID de autorización determinado

La función de tabla AUTH_LIST_GROUPS_FOR_AUTHID devuelve la lista de grupos de los que es miembro el ID de autorización determinado.

Sintaxis

El esquema es SYSPROC.

Parámetro de la función de tabla

idaut

Es un argumento de entrada de tipo VARCHAR(128) que especifica el ID de autorización que se está consultando. El ID de autorización sólo puede representar a un usuario. Si *idaut* no existe, es una serie NULL o vacía, se devuelve una tabla de resultados vacía.

Autorización

Privilegio EXECUTE sobre la función de tabla AUTH_LIST_GROUPS_FOR_AUTHID.

Ejemplo

Recuperar todos los grupos a los que pertenece AMY.

```
SELECT * FROM TABLE (SYSPROC.AUTH_LIST_GROUPS_FOR_AUTHID('AMY')) AS T
```

El siguiente ejemplo muestra los datos de salida de la consulta.

```
GROUP
-----
BUILD
PDXDB2
```

2 registro(s) seleccionado(s).

Notas de uso

La información de grupo devuelta puede ser distinta de la esperada por los siguientes motivos:

- En un entorno Windows Active Directory, el gestor de bases de datos:
 - da soporte a un nivel de anidamiento de grupo dentro de un grupo local, excepto a la anidamiento de un grupo local de dominio dentro de un grupo local. Por ejemplo, si *idaut* pertenece al grupo global G1 y G1 pertenece al grupo local L1, el grupo local L1 se devuelve como el grupo correspondiente a *idaut*. Sin embargo, si *idaut* pertenece al grupo local de dominio DL1 y DL1 pertenece al grupo local L1, no se devuelve ninguna información de grupo para *idaut*.
 - no da soporte a la anidamiento de grupos globales. Por ejemplo, si *idaut* pertenece al grupo global G2 y G2 pertenece al grupo global G3, sólo se devuelve G2 como el grupo correspondiente a *idaut*.
- La variable de registro DB2_GRP_LOOKUP especifica qué mecanismo de seguridad de Windows se utiliza para enumerar los grupos a los que pertenece un usuario.
- Para un ID de autorización perteneciente a un determinado dominio, si no se especifica el dominio como parte de *idaut*, y existe un *idaut* local y de dominio con el mismo nombre, se devuelven los grupos correspondientes al ID de autorización local.

Información devuelta

Tabla 119. Información devuelta por la función de tabla AUTH_LIST_GROUPS_FOR_AUTHID

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| GROUP | VARCHAR(128) | El grupo al que pertenece el ID de autorización. |

Función AUTH_LIST_ROLES_FOR_AUTHID - Devuelve la lista de roles

La función AUTH_LIST_ROLES_FOR_AUTHID devuelve la lista de roles de la que es miembro el ID de autorización determinado.

Sintaxis

►► AUTH_LIST_ROLES_FOR_AUTHID (—*idaut*—, —*tipoaut*—) ◀◀

El esquema es SYSPROC.

Parámetros de la función de tabla

idaut

Es un argumento de entrada de tipo VARCHAR(128) que especifica el ID de autorización que se está consultando. El ID de autorización puede ser un usuario, grupo o rol. Si *idaut* es una serie NULL o vacía, se devuelve una tabla de resultados vacía.

tipoaut

Es un argumento de entrada de tipo VARCHAR(1) que especifica el ID de autorización que se está consultando. Si *tipoaut* no existe, es una serie NULL o vacía, se devuelve una tabla de resultados vacía. Valores posibles para *tipoaut*:

- G: Grupo
- R: Rol
- U: Usuario

Autorización

Privilegio EXECUTE sobre la función de tabla AUTH_LIST_ROLES_FOR_AUTHID.

Información devuelta

Tabla 120. Los conjuntos de resultados para AUTH_LIST_ROLES_FOR_AUTHID

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| GRANTOR | VARCHAR(128) | El que otorga el rol. |
| GRANTORTYPE | CHAR(1) | Tipo de otorgante: <ul style="list-style-type: none">• U = El otorgante es un usuario individual |
| GRANTEE | VARCHAR(128) | Usuario al que se ha otorgado el rol. |
| GRANTEETYPE | CHAR(1) | Tipo de otorgado: <ul style="list-style-type: none">• G = Se otorga a un grupo• R = Se otorga a un rol• U = El otorgado es un usuario |

Tabla 120. Los conjuntos de resultados para AUTH_LIST_ROLES_FOR_AUTHID (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| ROLENAME | VARCHAR(128) | Nombre del rol otorgado al ID de autorización directa o indirectamente mediante un grupo u otro rol. |
| CREATE_TIME | TIMESTAMP | Hora en que se creó el rol. |
| ADMIN | CHAR(1) | Privilegio para otorgar el rol, revocar el rol o comentar sobre un rol: <ul style="list-style-type: none"> • N = No se mantiene • Y = Se mantiene |

Ejemplo

Imagine que otorga el rol INTERN a un rol DOCTOR y el rol DOCTOR al rol SPECIALIST y, luego, el rol SPECIALIST al usuario ALICE. ALICE pertenece al grupo HOSPITAL y el rol EMPLOYEE está otorgado al grupo HOSPITAL. ALICE también pertenece al grupo especial PUBLIC. Además, el rol PATIENTS está otorgado a PUBLIC.

Ejemplo 1: Reúna todos los roles otorgados al usuario ALICE.

```
SELECT GRANTOR, GRANTORTYPE, GRANTEE, GRANTEETYPE, ROLENAME,
       CREATE_TIME, ADMIN
FROM TABLE (SYSPROC.AUTH_LIST_ROLES_FOR_AUTHID ('ALICE', 'U') ) AS T
```

El siguiente ejemplo muestra los datos de salida de la consulta.

| GRANTOR | GRANTORTYPE | GRANTEE | GRANTEETYPE | ROLENAME | CREATE_TIME | ADMIN |
|---------|-------------|------------|-------------|------------|----------------------------|-------|
| ZURBIE | U | DOCTOR | R | INTERN | 2006-08-01-15.09.58.537399 | N |
| ZURBIE | U | SPECIALIST | R | DOCTOR | 2006-08-01-15.10.04.540660 | N |
| ZURBIE | U | ALICE | U | SPECIALIST | 2006-08-01-15.10.08.776218 | N |
| ZURBIE | U | HOSPITAL | G | EMPLOYEE | 2006-08-01-15.10.14.277576 | N |
| ZURBIE | U | PUBLIC | G | PATIENTS | 2006-08-01-15.10.18.878609 | N |

5 registro(s) seleccionado(s).

Ejemplo 2: Reúna todos los roles otorgados al grupo HOSPITAL.

```
SELECT GRANTOR, GRANTORTYPE, GRANTEE, GRANTEETYPE, ROLENAME,
       CREATE_TIME, ADMIN
FROM TABLE (SYSPROC.AUTH_LIST_ROLES_FOR_AUTHID ('HOSPITAL', 'G') ) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| GRANTOR | GRANTORTYPE | GRANTEE | GRANTEETYPE | ROLENAME | CREATE_TIME | ADMIN |
|---------|-------------|----------|-------------|----------|----------------------------|-------|
| ZURBIE | U | HOSPITAL | G | EMPLOYEE | 2006-08-01-15.10.14.277576 | N |

1 registro(s) seleccionado(s).

Ejemplo 3: Reúna todos los roles otorgados al rol SPECIALIST.

```
SELECT GRANTOR, GRANTORTYPE, GRANTEE, GRANTEETYPE, ROLENAME,
       CREATE_TIME, ADMIN
FROM TABLE (SYSPROC.AUTH_LIST_ROLES_FOR_AUTHID ('SPECIALIST', 'R') ) AS T
```

El siguiente ejemplo muestra los datos de salida de la consulta.

| GRANTOR | GRANTORTYPE | GRANTEE | GRANTEETYPE | ROLENAME | CREATE_TIME | ADMIN |
|---------|-------------|------------|-------------|----------|----------------------------|-------|
| ZURBIE | U | DOCTOR | R | INTERN | 2006-08-01-15.09.58.537399 | N |
| ZURBIE | U | SPECIALIST | R | DOCTOR | 2006-08-01-15.10.04.540660 | N |

2 registro(s) seleccionado(s).

Ejemplo 4: Reúna todos los roles otorgados al grupo PUBLIC.

```

SELECT GRANTOR, GRANTORTYPE, GRANTEE, GRANTEETYPE, ROLENAME,
       CREATE_TIME, ADMIN
FROM TABLE (SYSPROC.AUTH_LIST_ROLES_FOR_AUTHID ('PUBLIC', 'G') ) AS T

```

El ejemplo siguiente muestra los datos de salida de la consulta.

| GRANTOR | GRANTORTYPE | GRANTEE | GRANTEETYPE | ROLENAME | CREATE_TIME | ADMIN |
|---------|-------------|---------|-------------|----------|----------------------------|-------|
| ZURBIE | U | PUBLIC | G | PATIENTS | 2006-08-01-15.10.18.878609 | N |

1 registro(s) seleccionado(s).

Notas de uso

La salida de la función de tabla AUTH_LIST_ROLES_FOR_AUTHID depende de AUTHIDTYPE:

- Para un usuario, devuelve los roles otorgados al usuario directa o indirectamente mediante otros roles o grupos a los que pertenece el usuario (o PUBLIC).
- Para un grupo, devuelve los roles otorgados al grupo directa o indirectamente mediante otros roles.
- Para un rol, devuelve los roles otorgados al rol directa o indirectamente mediante otros roles.

Vista administrativa AUTHORIZATIONIDS – Recuperar ID y tipos de autorización

La vista administrativa AUTHORIZATIONIDS devuelve una lista de ID de autorización a los que se han otorgado privilegios o autorizaciones, junto con sus tipos, para todos los ID de autorización definidos en los catálogos de sistemas desde la base de datos conectada actualmente. Si se han otorgado privilegios o autorizaciones a los grupos o roles, sólo se devolverán los nombres de los grupos o los roles.

El esquema es SYSIBMADM.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa AUTHORIZATIONIDS
- Privilegio CONTROL en la vista administrativa AUTHORIZATIONIDS
- Autorización DATAACCESS

Ejemplo

Recuperar todos los ID de autorización a los que se han otorgado privilegios o autorizaciones, junto con sus tipos.

```

SELECT * FROM SYSIBMADM.AUTHORIZATIONIDS

```

El siguiente ejemplo muestra los datos de salida de la consulta.

| AUTHID | AUTHIDTYPE |
|----------|------------|
| PUBLIC | G |
| JESSICAE | U |
| DOCTOR | R |

3 registro(s) seleccionado(s).

Información devuelta

Tabla 121. Información devuelta por la vista administrativa AUTHORIZATIONIDS

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| AUTHID | VARCHAR(128) | ID de autorización al que se han otorgado de forma explícita privilegios o autorizaciones. |
| AUTHIDTYPE | CHAR(1) | Tipo de ID de autorización: <ul style="list-style-type: none"> • U: usuario • R: rol • G: grupo |

Vista administrativa OBJECTOWNERS – Recuperar información sobre propiedad de objetos

La vista administrativa OBJECTOWNERS devuelve toda la información sobre la propiedad del objeto para cada ID de autorización del tipo USER que posee un objeto y que está definido en los catálogos de sistemas desde la base de datos conectada actualmente.

El esquema es SYSIBMADM.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa OBJECTOWNERS
- Privilegio CONTROL en la vista administrativa OBJECTOWNERS
- Autorización DATAACCESS

Ejemplo

Recuperar toda la información sobre la propiedad del objeto para el esquema de objeto 'THERESAX'.

```
SELECT SUBSTR(OWNER,1,10) AS OWNER, OWNERTYPE,
       SUBSTR(OBJECTNAME,1,30) AS OBJECTNAME,
       SUBSTR(OBJECTSCHEMA,1,10) AS OBJECTSCHEMA, OBJECTTYPE
FROM SYSIBMADM.OBJECTOWNERS WHERE OBJECTSCHEMA='THERESAX'
```

El siguiente ejemplo muestra los datos de salida de la consulta.

| OWNER | OWNERTYPE | OBJECTNAME | OBJECTSCHEMA | OBJECTTYPE |
|----------|-----------|-------------------|--------------|------------------|
| THERESAX | U | MIN_SALARY | THERESAX | TRIGGER |
| THERESAX | U | POLICY_IR | SYSTOOLS | TRIGGER |
| THERESAX | U | CUSTOMER | THERESAX | XML SCHEMA |
| THERESAX | U | DB2DETAILDEADLOCK | | EVENTMONITORS |
| THERESAX | U | SAMPSEQUENCE | THERESAX | SEQUENCE |
| THERESAX | U | SQL0F00 | NULLID | PACKAGE |
| ... | | | | |
| THERESAX | U | HI_OBJ_UNIQ | SYSTOOLS | TABLE CONSTRAINT |

257 registro(s) seleccionado(s).

Información devuelta

Tabla 122. Información devuelta por la vista administrativa OBJECTOWNERS

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| OWNER | VARCHAR(128) | ID de autorización que posee este objeto. |
| OWNERTYPE | VARCHAR(1) | Tipo de ID de autorización: • U: usuario |
| OBJECTNAME | VARCHAR(128) | Nombre del objeto de base de datos. |
| OBJECTSCHEMA | VARCHAR(128) | Esquema del objeto de base de datos. |
| OBJECTTYPE | VARCHAR(24) | Tipo de objeto de base de datos. |

Vista administrativa PRIVILEGES – Recuperar información sobre privilegios

La vista administrativa PRIVILEGES devuelve todos los privilegios explícitos para todos los ID de autorización definidos en los catálogos de sistemas desde la base de datos conectada actualmente.

El esquema es SYSIBMADM.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT sobre la vista administrativa PRIVILEGES
- Privilegio CONTROL sobre la vista administrativa PRIVILEGES
- Autorización DATAACCESS

Ejemplo

Recuperar el privilegio otorgado, junto con el nombre de objeto, esquema y tipo, correspondiente a todos los ID de autorización.

```
SELECT AUTHID, PRIVILEGE, OBJECTNAME, OBJECTSCHEMA, OBJECTTYPE  
FROM SYSIBMADM.PRIVILEGES
```

El siguiente ejemplo muestra los datos de salida de la consulta.

| AUTHID | PRIVILEGE | OBJECTNAME | OBJECTSCHEMA | OBJECTTYPE |
|----------|-----------|--------------------|--------------|------------|
| JESSICAE | EXECUTE | SQL0F00 | NULLID | PACKAGE |
| PUBLIC | EXECUTE | SYSSH201 | NULLID | PACKAGE |
| JESSICAE | EXECUTE | SYSSH202 | NULLID | PACKAGE |
| PUBLIC | EXECUTE | SYSSH202 | NULLID | PACKAGE |
| DOCTOR | EXECUTE | PKG0123 | NULLID | PACKAGE |
| ... | | | | |
| PUBLIC | EXECUTE | SQL051109185227800 | SYSPROC | FUNCTION |
| JESSICAE | EXECUTE | SQL051109185227801 | SYSPROC | FUNCTION |
| PUBLIC | EXECUTE | SQL051109185227801 | SYSPROC | FUNCTION |
| JESSICAE | EXECUTE | SQL051109185227838 | SYSPROC | FUNCTION |
| PUBLIC | EXECUTE | SQL051109185227838 | SYSPROC | FUNCTION |
| ... | | | | |
| PUBLIC | EXECUTE | LIST_SRVR_TYPES | SYSPROC | PROCEDURE |

| | | | | |
|--------|----------|--------------------|----------|-----------|
| PUBLIC | EXECUTE | LIST_SRVR_VERSIONS | SYSPROC | PROCEDURE |
| PUBLIC | EXECUTE | LIST_WRAP_OPTIONS | SYSPROC | PROCEDURE |
| PUBLIC | EXECUTE | LIST_SRVR_OPTIONS | SYSPROC | PROCEDURE |
| ... | | | | |
| SYSTEM | | POLICY_UNQ | SYSTOOLS | INDEX |
| PUBLIC | CREATEIN | | NULLID | SCHEMA |
| PUBLIC | UPDATE | COLUMNS | SYSSTAT | VIEW |
| PUBLIC | UPDATE | COLGROUPS | SYSSTAT | VIEW |
| ... | | | | |

Información devuelta

Tabla 123. Información devuelta por la vista administrativa *PRIVILEGES*

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| AUTHID | VARCHAR(128) | ID de autorización al que se ha otorgado de forma explícita este privilegio. |
| AUTHIDTYPE | CHAR(1) | Tipo de ID de autorización: <ul style="list-style-type: none"> • U: usuario • R: rol • G: grupo |
| PRIVILEGE | VARCHAR(11) | Privilegio que se ha otorgado de forma explícita a este ID de autorización. |
| GRANTABLE | VARCHAR(1) | Indica si el privilegio se puede o no otorgar: <ul style="list-style-type: none"> • Y: Se puede otorgar • N: No se puede otorgar |
| OBJECTNAME | VARCHAR(128) | Nombre del objeto de base de datos. |
| OBJECTSCHEMA | VARCHAR(128) | Esquema del objeto de base de datos. |
| OBJECTTYPE | VARCHAR(24) | Tipo de objeto de base de datos. |

Capítulo 14. Rutinas y vistas de instantáneas

Vista administrativa APPL_PERFORMANCE – Recuperar porcentaje de filas seleccionadas para una aplicación

La vista administrativa APPL_PERFORMANCE visualiza información sobre el porcentaje de filas seleccionadas por una aplicación. La información devuelta corresponde a todas las particiones de base de datos para la base de datos conectada actualmente. Esta vista se puede utilizar para buscar las aplicaciones que puedan estar realizando exploraciones de tablas grandes o para buscar las consultas potencialmente problemáticas.

El esquema es SYSIBMADM.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa APPL_PERFORMANCE
- Privilegio CONTROL en la vista administrativa APPL_PERFORMANCE
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMMAINT
- SYSADM

Ejemplo

Recuperar el informe sobre rendimiento de las aplicaciones.

```
SELECT SNAPSHOT_TIMESTAMP, SUBSTR(AUTHID,1,10) AS AUTHID,  
       SUBSTR(APPL_NAME,1,10) AS APPL_NAME,AGENT_ID,  
       PERCENT_ROWS_SELECTED, DBPARTITIONNUM  
FROM SYSIBMADM.APPL_PERFORMANCE
```

El siguiente ejemplo muestra los datos de salida de la consulta.

```
SNAPSHOT_TIMESTAMP      AUTHID      APPL_NAME ...  
-----  
2006-01-07-17.01.15.966668 JESSICAE  db2bp.exe ...  
2006-01-07-17.01.15.980278 JESSICAE  db2taskd ...  
2006-01-07-17.01.15.980278 JESSICAE  db2bp.exe ...  
                                     ...  
3 registro(s) seleccionado(s).           ...
```

Datos de salida de esta consulta (continuación).

```
... AGENT_ID      PERCENT_ROWS_SELECTED DBPARTITIONNUM  
... -----  
...          67                -                1  
...          68                -                0  
...          67                57.14           0  
...
```

Información devuelta

Tabla 124. Información devuelta por la vista administrativa APPL_PERFORMANCE

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| AUTHID | VARCHAR(128) | auth_id - ID de autorización |
| APPL_NAME | VARCHAR(256) | appl_name - Nombre de aplicación |
| AGENT_ID | BIGINT | agent_id - Descriptor de contexto de aplicación (ID de agente) |
| PERCENT_ROWS_SELECTED | DECIMAL(5,2) | El porcentaje de las filas leídas del disco que se han devuelto a la aplicación. Nota: El porcentaje mostrado no será mayor que 100,00 (100%). |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa APPLICATIONS - Recuperar información de aplicación de base de datos conectada

La vista administrativa APPLICATIONS devuelve información sobre las aplicaciones de base de datos conectadas. La vista es una interfaz de SQL para el mandato CLP LIST APPLICATIONS SHOW DETAIL, pero solamente para la base de datos conectada actualmente. Su información se basa en la vista administrativa SNAPAPPL_INFO.

El esquema es SYSIBMADM.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa APPLICATIONS
- Privilegio CONTROL en la vista administrativa APPLICATIONS
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYCTRL
- SYSMOINT
- SYSADM

Ejemplo

Ejemplo 1: Listar la información correspondiente a todas las aplicaciones activas en la base de datos de una sola partición denominada SAMPLE.

```
SELECT AGENT_ID, SUBSTR(APPL_NAME,1,10) AS APPL_NAME, AUTHID,  
       APPL_STATUS FROM SYSIBMADM.APPLICATIONS WHERE DB_NAME = 'SAMPLE'
```

El siguiente ejemplo muestra los datos de salida de la consulta.

```
AGENT_ID          APPL_NAME  AUTHID  APPL_STATUS
-----
                23 db2bp.exe  JESSICAE  UOWEXEC
```

1 registro(s) seleccionado(s).

Ejemplo 2: Listar el número de agentes por aplicación en la partición de base de datos 0 para la base de datos de varias particiones denominada SAMPLE.

```
SELECT SUBSTR(APPL_NAME, 1, 10) AS APPL_NAME, COUNT(*) AS NUM
  FROM SYSIBMADM.APPLICATIONS WHERE DBPARTITIONNUM = 0
 AND DB_NAME = 'SAMPLE' GROUP BY APPL_NAME
```

El siguiente ejemplo muestra los datos de salida de la consulta.

```
APPL_NAME  NUM
-----
db2bp.exe      3
javaw.exe     1
```

2 registro(s) seleccionado(s).

Notas de uso

La vista no da soporte a la sintaxis GLOBAL disponible en CLP. Sin embargo, se puede realizar una agregación mediante las funciones de agregación de SQL a medida que se devuelven desde la vista los datos de todas las particiones de base de datos.

Información devuelta

Tabla 125. Información devuelta por la vista administrativa APPLICATIONS

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| CLIENT_DB_ALIAS | VARCHAR(128) | client_db_alias - Alias de base de datos utilizado por aplicación |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| AGENT_ID | BIGINT | agent_id - Descriptor de contexto de aplicación (ID de agente) |
| APPL_NAME | VARCHAR(256) | appl_name - Nombre de aplicación |
| AUTHID | VARCHAR(128) | auth_id - ID de autorización |
| APPL_ID | VARCHAR(128) | appl_id - ID de aplicación |

Tabla 125. Información devuelta por la vista administrativa APPLICATIONS (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|--|
| APPL_STATUS | VARCHAR(22) | appl_status - Estado de aplicación. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • BACKUP • COMMIT_ACT • COMP • CONNECTED • CONNECTPEND • CREATE_DB • DECOUPLED • DISCONNECTPEND • INTR • IOERROR_WAIT • LOAD • LOCKWAIT • QUIESCE_TABLESPACE • RECOMP • REMOTE_RQST • RESTART • RESTORE • ROLLBACK_ACT • ROLLBACK_TO_SAVEPOINT • TEND • THABRT • THCOMT • TPREP • UNLOAD • UOWEXEC • UOWWAIT • WAITFOR_REMOTE |
| STATUS_CHANGE_TIME | TIMESTAMP | status_change_time - Hora de cambio de estado de aplicación |
| SEQUENCE_NO | VARCHAR(4) | sequence_no - Número de secuencia |
| CLIENT_PRDID | VARCHAR(128) | client_prdid - ID de producto/versión de cliente |
| CLIENT_PID | BIGINT | client_pid - ID de proceso de cliente |

Tabla 125. Información devuelta por la vista administrativa APPLICATIONS (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|---|
| CLIENT_PLATFORM | VARCHAR(12) | <p>client_platform - Plataforma operativa de cliente. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes:</p> <ul style="list-style-type: none"> • AIX • AIX64 • AS400_DRDA • DOS • DYNIX • HP • HP64 • HPIA • HPIA64 • LINUX • LINUX390 • LINUXIA64 • LINUXPPC • LINUXPPC64 • LINUXX8664 • LINUXZ64 • MAC • MVS_DRDA • NT • NT64 • OS2 • OS390 • SCO • SGI • SNI • SUN • SUN64 • UNKNOWN • UNKNOWN_DRDA • VM_DRDA • VSE_DRDA • WINDOWS • WINDOWS95 |

Tabla 125. Información devuelta por la vista administrativa APPLICATIONS (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------|---------------|---|
| CLIENT_PROTOCOL | VARCHAR(10) | client_protocol - Protocolo de comunicaciones de cliente. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h, <ul style="list-style-type: none"> • CPIC • LOCAL • NETBIOS • NPIPE • TCPIP (para DB2 UDB) • TCPIP4 • TCPIP6 |
| CLIENT_NNAME | VARCHAR(128) | El elemento del supervisor client_nname ha quedado obsoleto. El valor devuelto no es un valor válido. |
| COORD_NODE_NUM | SMALLINT | coord_node - Nodo de coordinación |
| COORD_AGENT_PID | BIGINT | coord_agent_pid - Agente coordinador |
| NUM_ASSOC_AGENTS | BIGINT | num_assoc_agents - Número de agentes asociados |
| TPMON_CLIENT_USERID | VARCHAR(256) | tpmon_client_userid - ID de usuario de cliente de supervisor de TP |
| TPMON_CLIENT_WKSTN | VARCHAR(256) | tpmon_client_wkstn - Nombre de estación de trabajo de cliente de supervisor de TP |
| TPMON_CLIENT_APP | VARCHAR(256) | tpmon_client_app - Nombre de aplicación de cliente de supervisor de TP |
| TPMON_ACC_STR | VARCHAR(200) | tpmon_acc_str - Serie de contabilidad de cliente de supervisor de TP |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa BP_HITRATIO – Recuperar información sobre la tasa de aciertos de agrupación de almacenamientos intermedios

La vista administrativa BP_HITRATIO devuelve las tasas de aciertos de agrupación de almacenamientos intermedios, que incluye la tasa de aciertos total, la tasa de aciertos de datos, la tasa de aciertos de XDA y la tasa de aciertos de índice, correspondientes a todas las agrupaciones de almacenamientos intermedios y todas las particiones de base de datos existentes en la base de datos conectada actualmente.

El esquema es SYSIBMADM.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa BP_HITRATIO
- Privilegio CONTROL en la vista administrativa BP_HITRATIO
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar un informe correspondiente a todas las agrupaciones de almacenamientos intermedios en la base de datos conectada.

```
SELECT SUBSTR(DB_NAME,1,8) AS DB_NAME, SUBSTR(BP_NAME,1,14) AS BP_NAME,  
       TOTAL_HIT_RATIO_PERCENT, DATA_HIT_RATIO_PERCENT,  
       INDEX_HIT_RATIO_PERCENT, XDA_HIT_RATIO_PERCENT, DBPARTITIONNUM  
FROM SYSIBMADM.BP_HITRATIO ORDER BY DBPARTITIONNUM
```

El siguiente ejemplo muestra los datos de salida de la consulta.

| DB_NAME | BP_NAME | TOTAL_HIT_RATIO_PERCENT | DATA_HIT_RATIO_PERCENT | ... |
|---------|----------------|-------------------------|------------------------|-----|
| TEST | IBMDEFAULTBP | 63.09 | 68.94 | ... |
| TEST | IBMSYSTEMBP4K | - | - | ... |
| TEST | IBMSYSTEMBP8K | - | - | ... |
| TEST | IBMSYSTEMBP16K | - | - | ... |
| TEST | IBMSYSTEMBP32K | - | - | ... |

Datos de salida de esta consulta (continuación).

| ... | INDEX_HIT_RATIO_PERCENT | XDA_HIT_RATIO_PERCENT | DBPARTITIONNUM |
|-----|-------------------------|-----------------------|----------------|
| ... | 43.20 | - | 0 |
| ... | - | - | 0 |
| ... | - | - | 0 |
| ... | - | - | 0 |
| ... | - | - | 0 |

Notas de uso

La proporción de lecturas físicas respecto a las lecturas lógicas proporciona la tasa de aciertos de la agrupación de almacenamientos intermedios. Cuanto más baja sea la tasa de aciertos, más datos se leen del disco, en lugar de leerse de la agrupación de almacenamientos intermedios de la antememoria, con lo que la operación puede resultar más costosa.

Información devuelta

Tabla 126. Información devuelta por la vista administrativa BP_HITRATIO

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | Fecha y hora en la que se solicitó el informe. |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| BP_NAME | VARCHAR(128) | bp_name - Nombre de la agrupación de almacenamientos intermedios. |
| TOTAL_LOGICAL_READS | BIGINT | Lecturas lógicas totales (índice, XDA y datos) en la agrupación de almacenamientos intermedios. |
| TOTAL_PHYSICAL_READS | BIGINT | Lecturas físicas totales (índice, XDA y datos) en la agrupación de almacenamientos intermedios. |
| TOTAL_HIT_RATIO_PERCENT | DECIMAL(5,2) | Tasa de aciertos total (índice, XDA y datos). |
| DATA_LOGICAL_READS | BIGINT | pool_data_l_reads - Lecturas lógicas de datos de la agrupación de almacenamientos intermedios |
| DATA_PHYSICAL_READS | BIGINT | pool_data_p_reads - Lecturas físicas de datos de la agrupación de almacenamientos intermedios |
| DATA_HIT_RATIO_PERCENT | DECIMAL(5,2) | Tasa de aciertos de datos. |
| INDEX_LOGICAL_READS | BIGINT | pool_index_l_reads - Lecturas lógicas de índices de la agrupación de almacenamientos intermedios |
| INDEX_PHYSICAL_READS | BIGINT | pool_index_p_reads - Lecturas físicas de índices de la agrupación de almacenamientos intermedios |
| INDEX_HIT_RATIO_PERCENT | DECIMAL(5,2) | Tasa de aciertos de índice. |
| XDA_LOGICAL_READS | BIGINT | pool_xda_l_reads - Lecturas lógicas de datos de XDA de la agrupación de almacenamientos intermedios |
| XDA_PHYSICAL_READS | BIGINT | pool_xda_p_reads - Lecturas físicas de datos de XDA de la agrupación de almacenamientos intermedios |
| XDA_HIT_RATIO_PERCENT | DECIMAL(5,2) | Tasa de aciertos de objetos de almacenamiento auxiliar. |
| DBPARTITIONNUM | SMALLINT | La partición de base de datos desde la que se han recuperado los datos para la fila. |

Vista administrativa BP_READ_IO – Recuperar información de rendimiento de lectura de las agrupaciones de almacenamientos intermedios

La vista administrativa BP_READ_IO devuelve información de rendimiento de lectura de las agrupaciones de almacenamientos intermedios. Esta vista se puede utilizar para observar cada agrupación de almacenamientos intermedios con el fin de comprobar la eficacia de la captación previa.

El esquema es SYSIBMADM.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa BP_READ_IO
- Privilegio CONTROL en la vista administrativa BP_READ_IO
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMMAINT
- SYSADM

Ejemplo

Recuperar las lecturas físicas totales y el tiempo medio de lectura correspondientes a todas las agrupaciones de almacenamientos intermedios de todas las particiones de la base de datos conectada actualmente.

```
SELECT SUBSTR(BP_NAME, 1, 15) AS BP_NAME, TOTAL_PHYSICAL_READS,  
       AVERAGE_READ_TIME_MS, DBPARTITIONNUM  
FROM SYSIBMADM.BP_READ_IO ORDER BY DBPARTITIONNUM
```

El siguiente ejemplo muestra los datos de salida de la consulta.

| BP_NAME | TOTAL_PHYSICAL_READS | AVERAGE_READ_TIME_MS | DBPARTITIONNUM |
|----------------|----------------------|----------------------|----------------|
| IBMDEFAULTBP | 811 | 4 | 0 |
| IBMSYSTEMBP4K | 0 | - | 0 |
| IBMSYSTEMBP8K | 0 | - | 0 |
| IBMSYSTEMBP16K | 0 | - | 0 |
| IBMDEFAULTBP | 34 | 0 | 1 |
| IBMSYSTEMBP4K | 0 | - | 1 |
| IBMSYSTEMBP8K | 0 | - | 1 |
| IBMDEFAULTBP | 34 | 0 | 2 |
| IBMSYSTEMBP4K | 0 | - | 2 |
| IBMSYSTEMBP8K | 0 | - | 2 |

10 registro(s) seleccionado(s).

Información devuelta

Tabla 127. Información devuelta por la vista administrativa BP_READ_IO

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|----------------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | Fecha y hora en que se generó el informe. |
| BP_NAME | VARCHAR(128) | bp_name - Nombre de la agrupación de almacenamientos intermedios. |
| TOTAL_PHYSICAL_READS | BIGINT | Lecturas físicas totales. |
| AVERAGE_READ_TIME_MS | BIGINT | Tiempo medio de lectura en milisegundos. |
| TOTAL_ASYNC_READS | BIGINT | Lecturas asíncronas totales. |
| AVERAGE_ASYNC_READ_TIME_MS | BIGINT | Tiempo medio de lectura asíncrona en milisegundos. |
| TOTAL_SYNC_READS | BIGINT | Lecturas síncronas totales. |
| AVERAGE_SYNC_READ_TIME_MS | BIGINT | Tiempo medio de lectura síncrona en milisegundos. |
| PERCENT_SYNC_READS | DECIMAL(5,2) | Porcentaje de páginas leídas de forma síncrona sin captación previa. Si muchas de las aplicaciones leen datos de forma síncrona sin captación previa, es posible que el sistema no esté ajustado de forma óptima. |

Tabla 127. Información devuelta por la vista administrativa BP_READ_IO (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------|---------------|---|
| ASYNC_NOT_READ_PERCENT | DECIMAL(5,2) | Porcentaje de páginas leídas del disco de forma asíncrona, a las que nunca ha accedido una consulta. Si se leen demasiadas páginas de forma asíncrona del disco y se colocan en la agrupación de almacenamientos intermedios, pero ninguna consulta ha accedido nunca a estas páginas, la captación previa podría reducir el rendimiento. |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa BP_WRITE_IO – Recuperar información de rendimiento de grabación de las agrupaciones de almacenamientos intermedios

La vista administrativa BP_WRITE_IO devuelve información sobre el rendimiento de grabación de cada agrupación de almacenamientos intermedios.

El esquema es SYSIBMADM.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa BP_WRITE_IO
- Privilegio CONTROL en la vista administrativa BP_WRITE_IO
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMMAINT
- SYSADM

Ejemplo

Recuperar las grabaciones totales y el tiempo medio de grabación correspondientes a todas las agrupaciones de almacenamientos intermedios de todas las particiones de la base de datos conectada actualmente.

```
SELECT SUBSTR(BP_NAME, 1, 15) AS BP_NAME, TOTAL_WRITES,
       AVERAGE_WRITE_TIME_MS, DBPARTITIONNUM
FROM SYSIBMADM.BP_WRITE_IO ORDER BY DBPARTITIONNUM
```

El siguiente ejemplo muestra los datos de salida de la consulta.

| BP_NAME | TOTAL_WRITES | AVERAGE_WRITE_TIME_MS | DBPARTITIONNUM |
|----------------|--------------|-----------------------|----------------|
| IBMDEFAULTBP | 11 | 5 | 0 |
| IBMSYSTEMBP4K | 0 | - | 0 |
| IBMSYSTEMBP8K | 0 | - | 0 |
| IBMSYSTEMBP16K | 0 | - | 0 |
| IBMSYSTEMBP32K | 0 | - | 0 |
| IBMDEFAULTBP | 0 | - | 1 |
| IBMSYSTEMBP4K | 0 | - | 1 |
| IBMSYSTEMBP8K | 0 | - | 1 |
| IBMDEFAULTBP | 0 | - | 2 |
| IBMSYSTEMBP4K | 0 | - | 2 |
| IBMSYSTEMBP8K | 0 | - | 2 |

11 registro(s) seleccionados.

Información devuelta

Tabla 128. Información devuelta por la vista administrativa BP_WRITE_IO

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | Fecha y hora en que se generó el informe. |
| BP_NAME | VARCHAR(128) | bp_name - Nombre de la agrupación de almacenamientos intermedios. |
| TOTAL_WRITES | BIGINT | Grabaciones totales. |
| AVERAGE_WRITE_TIME_MS | BIGINT | Tiempo medio de grabación en milisegundos. |
| TOTAL_ASYNC_WRITES | BIGINT | Grabaciones asíncronas totales. |
| PERCENT_WRITES_ASYNC | BIGINT | Porcentaje de grabaciones que son asíncronas. |
| AVERAGE_ASYNC_WRITE_TIME_MS | BIGINT | Tiempo medio de grabación asíncrona en milisegundos. |
| TOTAL_SYNC_WRITES | BIGINT | Grabaciones síncronas totales. |
| AVERAGE_SYNC_WRITE_TIME_MS | BIGINT | Tiempo medio de grabación síncrona en milisegundos. |

Tabla 128. Información devuelta por la vista administrativa BP_WRITE_IO (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|--|
| DBPARTITIONNUM | SMALLINT | La partición de base de datos desde la que se han recuperado los datos para la fila. |

Vista administrativa CONTAINER_UTILIZATION - Recuperar información de utilización y contenedor de espacio de tablas

La vista administrativa CONTAINER_UTILIZATION devuelve información acerca de los contenedores de espacios de tablas y los índices de utilización. La vista es una interfaz de SQL para el mandato CLP LIST TABLESPACE CONTAINERS. Su información se basa en la vista administrativa SNAPCONTAINER.

El esquema es SYSIBMADM.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa CONTAINER_UTILIZATION
- Privilegio CONTROL en la vista administrativa CONTAINER_UTILIZATION
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMANT
- SYSADM

Ejemplo

Recuperar una lista de todos los contenedores de espacios de tablas de la base de datos de una sola partición conectada, con información acerca del total de páginas y de las páginas utilizables así como su estado de accesibilidad.

```
SELECT SUBSTR(TBSP_NAME,1,20) AS TBSP_NAME, INT(TBSP_ID) AS TBSP_ID,
       SUBSTR(CONTAINER_NAME,1,45) AS CONTAINER_NAME, INT(CONTAINER_ID)
       AS CONTAINER_ID, CONTAINER_TYPE, INT(TOTAL_PAGES) AS TOTAL_PAGES,
       INT(USABLE_PAGES) AS USABLE_PAGES, ACCESSIBLE
FROM SYSIBMADM.CONTAINER_UTILIZATION
```

El siguiente ejemplo muestra los datos de salida de la consulta.

| TBSP_NAME | TBSP_ID | CONTAINER_NAME | ... |
|-------------|---------|-------------------------------------|-----|
| SYSCATSPACE | 0 | D:\DB2\NODE0000\SQL00001\SQLT0000.0 | ... |
| TEMPSPACE1 | 1 | D:\DB2\NODE0000\SQL00001\SQLT0001.0 | ... |
| USERSPACE1 | 2 | D:\DB2\NODE0000\SQL00001\SQLT0002.0 | ... |

```

SYSTOOLSPACE          3 D:\DB2\NODE0000\SQL00001\SYSTOOLSPACE    ...
SYSTOOLSTMPSPACE     4 D:\DB2\NODE0000\SQL00001\SYSTOOLSTMPSPACE ...

```

5 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```

... CONTAINER_ID CONTAINER_TYPE TOTAL_PAGES USABLE_PAGES ACCESSIBLE
... -----
...          0 PATH                0            0            1
...          0 PATH                0            0            1
...          0 PATH                0            0            1
...          0 PATH                0            0            1
...          0 PATH                0            0            1

```

Información devuelta

Para que se devuelva la información del sistema de archivos, debe activarse el conmutador de supervisor de instantáneas BUFFERPOOL.

Tabla 129. Información devuelta por la vista administrativa CONTAINER_UTILIZATION

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| TBSP_NAME | VARCHAR(128) | tablespace_name - Nombre de espacio de tablas |
| TBSP_ID | BIGINT | tablespace_id - Identificación de espacio de tablas |
| CONTAINER_NAME | VARCHAR(256) | container_name - Nombre de contenedor |
| CONTAINER_ID | BIGINT | container_id - Identificación de contenedor |
| CONTAINER_TYPE | VARCHAR(16) | container_type - Tipo de contenedor. Es un identificador de texto que se basa en las sentencias define de sqlutil.h y es uno de los siguientes: <ul style="list-style-type: none"> • DISK_EXTENT_TAG • DISK_PAGE_TAG • FILE_EXTENT_TAG • FILE_PAGE_TAG • PATH |
| TOTAL_PAGES | BIGINT | container_total_pages - Total de páginas en contenedor |
| USABLE_PAGES | BIGINT | container_usable_pages - Páginas utilizables en contenedor |
| ACCESSIBLE | SMALLINT | container_accessible - Accesibilidad de contenedor |
| STRIPE_SET | BIGINT | container_stripe_set - Conjunto de bandas |
| FS_ID | VARCHAR(22) | fs_id - Número de identificación del sistema de archivos exclusivo |

Tabla 129. Información devuelta por la vista administrativa
CONTAINER_UTILIZATION (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|---|
| FS_TOTAL_SIZE_KB | BIGINT | fs_total_size - Tamaño total de un sistema de archivos. Esta interfaz devuelve un valor en KB. |
| FS_USED_SIZE_KB | BIGINT | fs_used_size - Cantidad de espacio utilizada por un sistema de archivos. Esta interfaz devuelve un valor en KB. |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa LOCKS_HELD – Recuperar información acerca de los bloqueos retenidos

La vista administrativa LOCKS_HELD devuelve información sobre los bloqueos actuales.

El esquema es SYSIBMADM.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa LOCKS_HELD
- Privilegio CONTROL en la vista administrativa LOCKS_HELD
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCtrl
- SYSMAINT
- SYSADM

Ejemplo

Ejemplo 1: Listar el número total de bloqueos realizados por cada tabla en la base de datos SAMPLE.

```
SELECT TABSCHEMA, TABNAME, COUNT(*) AS NUMBER_OF_LOCKS_HELD
FROM SYSIBMADM.LOCKS_HELD WHERE DB_NAME = 'SAMPLE'
GROUP BY DBPARTITIONNUM, TABSCHEMA, TABNAME
```

El siguiente ejemplo muestra los datos de salida de la consulta.

| TABSCHEMA | TABNAME | NUMBER_OF_LOCKS_HELD |
|-----------|------------|----------------------|
| JESSICAE | EMPLOYEE | 5 |
| JESSICAE | EMP_RESUME | 1 |
| JESSICAE | ORG | 3 |

Ejemplo 2: Listar todos los bloqueos que no se han reajustado en la base de datos conectada actualmente (SAMPLE).

```
SELECT AGENT_ID, TABSCHEMA, TABNAME, LOCK_OBJECT_TYPE, LOCK_MODE,
       LOCK_STATUS FROM SYSIBMADM.LOCKS_HELD WHERE LOCK_ESCALATION = 0
       AND DBPARTITIONNUM = 0
```

El siguiente ejemplo muestra los datos de salida de la consulta.

```
AGENT_ID    TABSCHEMA    TABNAME      LOCK_OBJECT_TYPE  LOCK_MODE  LOCK_STATUS
-----
        680 JESSICAE     EMPLOYEE     INTERNALV_LOCK    S          GRNT
        680 JESSICAE     EMPLOYEE     INTERNALP_LOCK    S          GRNT
```

Ejemplo 3: Listar la información acerca de los bloqueos mantenidos en este momento por la aplicación con el ID de agente 310.

```
SELECT TABSCHEMA, TABNAME, LOCK_OBJECT_TYPE, LOCK_MODE, LOCK_STATUS,
       LOCK_ESCALATION FROM SYSIBMADM.LOCKS_HELD WHERE AGENT_ID = 310
```

El siguiente ejemplo muestra los datos de salida de la consulta.

```
TABSCHEMA    TABNAME      LOCK_OBJECT_TYPE  LOCK_MODE  LOCK_STATUS
-----
JESSICAE     EMP_RESUME   TABLE_LOCK      S          GRNT
JESSICAE     EMPLOYEE     ROW_LOCK         S          GRNT
```

Información devuelta

Tabla 130. Información devuelta por la vista administrativa LOCKS_HELD

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | Fecha y hora en que se generó el informe. |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| AGENT_ID | BIGINT | agent_id - Descriptor de contexto de aplicación (ID de agente) |
| APPL_NAME | VARCHAR(256) | appl_name - Nombre de aplicación |
| AUTHID | VARCHAR(128) | auth_id - ID de autorización |
| TBSP_NAME | VARCHAR(128) | tablespace_name - Nombre de espacio de tablas |
| TABSCHEMA | VARCHAR(128) | table_schema - Nombre de esquema de tabla |
| TABNAME | VARCHAR(128) | table_name - Nombre de tabla |
| TAB_FILE_ID | BIGINT | table_file_id - Identificación de archivo de tabla |

Tabla 130. Información devuelta por la vista administrativa LOCKS_HELD (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|--|
| LOCK_OBJECT_TYPE | VARCHAR(18) | lock_object_type - Tipo de objeto de bloqueo esperado. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • AUTORESIZE_LOCK • AUTOSTORAGE_LOCK • BLOCK_LOCK • EOT_LOCK • INPLACE_REORG_LOCK • INTERNAL_LOCK • INTERNALB_LOCK • INTERNALC_LOCK • INTERNALJ_LOCK • INTERNALL_LOCK • INTERNALO_LOCK • INTERNALQ_LOCK • INTERNALP_LOCK • INTERNALS_LOCK • INTERNALT_LOCK • INTERNALV_LOCK • KEYVALUE_LOCK • ROW_LOCK • SYSBOOT_LOCK • TABLE_LOCK • TABLE_PART_LOCK • TABLESPACE_LOCK • XML_PATH_LOCK |
| LOCK_NAME | VARCHAR(32) | lock_name - Nombre de bloqueo |
| LOCK_MODE | VARCHAR(10) | lock_mode - Modalidad de bloqueo. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • IN • IS • IX • NON (si no existe bloqueo) • NS • NW • S • SIX • U • X • Z |

Tabla 130. Información devuelta por la vista administrativa LOCKS_HELD (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|--|
| LOCK_STATUS | VARCHAR(10) | lock_status - Estado de bloqueo. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • CONV • GRNT |
| LOCK_ESCALATION | SMALLINT | lock_escalation - Escalamiento de bloqueos |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa LOCKWAITS – Recuperar información de bloqueos actuales en espera de ser activados

La vista administrativa LOCKWAITS devuelve información acerca de los agentes de DB2 que se ejecutan en nombre de las aplicaciones que están a la espera de obtener bloqueos.

El esquema es SYSIBMADM.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa LOCKWAITS
- Privilegio CONTROL en la vista administrativa LOCKWAITS
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYCTRL
- SYSMAINT
- SYSADM

Ejemplos

Ejemplo 1: Listar la información correspondiente a todas las peticiones de bloqueo en espera de la aplicación con el ID de agente 89.

```
SELECT SUBSTR(TABSCHEMA,1,8) AS TABSCHEMA, SUBSTR(TABNAME,1,15) AS TABNAME,
       LOCK_OBJECT_TYPE, LOCK_MODE, LOCK_MODE_REQUESTED, AGENT_ID_HOLDING_LK
FROM SYSIBMADM.LOCKWAITS WHERE AGENT_ID = 89
```

El siguiente ejemplo muestra los datos de salida de la consulta.

```
TABSCHEMA TABNAME      LOCK_OBJECT_TYPE LOCK_MODE ...
-----
JESSICAE T1            ROW_LOCK          X          ...
```

1 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```
... LOCK_MODE_REQUESTED AGENT_ID_HOLDING_LK
... -----
... NS                      7
```

Ejemplo 2: Listar el número total de peticiones de bloqueo pendientes por tabla en la base de datos SAMPLE. Si se ordena la salida por número de peticiones, es posible identificar las tablas con la contención más alta.

```
SELECT SUBSTR(TABSCHEMA,1,8) AS TABSCHEMA, SUBSTR(TABNAME, 1, 15)
AS TABNAME, COUNT(*) AS NUM_OF_LOCK_REQUESTS_WAITING,
DBPARTITIONNUM
FROM SYSIBMADM.LOCKWAITS WHERE DB_NAME = 'SAMPLE'
GROUP BY TABSCHEMA, TABNAME, DBPARTITIONNUM
ORDER BY NUM_OF_LOCK_REQUESTS_WAITING DESC
```

El siguiente ejemplo muestra los datos de salida de la consulta.

```
TABSCHEMA TABNAME      NUM_OF_LOCK_REQUESTS_WAITING DBPARTITIONNUM
-----
JESSICAE T3            2                      0
JESSICAE T1            1                      0
JESSICAE T2            1                      0
```

3 registro(s) seleccionado(s).

Información devuelta

Tabla 131. Información devuelta por la vista administrativa LOCKWAITS

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | Fecha y hora en que se generó el informe. |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| AGENT_ID | BIGINT | agent_id - Descriptor de contexto de aplicación (ID de agente) |
| APPL_NAME | VARCHAR(256) | appl_name - Nombre de aplicación |
| AUTHID | VARCHAR(128) | auth_id - ID de autorización |
| TBSP_NAME | VARCHAR(128) | tablespace_name - Nombre de espacio de tablas |
| TABSCHEMA | VARCHAR(128) | table_schema - Nombre de esquema de tabla |
| TABNAME | VARCHAR(128) | table_name - Nombre de tabla |
| SUBSECTION_NUMBER | BIGINT | ss_number - Número de subsección |

Tabla 131. Información devuelta por la vista administrativa LOCKWAITS (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|----------------------|---------------|--|
| LOCK_OBJECT_TYPE | VARCHAR(18) | lock_object_type - Tipo de objeto de bloqueo esperado. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • AUTORESIZE_LOCK • AUTOSTORAGE_LOCK • BLOCK_LOCK • EOT_LOCK • INPLACE_REORG_LOCK • INTERNAL_LOCK • INTERNALB_LOCK • INTERNALC_LOCK • INTERNALJ_LOCK • INTERNALL_LOCK • INTERNALO_LOCK • INTERNALQ_LOCK • INTERNALP_LOCK • INTERNALS_LOCK • INTERNALT_LOCK • INTERNALV_LOCK • KEYVALUE_LOCK • ROW_LOCK • SYSBOOT_LOCK • TABLE_LOCK • TABLE_PART_LOCK • TABLESPACE_LOCK • XML_PATH_LOCK |
| LOCK_WAIT_START_TIME | TIMESTAMP | lock_wait_start_time - Indicación de fecha y hora de inicio de espera de bloqueo |
| LOCK_NAME | VARCHAR(32) | lock_name - Nombre de bloqueo |

Tabla 131. Información devuelta por la vista administrativa LOCKWAITS (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------|---------------|---|
| LOCK_MODE | VARCHAR(10) | lock_mode - Modalidad de bloqueo. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • IN • IS • IX • NON (si no existe bloqueo) • NS • NW • S • SIX • U • X • Z |
| LOCK_MODE_REQUESTED | VARCHAR(10) | lock_mode_requested - Modalidad de bloqueo solicitada. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • IN • IS • IX • NON (si no existe bloqueo) • NS • NW • S • SIX • U • X • Z |
| AGENT_ID_HOLDING_LK | BIGINT | agent_id_holding_lock - ID de agente que mantiene el bloqueo |
| APPL_ID_HOLDING_LK | VARCHAR(128) | appl_id_holding_lk - ID de aplicación que mantiene el bloqueo |
| LOCK_ESCALATION | SMALLINT | lock_escalation - Escalamiento de bloqueos |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa LOG_UTILIZATION – Recuperar información de utilización de las anotaciones cronológicas

La vista administrativa LOG_UTILIZATION devuelve información sobre la utilización de las anotaciones cronológicas para la base de datos conectada actualmente. Se devuelve una sola fila para cada partición de base de datos.

El esquema es SYSIBMADM.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa LOG_UTILIZATION
- Privilegio CONTROL en la vista administrativa LOG_UTILIZATION
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCtrl
- SYSMAINT
- SYSADM

Ejemplo

Listar la utilización de las anotaciones cronológicas para la base de datos conectada actualmente.

```
SELECT * FROM SYSIBMADM.LOG_UTILIZATION
```

El siguiente ejemplo muestra los datos de salida de la consulta.

```
DB_NAME  ... LOG_UTILIZATION_PERCENT TOTAL_LOG_USED_KB  ...
-----  ... -----
SAMPLE  ...                9.75                1989 ...
                                     ...
1 registro(s) seleccionado(s).                ...
```

Datos de salida de esta consulta (continuación).

```
... TOTAL_LOG_AVAILABLE_KB TOTAL_LOG_USED_TOP_KB DBPARTITIONNUM
... -----
...                18411                1990                0
...
...
...
```

Nota sobre uso

En el caso de las bases de datos configuradas para la anotación cronológica infinita, LOG_UTILIZATION_PERCENT y TOTAL_LOG_AVAILABLE_KB serán NULL.

Información devuelta

Tabla 132. Información devuelta por la vista administrativa LOG_UTILIZATION

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|--|
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| LOG_UTILIZATION_PERCENT | DECIMAL(5,2) | Porcentaje de utilización del espacio total de anotaciones cronológicas. |
| TOTAL_LOG_USED_KB | BIGINT | total_log_used - Espacio de registro total utilizado. Esta interfaz devuelve un valor en KB. |
| TOTAL_LOG_AVAILABLE_KB | BIGINT | total_log_available - Registro total disponible. Esta interfaz devuelve un valor en KB. |
| TOTAL_LOG_USED_TOP_KB | BIGINT | tot_log_used_top - Espacio máximo de anotación cronológica total utilizado. Esta interfaz devuelve un valor en KB. |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa LONG_RUNNING_SQL

La vista administrativa LONG_RUNNING_SQL devuelve las sentencias SQL de ejecución más larga de la base de datos conectada actualmente.

El esquema es SYSIBMADM.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa LONG_RUNNING_SQL
- Privilegio CONTROL en la vista administrativa LONG_RUNNING_SQL
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCtrl
- SYSMaint
- SYSADM

Ejemplo

Recuperar un informe de las sentencias SQL de ejecución más larga de la base de datos conectada actualmente.

```
SELECT SUBSTR(STMT_TEXT, 1, 50) AS STMT_TEXT, AGENT_ID,  
ELAPSED_TIME_MIN, APPL_STATUS, DBPARTITIONNUM  
FROM SYSIBMADM.LONG_RUNNING_SQL ORDER BY DBPARTITIONNUM
```

El siguiente ejemplo muestra los datos de salida de la consulta.

```

STMT_TEXT                AGENT_ID    ...
-----
select * from dbuser.employee      228 ...
select * from dbuser.employee      228 ...
select * from dbuser.employee      228 ...
...
3 registro(s) seleccionado(s).

```

Datos de salida de esta consulta (continuación).

```

... ELAPSED_TIME_MIN APPL_STATUS    DBPARTITIONNUM
... -----
...                2 UOWWAIT                0
...                0 CONNECTED                1
...                0 CONNECTED                2

```

Nota sobre uso

Esta vista se puede utilizar para identificar las sentencias SQL de larga ejecución en la base de datos. Puede comprobar las consultas que están en marcha para ver qué sentencias son las de ejecución más larga y el estado actual de la consulta. Se puede investigar más acerca de la aplicación que contiene la sentencia SQL; para ello, se utiliza el ID de agente como el identificador exclusivo. Si la ejecución dura mucho tiempo y está a la espera de un bloqueo, quizá sea conveniente indagar más con las vistas administrativas LOCKWAITS o LOCKS_HELD. Si está a la espera del usuario, significa que el servidor DB2 no está haciendo nada, sino que está esperando a que la aplicación lleve a cabo alguna acción (como emitir la siguiente captación o enviar la siguiente sentencia de SQL).

Información devuelta

Tabla 133. Información devuelta por la vista administrativa LONG_RUNNING_SQL

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | Hora en que se generó el informe. |
| ELAPSED_TIME_MIN | INTEGER | Tiempo transcurrido de la sentencia en minutos. |
| AGENT_ID | BIGINT | agent_id - Descriptor de contexto de aplicación (ID de agente) |
| APPL_NAME | VARCHAR(256) | appl_name - Nombre de aplicación |

Tabla 133. Información devuelta por la vista administrativa
LONG_RUNNING_SQL (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|----------------------|---------------|--|
| APPL_STATUS | VARCHAR(22) | <p>appl_status - Estado de aplicación. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h, y es uno de los siguientes:</p> <ul style="list-style-type: none"> • BACKUP • COMMIT_ACT • COMP • CONNECTED • CONNECTPEND • CREATE_DB • DECOUPLED • DISCONNECTPEND • INTR • IOERROR_WAIT • LOAD • LOCKWAIT • QUIESCE_TABLESPACE • RECOMP • REMOTE_RQST • RESTART • RESTORE • ROLLBACK_ACT • ROLLBACK_TO_SAVEPOINT • TEND • THABRT • THCOMT • TPREP • UNLOAD • UOWEXEC • UOWWAIT • WAITFOR_REMOTE |
| AUTHID | VARCHAR(128) | auth_id - ID de autorización |
| INBOUND_COMM_ADDRESS | VARCHAR(32) | inbound_comm_address - Dirección de comunicaciones de entrada |
| STMT_TEXT | CLOB(16 M) | stmt_text - Texto de sentencia de SQL |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa QUERY_PREP_COST – Recuperar información de tiempo de preparación de sentencias

La vista administrativa QUERY_PREP_COST devuelve una lista de las sentencias con información acerca del tiempo necesario para prepararlas.

El esquema es SYSIBMADM.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa QUERY_PREP_COST
- Privilegio CONTROL en la vista administrativa SNAPAGENT
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE para la función de tabla SNAPDYN_SQL
- Autorización DATAACCESS

Ejemplo

Recuperar un informe sobre las consultas con el porcentaje más alto de tiempo dedicado a la preparación.

```
SELECT NUM_EXECUTIONS, AVERAGE_EXECUTION_TIME_S, PREP_TIME_PERCENT,  
       SUBSTR(STMT_TEXT, 1, 30) AS STMT_TEXT, DBPARTITIONNUM  
FROM SYSIBMADM.QUERY_PREP_COST ORDER BY PREP_TIME_PERCENT
```

El siguiente ejemplo muestra los datos de salida de la consulta.

```
NUM_EXECUTIONS      AVERAGE_EXECUTION_TIME_S ...  
-----...-----  
1                      25 ...
```

1 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```
... PREP_TIME_PERCENT STMT_TEXT                                DBPARTITIONNUM  
... -----  
...                0.0 select * from dbuser.employee          0
```

Notas de uso

Al realizar una selección en la vista, se puede utilizar una cláusula ORDER BY para identificar las consultas con el coste de preparación más alto. Puede examinar esta vista para ver la frecuencia con la que se ejecuta una consulta, así como el tiempo medio de ejecución de cada una de estas consultas. Si el tiempo que se tarda en compilar y optimizar una consulta es casi tanto como el que se necesita para que la consulta se ejecute, quizás le interese ver la clase de optimización que está utilizando. La reducción de la clase de optimización puede hacer que la optimización completa de la consulta sea más rápida y, por lo tanto, devuelva un resultado en menos tiempo. Sin embargo, si una consulta tarda mucho tiempo en prepararse pero se ejecuta miles de veces (sin tener que prepararla de nuevo), es posible que la clase de optimización no sea un problema.

Información devuelta

Tabla 134. Información devuelta por la vista administrativa QUERY_PREP_COST

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | Fecha y hora en que se generó el informe. |
| NUM_EXECUTIONS | BIGINT | num_executions - Ejecuciones de sentencia |
| AVERAGE_EXECUTION_TIME_S | BIGINT | Tiempo de ejecución medio en segundos. |
| PREP_TIME_MS | BIGINT | prep_time_worst - Peor tiempo de preparación de sentencia |
| PREP_TIME_PERCENT | DECIMAL(5,2) | Porcentaje de tiempo de ejecución dedicado a la preparación. |
| STMT_TEXT | CLOB(2 M) | stmt_text - Texto de sentencia de SQL |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa SNAPAGENT y función de tabla SNAP_GET_AGENT – Recuperar información de instantáneas de la aplicación del grupo de datos lógicos agent

La vista administrativa SNAPAGENT y la función de tabla SNAP_GET_AGENT devuelven información sobre agentes de una instantánea de la aplicación, en concreto el grupo de datos lógicos agent.

Vista administrativa SNAPAGENT

Esta vista administrativa permite recuperar la información de instantáneas de la aplicación del grupo de datos lógicos agent para la base de datos conectada actualmente.

La vista administrativa SNAPAGENT, utilizada con las vistas administrativas SNAPAGENT_MEMORY_POOL, SNAPAPPL, SNAPAPPL_INFO, SNAPSTMT y SNAPSUBSECTION, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR APPLICATIONS ON alias de base de datos, pero recupera datos de todas las particiones de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 135 en la página 518 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPAGENT
- Privilegio CONTROL en la vista administrativa SNAPAGENT

- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE en la función de tabla SNAP_GET_AGENT
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar toda la información de instantáneas de la aplicación para la base de datos conectada actualmente desde el grupo de datos lógicos agent.

```
SELECT * FROM SYSIBMADM.SNAPAGENT
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| SNAPSHOT_TIMESTAMP | DB_NAME | AGENT_ID | ... |
|----------------------------|---------|----------|-----|
| 2005-07-19-11.03.26.740423 | SAMPLE | 101 | ... |
| 2005-07-19-11.03.26.740423 | SAMPLE | 49 | ... |

2 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

| AGENT_PID | LOCK_TIMEOUT_VAL | DBPARTITIONNUM |
|-----------|------------------|----------------|
| 11980 | -1 | 0 |
| 15940 | -1 | 0 |

SNAP_GET_AGENT, función de tabla

La función de tabla SNAP_GET_AGENT devuelve la misma información que la vista administrativa SNAPAGENT, pero permite recuperar la información para una base de datos concreta en una partición específica de la base de datos, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_AGENT, utilizada con las funciones de tabla SNAP_GET_AGENT_MEMORY_POOL, SNAP_GET_APPL_V95, SNAP_GET_APPL_INFO_V95, SNAP_GET_STMT y SNAP_GET_SUBSECTION, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR ALL APPLICATIONS, pero recupera datos de todas las particiones de base de datos.

Consulte la Tabla 135 en la página 518 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
▶▶ SNAP_GET_AGENT ( ( nombre-bd [ , número-partición-bd ] ) ) ▶▶
```


El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique una serie vacía para tomar la instantánea desde la base de datos conectada actualmente. Especifique un valor NULL para tomar la instantánea desde todas las bases de datos que se encuentran en la misma instancia que la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_AGENT toma una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE en la función de tabla SNAP_GET_AGENT
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMAINT
- SYSADM

Ejemplo

Recuperar toda la información de instantáneas de la aplicación para todas las aplicaciones en todas las bases de datos activas.

```
SELECT * FROM TABLE(SNAP_GET_AGENT(CAST(NULL AS VARCHAR(128)), -1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```

SNAPSHOT_TIMESTAMP      DB_NAME      AGENT_ID      ...
-----
2006-01-03-17.21.38.530785 SAMPLE        48 ...
2006-01-03-17.21.38.530785 SAMPLE        47 ...
2006-01-03-17.21.38.530785 SAMPLE        46 ...
2006-01-03-17.21.38.530785 TESTDB        30 ...
2006-01-03-17.21.38.530785 TESTDB        29 ...
2006-01-03-17.21.38.530785 TESTDB        28 ...

```

6 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```

... AGENT_PID      LOCK_TIMEOUT_VAL      DBPARTITIONNUM
... -----
...          7696          -1          0
...          8536          -1          0
...          6672          -1          0
...          2332          -1          0
...          8360          -1          0
...          6736          -1          0
...

```

Información devuelta

Tabla 135. Información devuelta por la vista administrativa SNAPAGENT y la función de tabla SNAP_GET_AGENT

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| AGENT_ID | BIGINT | agent_id - Descriptor de contexto de aplicación (ID de agente) |
| AGENT_PID | BIGINT | agent_pid - Unidad susceptible de envío de motor (EDU) |
| LOCK_TIMEOUT_VAL | BIGINT | lock_timeout_val - Tiempo de espera excedido de bloqueo (segundos) |
| DBPARTITIONNUM | SMALLINT | La partición de base de datos desde la que se han recuperado los datos para la fila. |

Vista administrativa SNAPAGENT_MEMORY_POOL y función de tabla SNAP_GET_AGENT_MEMORY_POOL – Recuperar información de instantáneas del grupo de datos lógicos memory_pool

La vista administrativa SNAPAGENT_MEMORY_POOL y la tabla de función SNAP_GET_AGENT_MEMORY_POOL devuelven información acerca del uso de la memoria en el nivel de agente.

Vista administrativa SNAPAGENT_MEMORY_POOL

Esta vista administrativa permite recuperar la información de instantáneas del grupo de datos lógicos memory_pool sobre el uso de la memoria en el nivel del agente para la base de datos conectada actualmente.

La vista administrativa SNAPAGENT_MEMORY_POOL, utilizada con las vistas administrativas SNAPAGENT, SNAPAPPL, SNAPAPPL_INFO, SNAPSTMT y SNAPSUBSECTION, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR APPLICATIONS ON alias de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 136 en la página 521 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPAGENT_MEMORY_POOL
- Privilegio CONTROL en la vista administrativa SNAPAGENT_MEMORY_POOL
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_AGENT_MEMORY_POOL
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCtrl
- SYSMAINT
- SYSADM

Ejemplo

Recuperar una lista de agrupaciones de memoria y su tamaño actual.

```
SELECT AGENT_ID, POOL_ID, POOL_CUR_SIZE FROM SYSIBMADM.SNAPAGENT_MEMORY_POOL
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
AGENT_ID      POOL_ID POOL_  CUR_SIZE
-----
.....
          48 APPLICATION          65536
          48 OTHER              65536
          48 APPL_CONTROL        65536
          47 APPLICATION          65536
          47 OTHER              131072
          47 APPL_CONTROL        65536
          46 OTHER              327680
          46 APPLICATION          262144
          46 APPL_CONTROL        65536
```

9 registro(s) seleccionado(s).

SNAP_GET_AGENT_MEMORY_POOL, función de tabla

La función de tabla SNAP_GET_AGENT_MEMORY_POOL devuelve la misma información que la vista administrativa SNAPAGENT_MEMORY_POOL, pero permite recuperar la información para una base de datos concreta en una partición específica de la base de datos, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_AGENT_MEMORY_POOL, utilizada con las funciones de tabla SNAP_GET_AGENT, SNAP_GET_APPL_V95, SNAP_GET_APPL_INFO_V95, SNAP_GET_STMT y SNAP_GET_SUBSECTION, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR ALL APPLICATIONS.

Consulte la Tabla 136 en la página 521 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
▶▶ SNAP_GET_AGENT_MEMORY_POOL ( ( nombrebd ) , numparticiónbd ) ▶▶
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique una serie vacía para tomar la instantánea desde la base de datos conectada actualmente. Especifique un valor NULL para tomar la instantánea desde todas las bases de datos que se encuentran en la misma instancia que la base de datos conectada actualmente.

numparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *numparticiónbd* se establece en NULL, -1 se establece implícitamente para *numparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *numparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_AGENT_MEMORY_POOL toma una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_AGENT_MEMORY_POOL
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar una lista de agrupaciones de memoria y su tamaño actual para todas las bases de datos.

```
SELECT SUBSTR(DB_NAME,1,8) AS DB_NAME, AGENT_ID, POOL_ID, POOL_CUR_SIZE
FROM TABLE(SNAP_GET_AGENT_MEMORY_POOL(CAST (NULL AS VARCHAR(128)), -1))
AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| DB_NAME | AGENT_ID | POOL_ID | POOL_CUR_SIZE |
|---------|----------|--------------|---------------|
| SAMPLE | 48 | APPLICATION | 65536 |
| SAMPLE | 48 | OTHER | 65536 |
| SAMPLE | 48 | APPL_CONTROL | 65536 |
| SAMPLE | 47 | APPLICATION | 65536 |
| SAMPLE | 47 | OTHER | 131072 |
| SAMPLE | 47 | APPL_CONTROL | 65536 |
| SAMPLE | 46 | OTHER | 327680 |
| SAMPLE | 46 | APPLICATION | 262144 |
| SAMPLE | 46 | APPL_CONTROL | 65536 |
| TESTDB | 30 | APPLICATION | 65536 |
| TESTDB | 30 | OTHER | 65536 |
| TESTDB | 30 | APPL_CONTROL | 65536 |
| TESTDB | 29 | APPLICATION | 65536 |
| TESTDB | 29 | OTHER | 131072 |
| TESTDB | 29 | APPL_CONTROL | 65536 |
| TESTDB | 28 | OTHER | 327680 |
| TESTDB | 28 | APPLICATION | 65536 |
| TESTDB | 28 | APPL_CONTROL | 65536 |

18 registro(s) seleccionado(s).

Información devuelta

Tabla 136. Información devuelta por la vista administrativa SNAPAGENT_MEMORY_POOL y la función de tabla SNAP_GET_AGENT_MEMORY_POOL

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |

Tabla 136. Información devuelta por la vista administrativa SNAPAGENT_MEMORY_POOL y la función de tabla SNAP_GET_AGENT_MEMORY_POOL (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|--|
| AGENT_ID | BIGINT | agent_id - Descriptor de contexto de aplicación (ID de agente) |
| AGENT_PID | BIGINT | agent_pid - Unidad susceptible de envío de motor (EDU) |
| POOL_ID | VARCHAR(14) | pool_id - Identificador de agrupación de memoria. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • APP_GROUP • APPL_CONTROL • APPLICATION • BP • CAT_CACHE • DATABASE • DFM • FCMBP • IMPORT_POOL • LOCK_MGR • MONITOR • OTHER • PACKAGE_CACHE • QUERY • SHARED_SORT • SORT • STATEMENT • STATISTICS • UTILITY |
| POOL_CUR_SIZE | BIGINT | pool_cur_size - Tamaño actual de la agrupación de memoria |
| POOL_WATERMARK | BIGINT | pool_watermark - Marca límite de agrupación de memoria |
| POOL_CONFIG_SIZE | BIGINT | pool_config_size - Tamaño configurado de la agrupación de memoria |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa SNAPAPPL_INFO y función de tabla SNAP_GET_APPL_INFO_V95 – Recuperar información de instantáneas del grupo de datos lógicos appl_info

La vista administrativa SNAPAPPL_INFO y la función de tabla SNAP_GET_APPL_INFO_V95 devuelven información acerca de las aplicaciones a partir de una instantánea de la aplicación, en concreto del grupo de datos lógico appl_info.

SNAPAPPL_INFO, vista administrativa

Esta vista administrativa permite recuperar la información de instantáneas del grupo de datos lógicos appl_info para la base de datos conectada actualmente.

La vista administrativa SNAPAPPL_INFO, utilizada con las vistas administrativas SNAPAGENT, SNAPAGENT_MEMORY_POOL, SNAPAPPL, SNAPSTMT y SNAPSUBSECTION, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR APPLICATIONS ON alias de base de datos, pero recupera datos de todas las particiones de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 137 en la página 526 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPAPPL_INFO
- Privilegio CONTROL en la vista administrativa SNAPAPPL_INFO
- Autorización DATAACCESS

Además, se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE en la función de tabla SNAP_GET_APPL_INFO_V95
- Autorización DATAACCESS

Además, se necesita una de las autorizaciones siguientes:

- SYSMON
- SYSMAINT
- SYSCTRL
- SYSADM

Ejemplo

Recuperar el estado de las aplicaciones conectadas a la base de datos actual.

```
SELECT AGENT_ID, SUBSTR(APPL_NAME,1,10) AS APPL_NAME, APPL_STATUS  
FROM SYSIBMADM.SNAPAPPL_INFO
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| AGENT_ID | APPL_NAME | APPL_STATUS |
|----------|-----------|-------------|
| 101 | db2bp.exe | UOWEXEC |
| 49 | db2bp.exe | CONNECTED |

2 registro(s) seleccionado(s).

SNAP_GET_APPL_INFO_V95, función de tabla

La función de tabla SNAP_GET_APPL_INFO_V95 devuelve la misma información que la vista administrativa SNAPAPPL_INFO, pero permite recuperar la información para una base de datos concreta en una partición específica de la base de datos, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_APPL_INFO_V95, utilizada con las funciones de tabla SNAP_GET_AGENT, SNAP_GET_AGENT_MEMORY_POOL, SNAP_GET_APPL_V95, SNAP_GET_STMT y SNAP_GET_SUBSECTION, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR ALL APPLICATIONS, pero recupera datos de todas las particiones de base de datos.

Consulte la Tabla 137 en la página 526 para ver una lista completa de la información que se puede devolver.

Sintaxis

```

→ SNAP_GET_APPL_INFO_V95 ( ( nombrebd [ , numparticiónbd ] ) )

```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique una serie vacía para tomar la instantánea desde la base de datos conectada actualmente. Especifique un valor NULL para tomar la instantánea desde todas las bases de datos que se encuentran en la misma instancia que la base de datos conectada actualmente.

numparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *numparticiónbd* se establece en NULL, -1 se establece implícitamente para *numparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *numparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es

posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_APPL_INFO_V95 tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE en la función de tabla SNAP_GET_APPL_INFO_V95
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplos

Recuperar el estado de todas las aplicaciones en la partición de base de datos conectada.

```
SELECT SUBSTR(DB_NAME,1,8) AS DB_NAME, AGENT_ID,
       SUBSTR(APPL_NAME,1,10) AS APPL_NAME, APPL_STATUS
FROM TABLE(SNAP_GET_APPL_INFO_V95(CAST(NULL AS VARCHAR(128)),-1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| DB_NAME | AGENT_ID | APPL_NAME | APPL_STATUS |
|---------|----------|-----------|-------------|
| TOOLSDB | 14 | db2bp.exe | CONNECTED |
| SAMPLE | 15 | db2bp.exe | UOWEXEC |
| SAMPLE | 8 | javaw.exe | CONNECTED |
| SAMPLE | 7 | db2bp.exe | UOWWAIT |

4 registro(s) seleccionado(s).

A continuación se muestra lo que se obtiene al ejecutar SELECT sobre el resultado de la función de tabla.

```
SELECT SUBSTR(DB_NAME,1,8) AS DB_NAME, AUTHORITY_LVL
FROM TABLE(SNAP_GET_APPL_INFO_V95(CAST(NULL AS VARCHAR(128)),-1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| DB_NAME | AUTHORITY_LVL |
|---------|---|
| TESTDB | SYSADM(GROUP) + DBADM(USER) + CREATETAB(USER, GROUP) + BINDADD(USER, GROUP) + CONNECT(USER, GROUP) + CREATE_NOT_FENC(USER) + IMPLICIT_SCHEMA(USER, GROUP) + LOAD(USER) + CREATE_EXT_RT(USER) + QUIESCE_CONN(USER) |
| TESTDB | SYSADM(GROUP) + DBADM(USER) + CREATETAB(USER, GROUP) + BINDADD(USER, GROUP) + CONNECT(USER, GROUP) + CREATE_NOT_FENC(USER) + IMPLICIT_SCHEMA(USER, GROUP) + LOAD(USER) + CREATE_EXT_RT(USER) + QUIESCE_CONN(USER) |
| TESTDB | SYSADM(GROUP) + DBADM(USER) + CREATETAB(USER, GROUP) + BINDADD(USER, GROUP) + CONNECT(USER, GROUP) + CREATE_NOT_FENC(USER) + IMPLICIT_SCHEMA(USER, GROUP) + LOAD(USER) + CREATE_EXT_RT(USER) + QUIESCE_CONN(USER) |

3 registro(s) seleccionado(s).

Información devuelta

Tabla 137. Información devuelta por la vista administrativa SNAPAPPL_INFO y la función de tabla SNAP_GET_APPL_INFO_V95

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| AGENT_ID | BIGINT | agent_id - Descriptor de contexto de aplicación (ID de agente) |
| APPL_STATUS | VARCHAR(22) | appl_status - Estado de aplicación. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • BACKUP • COMMIT_ACT • COMP • CONNECTED • CONNECTPEND • CREATE_DB • DECOUPLED • DISCONNECTPEND • INTR • IOERROR_WAIT • LOAD • LOCKWAIT • QUIESCE_TABLESPACE • RECOMP • REMOTE_RQST • RESTART • RESTORE • ROLLBACK_ACT • ROLLBACK_TO_SAVEPOINT • TEND • THABRT • THCOMT • TPREP • UNLOAD • UOWEXEC • UOWWAIT • WAITFOR_REMOTE |
| CODEPAGE_ID | BIGINT | codepage_id - ID de página de códigos utilizada por aplicación |
| NUM_ASSOC_AGENTS | BIGINT | num_assoc_agents - Número de agentes asociados |
| COORD_NODE_NUM | SMALLINT | coord_node - Nodo de coordinación |

Tabla 137. Información devuelta por la vista administrativa SNAPAPPL_INFO y la función de tabla SNAP_GET_APPL_INFO_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|---|
| AUTHORITY_LVL | VARCHAR(512) | <p>authority_bitmap - Nivel de autorización del usuario.</p> <p>Esta interfaz devuelve un identificador de texto basado en las autorizaciones de base de datos definidas en sql.h y su fuente y tiene el siguiente formato: autorización(fuente, ...) + autorización(fuente , ...) + ... La fuente de una autorización puede ser múltiple: de USER, de GROUP, o de USER y GROUP.</p> <p>Los posibles valores para "autorización" son:</p> <ul style="list-style-type: none"> • ACCESSCTRL • BINDADD • CONNECT • CREATE_EXT_RT • CREATE_NOT_FENC • CREATETAB • DATAACCESS • DBADM • EXPLAIN • IMPLICIT_SCHEMA • LOAD • LIBADM • QUIESCE_CONN • SECADM • SQLADM • SYSADM • SYSCTRL • SYSMANT • SYSMON • SYSQUIESCE • WLMADM <p>Los posibles valores para "fuente" son:</p> <ul style="list-style-type: none"> • USER – autorización que se otorga al usuario o a un rol otorgado al usuario. • GROUP – autorización que se otorga a un grupo al que pertenece el usuario o a un rol que se ha otorgado al grupo al que pertenece el usuario. |

Tabla 137. Información devuelta por la vista administrativa SNAPAPPL_INFO y la función de tabla SNAP_GET_APPL_INFO_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| CLIENT_PID | BIGINT | client_pid - ID de proceso de cliente |
| COORD_AGENT_PID | BIGINT | coord_agent_pid - Agente coordinador |
| STATUS_CHANGE_TIME | TIMESTAMP | status_change_time - Hora de cambio de estado de aplicación |
| CLIENT_PLATFORM | VARCHAR(12) | <p>client_platform - Plataforma operativa de cliente. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h,</p> <ul style="list-style-type: none"> • AIX • AIX64 • AS400_DRDA • DOS • DYNIX • HP • HP64 • HPIA • HPIA64 • LINUX • LINUX390 • LINUXIA64 • LINUXPPC • LINUXPPC64 • LINUXX8664 • LINUXZ64 • MAC • MVS_DRDA • NT • NT64 • OS2 • OS390 • SCO • SGI • SNI • SUN • SUN64 • UNKNOWN • UNKNOWN_DRDA • VM_DRDA • VSE_DRDA • WINDOWS |

Tabla 137. Información devuelta por la vista administrativa SNAPAPPL_INFO y la función de tabla SNAP_GET_APPL_INFO_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------|---------------|---|
| CLIENT_PROTOCOL | VARCHAR(10) | client_protocol - Protocolo de comunicaciones de cliente. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h, <ul style="list-style-type: none"> • CPIC • LOCAL • NETBIOS • NPIPE • TCPIP (para DB2 UDB) • TCPIP4 • TCPIP6 |
| TERRITORY_CODE | SMALLINT | territory_code - Código de territorio de base de datos |
| APPL_NAME | VARCHAR(256) | appl_name - Nombre de aplicación |
| APPL_ID | VARCHAR(128) | appl_id - ID de aplicación |
| SEQUENCE_NO | VARCHAR(4) | sequence_no - Número de secuencia |
| PRIMARY_AUTH_ID | VARCHAR(128) | auth_id - ID de autorización |
| SESSION_AUTH_ID | VARCHAR(128) | session_auth_id - ID de autorización de sesión |
| CLIENT_NNAME | VARCHAR(128) | El elemento del supervisor client_nname ha quedado obsoleto. El valor devuelto no es un valor válido. |
| CLIENT_PRDID | VARCHAR(128) | client_prdid - ID de producto/versión de cliente |
| INPUT_DB_ALIAS | VARCHAR(128) | input_db_alias - Alias de base de datos de entrada |
| CLIENT_DB_ALIAS | VARCHAR(128) | client_db_alias - Alias de base de datos utilizado por aplicación |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| DB_PATH | VARCHAR(1024) | db_path - Vía de acceso de base de datos |
| EXECUTION_ID | VARCHAR(128) | execution_id - ID de inicio de sesión de usuario |
| CORR_TOKEN | VARCHAR(128) | corr_token - Símbolo de correlación DRDA |
| TPMON_CLIENT_USERID | VARCHAR(256) | tpmon_client_userid - ID de usuario de cliente de supervisor de TP |
| TPMON_CLIENT_WKSTN | VARCHAR(256) | tpmon_client_wkstn - Nombre de estación de trabajo de cliente de supervisor de TP |

Tabla 137. Información devuelta por la vista administrativa SNAPAPPL_INFO y la función de tabla SNAP_GET_APPL_INFO_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|---|
| TPMON_CLIENT_APP | VARCHAR(256) | tpmon_client_app - Nombre de aplicación de cliente de supervisor de TP |
| TPMON_ACC_STR | VARCHAR(200) | tpmon_acc_str - Serie de contabilidad de cliente de supervisor de TP |
| DBPARTITIONNUM | SMALLINT | La partición de base de datos desde la que se han recuperado los datos para la fila. |
| WORKLOAD_ID | INTEGER | ID de carga de trabajo actual. |
| IS_SYSTEM_APPL | SMALLINT | El valor de IS_SYSTEM_APPL indica si la aplicación es o no una aplicación del sistema interno DB2 0 significa que se una aplicación de usuario 1 significa que es una aplicación del sistema. Un ejemplo de una aplicación del sistema DB2 es un supervisor de sucesos de DB2. En general, los nombres de aplicaciones del sistema DB2 comienzan por "db2". Por ejemplo: db2stmm, db2taskd. |

Vista administrativa SNAPAPPL y función de tabla SNAP_GET_APPL_V95 – Recuperar información de instantáneas del grupo de datos lógicos appl

La "SNAPAPPL, vista administrativa" y la "SNAP_GET_APPL_V95, función de tabla" en la página 531 devuelven información sobre aplicaciones desde una instantánea de aplicaciones, en concreto, el grupo de datos lógicos appl.

SNAPAPPL, vista administrativa

Esta vista administrativa permite recuperar la información de instantáneas del grupo de datos lógicos appl para la base de datos conectada actualmente.

La vista administrativa SNAPAPPL, utilizada con las vistas administrativas SNAPAGENT, SNAPAGENT_MEMORY_POOL, SNAPAPPL_INFO, SNAPSTMT y SNAPSUBSECTION, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR APPLICATIONS ON alias de base de datos, pero recupera datos de todas las particiones de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 138 en la página 533 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPAPPL
- Privilegio CONTROL en la vista administrativa SNAPAPPL
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_APPL_V95
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMAINT
- SYSADM

Ejemplo

Recuperar detalles en filas leídas y grabadas para cada aplicación en la base de datos conectada.

```
SELECT SUBSTR(DB_NAME,1,8) AS DB_NAME, AGENT_ID, ROWS_READ, ROWS_WRITTEN
      FROM SYSIBMADM.SNAPAPPL
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| DB_NAME | AGENT_ID | ROWS_READ | ROWS_WRITTEN |
|---------|----------|-----------|--------------|
| SAMPLE | | 7 | 25 |

1 registro(s) seleccionado(s).

SNAP_GET_APPL_V95, función de tabla

La función de tabla SNAP_GET_APPL_V95 devuelve la misma información que la vista administrativa SNAPAPPL, pero permite recuperar la información para una base de datos concreta en una partición específica de la base de datos, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_APPL_V95, utilizada con las funciones de tabla SNAP_GET_AGENT, SNAP_GET_AGENT_MEMORY_POOL, SNAP_GET_APPL_INFO_V95, SNAP_GET_STMT y SNAP_GET_SUBSECTION, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR ALL APPLICATIONS, pero recupera datos de todas las particiones de base de datos.

Consulte la Tabla 138 en la página 533 para ver una lista completa de la información que se puede devolver.

Sintaxis

► SNAP_GET_APPL_V95 (—*nombrebd* [—*númparticiónbd*—]) ►

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique una serie vacía para tomar la instantánea desde la base de datos conectada actualmente. Especifique un valor NULL para tomar la instantánea desde todas las bases de datos que se encuentran en la misma instancia que la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_APPL_V95 tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_APPL_V95
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar detalles en filas leídas y grabadas para cada aplicación en todas las bases de datos activas.

```
SELECT SUBSTR(DB_NAME,1,8) AS DB_NAME, AGENT_ID, ROWS_READ, ROWS_WRITTEN
FROM TABLE (SNAP_GET_APPL_V95(CAST(NULL AS VARCHAR(128)),-1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| DB_NAME | AGENT_ID | ROWS_READ | ROWS_WRITTEN |
|---------|----------|-----------|--------------|
| WSDB | 679 | 0 | 0 |
| WSDB | 461 | 3 | 0 |
| WSDB | 460 | 4 | 0 |
| TEST | 680 | 4 | 0 |
| TEST | 455 | 6 | 0 |
| TEST | 454 | 0 | 0 |
| TEST | 453 | 50 | 0 |

Información devuelta

Tabla 138. Información devuelta por la vista administrativa SNAPAPPL y la función de tabla SNAP_GET_APPL_V95

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| AGENT_ID | BIGINT | agent_id - Descriptor de contexto de aplicación (ID de agente) |
| UOW_LOG_SPACE_USED | BIGINT | uow_log_space_used - Unidad de espacio de registro de trabajo utilizado |
| ROWS_READ | BIGINT | rows_read - Filas leídas |
| ROWS_WRITTEN | BIGINT | rows_written - Filas grabadas |
| INACT_STMTHIST_SZ | BIGINT | stmt_history_list_size - Tamaño de lista de historial de sentencias |
| POOL_DATA_L_READS | BIGINT | pool_data_l_reads - Lecturas lógicas de datos de la agrupación de almacenamientos intermedios |
| POOL_DATA_P_READS | BIGINT | pool_data_p_reads - Lecturas físicas de datos de la agrupación de almacenamientos intermedios |
| POOL_DATA_WRITES | BIGINT | pool_data_writes - Grabaciones de datos de la agrupación de almacenamientos intermedios |
| POOL_INDEX_L_READS | BIGINT | pool_index_l_reads - Lecturas lógicas de índices de la agrupación de almacenamientos intermedios |
| POOL_INDEX_P_READS | BIGINT | pool_index_p_reads - Lecturas físicas de índices de la agrupación de almacenamientos intermedios |

Tabla 138. Información devuelta por la vista administrativa SNAPAPPL y la función de tabla SNAP_GET_APPL_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|--|
| POOL_INDEX_WRITES | BIGINT | pool_index_writes - Grabaciones de índices de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_L_READS | BIGINT | pool_temp_data_l_reads - Lecturas lógicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_P_READS | BIGINT | pool_temp_data_p_reads - Lecturas físicas de datos temporales de agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_L_READS | BIGINT | pool_temp_index_l_reads - Lecturas lógicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_P_READS | BIGINT | pool_temp_index_p_reads - Lecturas físicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_L_READS | BIGINT | pool_temp_xda_l_reads - Lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_P_READS | BIGINT | pool_temp_xda_p_reads - Elemento del supervisor de las lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |
| POOL_XDA_L_READS | BIGINT | pool_xda_l_reads - Lecturas lógicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_XDA_P_READS | BIGINT | pool_xda_p_reads - Lecturas físicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_XDA_WRITES | BIGINT | pool_xda_writes - Grabaciones de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_READ_TIME | BIGINT | pool_read_time - Tiempo total de lectura física de la agrupación de almacenamientos intermedios |
| POOL_WRITE_TIME | BIGINT | pool_write_time - Tiempo total de grabación física de la agrupación de almacenamientos intermedios |
| DIRECT_READS | BIGINT | direct_reads - Lecturas directas de la base de datos |
| DIRECT_WRITES | BIGINT | direct_writes - Grabaciones directas en la base de datos |

Tabla 138. Información devuelta por la vista administrativa SNAPAPPL y la función de tabla SNAP_GET_APPL_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------|---------------|--|
| DIRECT_READ_REQS | BIGINT | direct_read_reqs - Peticiones de lectura directa |
| DIRECT_WRITE_REQS | BIGINT | direct_write_reqs - Peticiones de grabación directa |
| DIRECT_READ_TIME | BIGINT | direct_read_time - Tiempo de lectura directa |
| DIRECT_WRITE_TIME | BIGINT | direct_write_time - Tiempo de grabación directa |
| UNREAD_PREFETCH_PAGES | BIGINT | unread_prefetch_pages - Páginas de captación previa no leídas |
| LOCKS_HELD | BIGINT | locks_held - Bloqueos retenidos |
| LOCK_WAITS | BIGINT | lock_waits - Esperas de bloqueo |
| LOCK_WAIT_TIME | BIGINT | lock_wait_time - Tiempo esperado en bloqueos |
| LOCK_ESCALS | BIGINT | lock_escals - Número de escalamientos de bloqueo |
| X_LOCK_ESCALS | BIGINT | x_lock_escals - Escalas de bloqueo exclusivas |
| DEADLOCKS | BIGINT | deadlocks - Puntos muertos detectados |
| TOTAL_SORTS | BIGINT | total_sorts - Clasificaciones totales |
| TOTAL_SORT_TIME | BIGINT | total_sort_time - Tiempo de clasificación total |
| SORT_OVERFLOWS | BIGINT | sort_overflows - Desbordamientos de clasificación |
| COMMIT_SQL_STMTS | BIGINT | commit_sql_stmts - Sentencias de confirmación intentadas |
| ROLLBACK_SQL_STMTS | BIGINT | rollback_sql_stmts - Sentencias de retrotracción intentadas |
| DYNAMIC_SQL_STMTS | BIGINT | dynamic_sql_stmts - Sentencias de SQL dinámico intentadas |
| STATIC_SQL_STMTS | BIGINT | static_sql_stmts - Sentencias de SQL estático intentadas |
| FAILED_SQL_STMTS | BIGINT | failed_sql_stmts - Operaciones de sentencia anómalas |
| SELECT_SQL_STMTS | BIGINT | select_sql_stmts - Sentencias Select de SQL ejecutadas |
| DDL_SQL_STMTS | BIGINT | ddl_sql_stmts - Sentencias de lenguaje de definición de datos (DDL) de SQL |
| UID_SQL_STMTS | BIGINT | uid_sql_stmts - Sentencias UPDATE/INSERT/DELETE de SQL ejecutadas |
| INT_AUTO_REBINDS | BIGINT | int_auto_rebinds - Revinculaciones automáticas internas |

Tabla 138. Información devuelta por la vista administrativa SNAPAPPL y la función de tabla SNAP_GET_APPL_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------|---------------|--|
| INT_ROWS_DELETED | BIGINT | int_rows_deleted - Filas internas suprimidas |
| INT_ROWS_UPDATED | BIGINT | int_rows_updated - Filas internas actualizadas |
| INT_COMMITS | BIGINT | int_commits - Confirmaciones internas |
| INT_ROLLBACKS | BIGINT | int_rollback - Retrotracciones internas |
| INT_DEADLOCK_ROLLBACKS | BIGINT | int_deadlock_rollback - Retrotracciones internas debido a punto muerto |
| ROWS_DELETED | BIGINT | rows_deleted - Filas suprimidas |
| ROWS_INSERTED | BIGINT | rows_inserted - Filas insertadas |
| ROWS_UPDATED | BIGINT | rows_updated - Filas actualizadas |
| ROWS_SELECTED | BIGINT | rows_selected - Filas seleccionadas |
| BINDS_PRECOMPILES | BIGINT | binds_precompiles - Vinculaciones/precompilaciones intentadas |
| OPEN_REM_CURS | BIGINT | open_rem_curs - Cursores remotos abiertos |
| OPEN_REM_CURS_BLK | BIGINT | open_rem_curs_blk - Cursores remotos abiertos con bloqueo |
| REJ_CURS_BLK | BIGINT | rej_curs_blk - Peticiones de cursor de bloqueo rechazadas |
| ACC_CURS_BLK | BIGINT | acc_curs_blk - Peticiones de cursor de bloqueo aceptadas |
| SQL_REQS_SINCE_COMMIT | BIGINT | sql_reqs_since_commit - Peticiones SQL desde la última confirmación |
| LOCK_TIMEOUTS | BIGINT | lock_timeouts - Número de tiempos de espera de bloqueo |
| INT_ROWS_INSERTED | BIGINT | int_rows_inserted - Filas internas insertadas |
| OPEN_LOC_CURS | BIGINT | open_loc_curs - Cursores locales abiertos |
| OPEN_LOC_CURS_BLK | BIGINT | open_loc_curs_blk - Cursores locales abiertos con bloqueo |
| PKG_CACHE_LOOKUPS | BIGINT | pkg_cache_lookups - Búsquedas de antememoria de paquete |
| PKG_CACHE_INSERTS | BIGINT | pkg_cache_inserts - Inserciones de antememoria de paquete |
| CAT_CACHE_LOOKUPS | BIGINT | cat_cache_lookups - Búsquedas de antememoria de catálogo |
| CAT_CACHE_INSERTS | BIGINT | cat_cache_inserts - Inserciones de antememoria de catálogo |

Tabla 138. Información devuelta por la vista administrativa SNAPAPPL y la función de tabla SNAP_GET_APPL_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------------|---------------|---|
| CAT_CACHE_OVERFLOWS | BIGINT | cat_cache_overflows - Desbordamientos de antememoria de catálogo |
| NUM_AGENTS | BIGINT | num_agents - Número de agentes trabajando en una sentencia |
| AGENTS_STOLEN | BIGINT | agents_stolen - Agentes robados |
| ASSOCIATED_AGENTS_TOP | BIGINT | associated_agents_top - Número máximo de agentes asociados |
| APPL_PRIORITY | BIGINT | appl_priority - Prioridad de agente de aplicación |
| APPL_PRIORITY_TYPE | VARCHAR(16) | appl_priority_type - Tipo de prioridad de la aplicación. Esta interfaz devuelve un identificador de texto, basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • DYNAMIC_PRIORITY • FIXED_PRIORITY |
| PREFETCH_WAIT_TIME | BIGINT | prefetch_wait_time - Tiempo esperado para captación previa |
| APPL_SECTION_LOOKUPS | BIGINT | appl_section_lookups - Búsquedas de sección |
| APPL_SECTION_INSERTS | BIGINT | appl_section_inserts - Inserciones de sección |
| LOCKS_WAITING | BIGINT | locks_waiting - Agentes actuales en espera de bloqueos |
| TOTAL_HASH_JOINS | BIGINT | total_hash_joins - Total de uniones hash |
| TOTAL_HASH_LOOPS | BIGINT | total_hash_loops - Total de bucles hash |
| HASH_JOIN_OVERFLOWS | BIGINT | hash_join_overflows - Desbordamientos de uniones hash |
| HASH_JOIN_SMALL_OVERFLOWS | BIGINT | hash_join_small_overflows - Desbordamientos pequeños de uniones hash |
| APPL_IDLE_TIME | BIGINT | appl_idle_time - Tiempo de inactividad de aplicación |
| UOW_LOCK_WAIT_TIME | BIGINT | uow_lock_wait_time - Tiempo total que la unidad de trabajo ha esperado en bloqueos |

Tabla 138. Información devuelta por la vista administrativa SNAPAPPL y la función de tabla SNAP_GET_APPL_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------|---------------|---|
| UOW_COMP_STATUS | VARCHAR(14) | uow_comp_status - Estado de finalización de la unidad de trabajo. Esta interfaz devuelve un identificador de texto, basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • APPL_END • UOWABEND • UOWCOMMIT • UOWDEADLOCK • UOWLOCKTIMEOUT • UOWROLLBACK • UOWUNKNOWN |
| AGENT_USR_CPU_TIME_S | BIGINT | agent_usr_cpu_time - Tiempo de CPU de usuario utilizado por el agente |
| AGENT_USR_CPU_TIME_MS | BIGINT | agent_usr_cpu_time - Tiempo de CPU de usuario utilizado por el agente |
| AGENT_SYS_CPU_TIME_S | BIGINT | agent_sys_cpu_time - Tiempo de CPU de sistema utilizado por el agente |
| AGENT_SYS_CPU_TIME_MS | BIGINT | agent_sys_cpu_time - Tiempo de CPU de sistema utilizado por el agente |
| APPL_CON_TIME | TIMESTAMP | appl_con_time - Indicación de fecha y hora de inicio de petición de conexión |
| CONN_COMPLETE_TIME | TIMESTAMP | conn_complete_time - Indicación de fecha y hora de finalización de petición de conexión |
| LAST_RESET | TIMESTAMP | last_reset - Indicación de fecha y hora de último restablecimiento |
| UOW_START_TIME | TIMESTAMP | uow_start_time - Indicación de fecha y hora de inicio de unidad de trabajo |
| UOW_STOP_TIME | TIMESTAMP | uow_stop_time - Indicación de fecha y hora de detención de unidad de trabajo |
| PREV_UOW_STOP_TIME | TIMESTAMP | prev_uow_stop_time - Indicación de fecha y hora de finalización de unidad de trabajo anterior |
| UOW_ELAPSED_TIME_S | BIGINT | uow_elapsed_time - Tiempo transcurrido de unidad de trabajo más reciente |
| UOW_ELAPSED_TIME_MS | BIGINT | uow_elapsed_time - Tiempo transcurrido de unidad de trabajo más reciente |

Tabla 138. Información devuelta por la vista administrativa SNAPAPPL y la función de tabla SNAP_GET_APPL_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------------------|---------------|--|
| ELAPSED_EXEC_TIME_S | BIGINT | elapsed_exec_time - Tiempo transcurrido de ejecución de sentencia |
| ELAPSED_EXEC_TIME_MS | BIGINT | elapsed_exec_time - Tiempo transcurrido de ejecución de sentencia |
| INBOUND_COMM_ADDRESS | VARCHAR(32) | inbound_comm_address - Dirección de comunicaciones de entrada |
| LOCK_TIMEOUT_VAL | BIGINT | lock_timeout_val - Tiempo de espera excedido de bloqueo (segundos) |
| PRIV_WORKSPACE_NUM_OVERFLOWS | BIGINT | priv_workspace_num_overflows - Desbordamientos de espacio de trabajo privado |
| PRIV_WORKSPACE_SECTION_INSERTS | BIGINT | priv_workspace_section_inserts - Inserciones de sección de espacio de trabajo privado |
| PRIV_WORKSPACE_SECTION_LOOKUPS | BIGINT | priv_workspace_section_lookups - Búsquedas de sección de espacio de trabajo privado |
| PRIV_WORKSPACE_SIZE_TOP | BIGINT | priv_workspace_size_top - Tamaño máximo de espacio de trabajo privado |
| SHR_WORKSPACE_NUM_OVERFLOWS | BIGINT | shr_workspace_num_overflows - Desbordamientos de espacio de trabajo compartido |
| SHR_WORKSPACE_SECTION_INSERTS | BIGINT | shr_workspace_section_inserts - Inserciones de sección de espacio de trabajo compartido |
| SHR_WORKSPACE_SECTION_LOOKUPS | BIGINT | shr_workspace_section_lookups - Búsquedas de sección de espacio de trabajo compartido |
| SHR_WORKSPACE_SIZE_TOP | BIGINT | shr_workspace_size_top - Tamaño máximo de espacio de trabajo compartido |
| DBPARTITIONNUM | SMALLINT | La partición de base de datos desde la que se han recuperado los datos para la fila. |
| CAT_CACHE_SIZE_TOP | BIGINT | cat_cache_size_top - Marca de límite superior de antememoria de catálogo |
| TOTAL_OLAP_FUNCS | BIGINT | Número total de funciones OLAP ejecutadas. |
| OLAP_FUNC_OVERFLOWS | BIGINT | Número de veces que los datos de función OLAP han superado el espacio de pila de clasificación disponible. |

Vista administrativa SNAPBP y función de tabla SNAP_GET_BP_V95 – Recuperar información de instantáneas del grupo lógico bufferpool

La vista administrativa SNAPBP y la función de tabla SNAP_GET_BP_V95 devuelven información sobre agrupaciones de almacenamientos intermedios de una instantánea de bufferpool, en particular el grupo de datos lógicos de bufferpool.

SNAPBP, vista administrativa

Esta vista administrativa permite recuperar la información de instantáneas del grupo de datos lógicos bufferpool para la base de datos conectada actualmente.

La vista administrativa SNAPBP, utilizada con la vista administrativa SNAPBP_PART, ofrece los datos equivalentes al mandato CLP GET SNAPSHOT FOR BUFFERPOOLS ON alias de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 139 en la página 542 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPBP
- Privilegio CONTROL en la vista administrativa SNAPBP
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_BP_V95
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCtrl
- SYSMAINT
- SYSADM

Ejemplo

Recuperar grabaciones en el índice y los datos de todas las agrupaciones de almacenamientos intermedios de la base de datos conectada actualmente.

```
SELECT SUBSTR(DB_NAME,1,8) AS DB_NAME,SUBSTR(BP_NAME,1,15)
      AS BP_NAME,POOL_DATA_WRITES,POOL_INDEX_WRITES
FROM SYSIBMADM.SNAPBP
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| DB_NAME | BP_NAME | POOL_DATA_WRITES | POOL_INDEX_WRITES |
|---------|----------------|------------------|-------------------|
| TEST | IBMDEFAULTBP | 0 | 0 |
| TEST | IBMSYSTEMBP4K | 0 | 0 |
| TEST | IBMSYSTEMBP8K | 0 | 0 |
| TEST | IBMSYSTEMBP16K | 0 | 0 |
| TEST | IBMSYSTEMBP32K | 0 | 0 |

5 registro(s) seleccionado(s)

SNAP_GET_BP_V95, función de tabla

La función de tabla SNAP_GET_BP_V95 devuelve la misma información que la vista administrativa SNAPBP, pero permite recuperar la información de una base de datos concreta en una partición específica de la base de datos, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_BP_V95, utilizada con la función de tabla SNAP_GET_BP_PART, ofrece los datos equivalentes al mandato CLP GET SNAPSHOT FOR ALL BUFFERPOOLS.

Consulte la Tabla 139 en la página 542 para ver una lista completa de la información que se puede devolver.

Sintaxis

```

▶▶ SNAP_GET_BP_V95 ( ( nombrebd
                    [ , númparticiónbd ] )
▶▶

```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique una serie vacía para tomar la instantánea desde la base de datos conectada actualmente. Especifique un valor NULL para tomar la instantánea desde todas las bases de datos que se encuentran en la misma instancia que la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es

posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_BP_V95 tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_BP_V95
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar lecturas lógicas y físicas para todas las agrupaciones de almacenamientos intermedios de todas las bases de datos activas en la partición de base de datos conectada actualmente.

```
SELECT SUBSTR(T.DB_NAME,1,10) AS DB_NAME,
       SUBSTR(T.BP_NAME,1,20) AS BP_NAME,
       (T.POOL_DATA_L_READS+T.POOL_INDEX_L_READS) AS TOTAL_LOGICAL_READS,
       (T.POOL_DATA_P_READS+T.POOL_INDEX_P_READS) AS TOTAL_PHYSICAL_READS,
       T.DBPARTITIONNUM
FROM TABLE(SNAP_GET_BP_V95(CAST(NULL AS VARCHAR(128)), -1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
DB_NAME      BP_NAME      TOTAL_LOGICAL_READS  ...
-----
SAMPLE      IBMDEFAULTBP      0 ...
TOOLSDB     IBMDEFAULTBP      0 ...
TOOLSDB     BP32K0000         0 ...
```

3 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```
... TOTAL_PHYSICAL_READS DBPARTITIONNUM
... -----
...                0                0
...                0                0
...                0                0
```

Información devuelta

Tabla 139. Información devuelta por la vista administrativa SNAPBP y la función de tabla SNAP_GET_BP_V95

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |

Tabla 139. Información devuelta por la vista administrativa SNAPBP y la función de tabla SNAP_GET_BP_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------|---------------|---|
| BP_NAME | VARCHAR(128) | bp_name - Nombre de la agrupación de almacenamientos intermedios. |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| DB_PATH | VARCHAR(1024) | db_path - Vía de acceso de base de datos |
| INPUT_DB_ALIAS | VARCHAR(128) | input_db_alias - Alias de base de datos de entrada |
| POOL_DATA_L_READS | BIGINT | pool_data_l_reads - Lecturas lógicas de datos de la agrupación de almacenamientos intermedios |
| POOL_DATA_P_READS | BIGINT | pool_data_p_reads - Lecturas físicas de datos de la agrupación de almacenamientos intermedios |
| POOL_DATA_WRITES | BIGINT | pool_data_writes - Grabaciones de datos de la agrupación de almacenamientos intermedios |
| POOL_INDEX_L_READS | BIGINT | pool_index_l_reads - Lecturas lógicas de índices de la agrupación de almacenamientos intermedios |
| POOL_INDEX_P_READS | BIGINT | pool_index_p_reads - Lecturas físicas de índices de la agrupación de almacenamientos intermedios |
| POOL_INDEX_WRITES | BIGINT | pool_index_writes - Grabaciones de índices de la agrupación de almacenamientos intermedios |
| POOL_XDA_L_READS | BIGINT | pool_xda_l_reads - Lecturas lógicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_XDA_P_READS | BIGINT | pool_xda_p_reads - Lecturas físicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_XDA_WRITES | BIGINT | pool_xda_writes - Grabaciones de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_READ_TIME | BIGINT | pool_read_time - Tiempo total de lectura física de la agrupación de almacenamientos intermedios |
| POOL_WRITE_TIME | BIGINT | pool_write_time - Tiempo total de grabación física de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_READS | BIGINT | pool_async_data_reads - Lecturas de datos asíncronas de agrupación de almacenamientos intermedios |

Tabla 139. Información devuelta por la vista administrativa SNAPBP y la función de tabla SNAP_GET_BP_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|----------------------------|---------------|---|
| POOL_ASYNC_DATA_WRITES | BIGINT | pool_async_data_writes - Grabaciones de datos asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_READS | BIGINT | pool_async_index_reads - Lecturas de índice asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_WRITES | BIGINT | pool_async_index_writes - Grabaciones de índice asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_XDA_READS | BIGINT | pool_async_xda_reads - Lecturas de datos XDA asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_XDA_WRITES | BIGINT | pool_async_xda_writes - Grabaciones de datos XDA asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_READ_TIME | BIGINT | pool_async_read_time - Tiempo de lectura asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_WRITE_TIME | BIGINT | pool_async_write_time - Tiempo de grabación asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_READ_REQS | BIGINT | pool_async_data_read_reqs - Peticiones de lectura asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_READ_REQS | BIGINT | pool_async_index_read_reqs - Peticiones de lectura de índice asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_XDA_READ_REQS | BIGINT | pool_async_xda_read_reqs - Peticiones de lectura XDA asíncrona de agrupación de almacenamientos intermedios |
| DIRECT_READS | BIGINT | direct_reads - Lecturas directas de la base de datos |
| DIRECT_WRITES | BIGINT | direct_writes - Grabaciones directas en la base de datos |
| DIRECT_READ_REQS | BIGINT | direct_read_reqs - Peticiones de lectura directa |
| DIRECT_WRITE_REQS | BIGINT | direct_write_reqs - Peticiones de grabación directa |
| DIRECT_READ_TIME | BIGINT | direct_read_time - Tiempo de lectura directa |

Tabla 139. Información devuelta por la vista administrativa SNAPBP y la función de tabla SNAP_GET_BP_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|--|
| DIRECT_WRITE_TIME | BIGINT | direct_write_time - Tiempo de grabación directa |
| UNREAD_PREFETCH_PAGES | BIGINT | unread_prefetch_pages - Páginas de captación previa no leídas |
| FILES_CLOSED | BIGINT | files_closed - Archivos de base de datos cerrados |
| POOL_TEMP_DATA_L_READS | BIGINT | pool_temp_data_l_reads - Lecturas lógicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_P_READS | BIGINT | pool_temp_data_p_reads - Lecturas físicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_L_READS | BIGINT | pool_temp_index_l_reads - Lecturas lógicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_P_READS | BIGINT | pool_temp_index_p_reads - Lecturas físicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_L_READS | BIGINT | pool_temp_xda_l_reads - Lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_P_READS | BIGINT | pool_temp_xda_p_reads - Elemento del supervisor de las lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |
| POOL_NO_VICTIM_BUFFER | BIGINT | pool_no_victim_buffer - Almacenamiento intermedio sin víctimas de la agrupación de almacenamientos intermedios |
| PAGES_FROM_BLOCK_IOS | BIGINT | pages_from_block_ios - Número total de páginas leídas por E/S de bloque |
| PAGES_FROM_VECTORED_IOS | BIGINT | pages_from_vectored_ios - Número total de páginas leídas por E/S de vector |
| VECTORED_IOS | BIGINT | vectored_ios - Número de peticiones de E/S de vector |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa SNAPBP_PART y función de tabla SNAP_GET_BP_PART – Recuperar información de instantáneas del grupo de datos lógicos bufferpool_nodeinfo

La vista administrativa SNAPBP_PART y la función de tabla SNAP_GET_BP_PART devuelven información acerca de las agrupaciones de almacenamientos intermedios a partir de una instantánea determinada, el grupo de datos lógico bufferpool_nodeinfo.

SNAPBP_PART, vista administrativa

Esta vista administrativa permite recuperar la información de instantáneas del grupo de datos lógicos bufferpool_nodeinfo para la base de datos conectada actualmente.

La vista administrativa SNAPBP_PART, utilizada con la vista administrativa SNAPBP, ofrece los datos equivalentes al mandato CLP GET SNAPSHOT FOR BUFFERPOOLS ON alias de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 140 en la página 549 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPBP_PART
- Privilegio CONTROL en la vista administrativa SNAPBP_PART
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_BP_PART
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar datos de todas las agrupaciones de almacenamientos intermedios cuando esté conectado con la base de datos SAMPLE.

```
SELECT SUBSTR(DB_NAME,1,8) AS DB_NAME, SUBSTR(BP_NAME,1,15) AS BP_NAME,  
       BP_CUR_BUFFSZ, BP_NEW_BUFFSZ, BP_PAGES_LEFT_TO_REMOVE, BP_TBSP_USE_COUNT  
FROM SYSIBMADM.SNAPBP_PART
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| DB_NAME | BP_NAME | BP_CUR_BUFFSZ | BP_NEW_BUFFSZ | ... |
|---------|----------------|---------------|---------------|-----|
| SAMPLE | IBMDEFAULTBP | 1000 | 1000 | ... |
| SAMPLE | IBMSYSTEMBP4K | 16 | 16 | ... |
| SAMPLE | IBMSYSTEMBP8K | 16 | 16 | ... |
| SAMPLE | IBMSYSTEMBP16K | 16 | 16 | ... |

4 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

| ... | BP_PAGES_LEFT_TO_REMOVE | BP_TBSP_USE_COUNT |
|-----|-------------------------|-------------------|
| ... | 0 | 3 |
| ... | 0 | 0 |
| ... | 0 | 0 |
| ... | 0 | 0 |
| ... | | |

SNAP_GET_BP_PART, función de tabla

La función de tabla SNAP_GET_BP_PART devuelve la misma información que la vista administrativa SNAPBP_PART, pero permite recuperar la información para una base de datos concreta en una partición específica de la base de datos, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

Junto con la función de tabla SNAP_GET_BP_V95, la función de tabla SNAP_GET_BP_PART proporciona los datos equivalentes al mandato CLP GET SNAPSHOT FOR ALL BUFFERPOOLS.

Consulte la Tabla 140 en la página 549 para ver una lista completa de la información que se puede devolver.

Sintaxis

```

▶▶ SNAP_GET_BP_PART ( ( nombrebd [ , numparticiónbd ] ) )

```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique una serie vacía para tomar la instantánea desde la base de datos conectada actualmente. Especifique un valor NULL para tomar la instantánea de todas las agrupaciones de almacenamientos intermedios de todas las bases de datos de la misma instancia que la base de datos conectada actualmente.

numparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de

datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_BP_PART toma una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_BP_PART
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar datos de todas las agrupaciones de almacenamientos intermedios de todas las bases de datos activas cuando estén conectadas con la base de datos SAMPLE.

```
SELECT SUBSTR(DB_NAME,1,8) AS DB_NAME, SUBSTR(BP_NAME,1,15) AS BP_NAME,
       BP_CUR_BUFFSZ, BP_NEW_BUFFSZ, BP_PAGES_LEFT_TO_REMOVE, BP_TBSP_USE_COUNT
FROM TABLE(SNAP_GET_BP_PART(CAST(NULL AS VARCHAR(128)),-1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| DB_NAME | BP_NAME | BP_CUR_BUFFSZ | BP_NEW_BUFFSZ | ... |
|---------|----------------|---------------|---------------|-----|
| SAMPLE | IBMDEFAULTBP | 250 | 250 | ... |
| SAMPLE | IBMSYSTEMBP4K | 16 | 16 | ... |
| SAMPLE | IBMSYSTEMBP8K | 16 | 16 | ... |
| SAMPLE | IBMSYSTEMBP16K | 16 | 16 | ... |
| SAMPLE | IBMSYSTEMBP32K | 16 | 16 | ... |
| TESTDB | IBMDEFAULTBP | 250 | 250 | ... |
| TESTDB | IBMSYSTEMBP4K | 16 | 16 | ... |
| TESTDB | IBMSYSTEMBP8K | 16 | 16 | ... |
| TESTDB | IBMSYSTEMBP16K | 16 | 16 | ... |
| TESTDB | IBMSYSTEMBP32K | 16 | 16 | ... |

...

Datos de salida de esta consulta (continuación).

| ... | BP_PAGES_LEFT_TO_REMOVE | BP_TBSP_USE_COUNT |
|-----|-------------------------|-------------------|
| ... | 0 | 3 |


```

...           0           0
...           0           0
...           0           0
...           0           0
...           0           3
...           0           0
...           0           0
...           0           0
...           0           0
...

```

Información devuelta

Tabla 140. Información devuelta por la vista administrativa SNAPBP_PART y la función de tabla SNAP_GET_BP_PART

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| BP_NAME | VARCHAR(128) | bp_name - Nombre de la agrupación de almacenamientos intermedios. |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| BP_CUR_BUFFSZ | BIGINT | bp_cur_buffsz - Tamaño actual de agrupación de almacenamientos intermedios |
| BP_NEW_BUFFSZ | BIGINT | bp_new_buffsz - Tamaño de agrupación de almacenamientos intermedios nuevo |
| BP_PAGES_LEFT_TO_REMOVE | BIGINT | bp_pages_left_to_remove - Número de páginas restantes para eliminar |
| BP_TBSP_USE_COUNT | BIGINT | bp_tbsp_use_count - Número de espacios de tablas correlacionados con la agrupación de almacenamientos intermedios |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa SNAPCONTAINER y función de tabla SNAP_GET_CONTAINER_V91 – Recuperar información acerca de la instantánea de grupo de datos lógicos tablespace_container

La vista administrativa SNAPCONTAINER y la función de tabla SNAP_GET_CONTAINER_V91 devuelve información de la instantánea del espacio de tablas del grupo de datos lógicos tablespace_container.

SNAPCONTAINER, vista administrativa

Esta vista administrativa le permite recuperar información de instantánea de grupo de datos lógicos tablespace_container para la base de datos conectada actualmente.

La vista administrativa SNAPCONTAINER, utilizada con las vistas administrativas SNAPTBSP, SNAPTBSP_PART, SNAPTBSP QUIESCER y SNAPTBSP_RANGE, devuelve datos que son equivalentes a los datos que devuelve el mandato CLP GET SNAPSHOT FOR TABLESPACES ON alias de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 141 en la página 553 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPCONTAINER
- Privilegio CONTROL en la vista administrativa SNAPCONTAINER
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_CONTAINER_V91
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar información detallada acerca de los contenedores de espacio de tablas para todas las particiones de base de datos de la base de datos conectada actualmente.

```
SELECT SNAPSHOT_TIMESTAMP, SUBSTR(TBSP_NAME, 1, 15) AS TBSP_NAME,
       TBSP_ID, SUBSTR(CONTAINER_NAME, 1, 20) AS CONTAINER_NAME,
       CONTAINER_ID, CONTAINER_TYPE, ACCESSIBLE, DBPARTITIONNUM
FROM SYSIBMADM.SNAPCONTAINER ORDER BY DBPARTITIONNUM
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
SNAPSHOT_TIMESTAMP      TBSP_NAME      TBSP_ID      ...
-----
2006-01-08-16.49.24.639945 SYSCATSPACE      0 ...
2006-01-08-16.49.24.639945 TEMPSPACE1      1 ...
2006-01-08-16.49.24.639945 USERSPACE1      2 ...
2006-01-08-16.49.24.639945 SYSTOOLSPACE      3 ...
2006-01-08-16.49.24.640747 TEMPSPACE1      1 ...
2006-01-08-16.49.24.640747 USERSPACE1      2 ...
2006-01-08-16.49.24.639981 TEMPSPACE1      1 ...
2006-01-08-16.49.24.639981 USERSPACE1      2 ...
...
8 registro(s) seleccionado(s).
```

Datos de salida de esta consulta (continuación).

```
... CONTAINER_NAME      CONTAINER_ID      CONTAINER_TYPE      ...
-----
... /home/swalkty/swalkt      0 FILE_EXTENT_TAG      ...
```

```

... /home/swalkty/swalkt          0 PATH          ...
... /home/swalkty/swalkt          0 FILE_EXTENT_TAG ...
... /home/swalkty/swalkt          0 FILE_EXTENT_TAG ...
... /home/swalkty/swalkt          0 PATH          ...
... /home/swalkty/swalkt          0 FILE_EXTENT_TAG ...
... /home/swalkty/swalkt          0 PATH          ...
... /home/swalkty/swalkt          0 FILE_EXTENT_TAG ...

```

Datos de salida de esta consulta (continuación).

```

... ACCESSIBLE DBPARTITIONNUM
... -----
...          1          0
...          1          0
...          1          0
...          1          0
...          1          1
...          1          1
...          1          2
...          1          2

```

SNAP_GET_CONTAINER_V91, función de tabla

La función de tabla SNAP_GET_CONTAINER_V91 devuelve la misma información que la vista administrativa SNAPCONTAINER, pero le permite recuperar la información para una base de datos específica de una partición de base de datos específica, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_CONTAINER_V91, utilizada con las funciones de tabla SNAP_GET_TBSP_V91, SNAP_GET_TBSP_PART_V91, SNAP_GET_TBSP QUIESCER y SNAP_GET_TBSP_RANGE, devuelve datos que son equivalentes a los que devuelve el mandato CLP GET SNAPSHOT FOR TABLESPACES ON alias de base de datos.

Consulte la Tabla 141 en la página 553 para ver una lista completa de la información que se puede devolver.

Sintaxis

```

▶▶ SNAP_GET_CONTAINER_V91 (—nombrebd —————)
                                |
                                |, númparticiónbd

```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique NULL o una serie vacía para tomar la instantánea de la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de

datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_CONTAINER_V91 tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_CONTAINER_V91
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar información detallada acerca de los contenedores de espacio de tablas de la base de datos conectada actualmente de la partición de base de datos conectada actualmente.

```
SELECT SNAPSHOT_TIMESTAMP, TBSP_NAME, TBSP_ID, CONTAINER_NAME,
       CONTAINER_ID, CONTAINER_TYPE, ACCESSIBLE
FROM TABLE(SNAP_GET_CONTAINER_V91(' ', -1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| SNAPSHOT_TIMESTAMP | TBSP_NAME | TBSP_ID | ... |
|----------------------------|-------------------|---------|-----|
| 2005-04-25-14.42.10.899253 | SYSCATSPACE | 0 | ... |
| 2005-04-25-14.42.10.899253 | TEMPSPACE1 | 1 | ... |
| 2005-04-25-14.42.10.899253 | USERSPACE1 | 2 | ... |
| 2005-04-25-14.42.10.899253 | SYSTOOLSPACE | 3 | ... |
| 2005-04-25-14.42.10.899253 | MYTEMP | 4 | ... |
| 2005-04-25-14.42.10.899253 | WHATSNEWTEMPSPACE | 5 | ... |

Datos de salida de esta consulta (continuación).

| ... | CONTAINER_NAME | CONTAINER_ID | ... |
|-----|---------------------------------------|--------------|-----|
| ... | D:\DB2\NODE0000\SQL00002\SQLT0000.0 | 0 | ... |
| ... | D:\DB2\NODE0000\SQL00002\SQLT0001.0 | 0 | ... |
| ... | D:\DB2\NODE0000\SQL00002\SQLT0002.0 | 0 | ... |
| ... | D:\DB2\NODE0000\SQL00002\SYSTOOLSPACE | 0 | ... |
| ... | D:\DB2\NODE0000\SQL003 | 0 | ... |
| ... | d:\DGTTsWhatsNewContainer | 0 | ... |

Datos de salida de esta consulta (continuación).

```

... CONTAINER_TYPE ACCESSIBLE
... -----
... CONT_PATH          1
... CONT_PATH          1
... CONT_PATH          1
... CONT_PATH          1
... CONT_PATH          1
... CONT_PATH          1

```

Información devuelta

NOTA: Para que se devuelva la información del sistema de archivos, debe activarse el conmutador de supervisor del gestor de bases de datos BUFFERPOOL.

Tabla 141. Información devuelta por la vista administrativa SNAPCONTAINER y la función de tabla SNAP_GET_CONTAINER_V91

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| TBSP_NAME | VARCHAR(128) | tablespace_name - Nombre de espacio de tablas |
| TBSP_ID | BIGINT | tablespace_id - Identificación de espacio de tablas |
| CONTAINER_NAME | VARCHAR(256) | container_name - Nombre de contenedor |
| CONTAINER_ID | BIGINT | container_id - Identificación de contenedor |
| CONTAINER_TYPE | VARCHAR(16) | container_type - Tipo de contenedor. Es un identificador de texto basado en las sentencias define de sqlutil.h y es uno de los siguientes: <ul style="list-style-type: none"> • DISK_EXTENT_TAG • DISK_PAGE_TAG • FILE_EXTENT_TAG • FILE_PAGE_TAG • PATH |
| TOTAL_PAGES | BIGINT | container_total_pages - Total de páginas en contenedor |
| USABLE_PAGES | BIGINT | container_usable_pages - Páginas utilizables en contenedor |
| ACCESSIBLE | SMALLINT | container_accessible - Accesibilidad de contenedor |
| STRIPE_SET | BIGINT | container_stripe_set - Conjunto de bandas |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |
| FS_ID | VARCHAR(22) | fs_id - Número de identificación del sistema de archivos exclusivo |

Tabla 141. Información devuelta por la vista administrativa SNAPCONTAINER y la función de tabla SNAP_GET_CONTAINER_V91 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|---|
| FS_TOTAL_SIZE | BIGINT | fs_total_size - Tamaño total de un sistema de archivos |
| FS_USED_SIZE | BIGINT | fs_used_size - Cantidad de espacio utilizada por un sistema de archivos |

Vista administrativa SNAPDB y función de tabla SNAP_GET_DB_V95 – Recuperar información de instantánea desde el grupo lógico dbase

La “SNAPDB, vista administrativa” y la “SNAP_GET_DB_V95, función de tabla” en la página 555 devuelven información de instantánea desde el grupo lógico de base de datos (dbase).

SNAPDB, vista administrativa

Esta vista administrativa le permite recuperar información de instantánea desde el grupo lógico dbase para la base de datos conectada actualmente.

La vista administrativa SNAPDB, que se utiliza junto con las vistas administrativas SNAPDB_MEMORY_POOL, SNAPDETAILOG, SNAPHADR y SNAPSTORAGE_PATHS, proporciona información que es equivalente a la información que proporciona el mandato CLP GET SNAPSHOT FOR DATABASE ON alias de base de datos.

El esquema es SYSIBMADM.

En la Tabla 142 en la página 558 encontrará una lista completa de la información que se devuelve.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPDB
- Privilegio CONTROL en la vista administrativa SNAPDB
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_DB_V95
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplos

Recuperar el estado, la plataforma, la ubicación y el tiempo de conexión para todas las particiones de base de datos de la base de datos conectada actualmente.

```
SELECT SUBSTR(DB_NAME, 1, 20) AS DB_NAME, DB_STATUS, SERVER_PLATFORM,
       DB_LOCATION, DB_CONN_TIME, DBPARTITIONNUM
FROM SYSIBMADM.SNAPDB ORDER BY DBPARTITIONNUM
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| DB_NAME | DB_STATUS | SERVER_PLATFORM | DB_LOCATION | ... |
|---------|-----------|-----------------|-------------|-----|
| TEST | ACTIVE | AIX64 | LOCAL | ... |
| TEST | ACTIVE | AIX64 | LOCAL | ... |
| TEST | ACTIVE | AIX64 | LOCAL | ... |

3 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

| DB_CONN_TIME | DBPARTITIONNUM |
|----------------------------|----------------|
| 2006-01-08-16.48.30.665477 | 0 |
| 2006-01-08-16.48.34.005328 | 1 |
| 2006-01-08-16.48.34.007937 | 2 |

Esta rutina se puede utilizar realizando la siguiente llamada en la línea de mandatos:

```
SELECT TOTAL_OLAP_FUNCS, OLAP_FUNC_OVERFLOWS, ACTIVE_OLAP_FUNCS
FROM SYSIBMADM.SNAPDB
```

| TOTAL_OLAP_FUNCS | OLAP_FUNC_OVERFLOWS | ACTIVE_OLAP_FUNCS |
|------------------|---------------------|-------------------|
| 7 | 2 | 1 |

1 registro(s) seleccionado(s).

Después de ejecutar una carga de trabajo, el usuario puede utilizar la siguiente consulta:

```
SELECT STATS_CACHE_SIZE, STATS_FABRICATIONS, SYNC_RUNSTATS,
       ASYNC_RUNSTATS, STATS_FABRICATE_TIME, SYNC_RUNSTATS_TIME
FROM SYSIBMADM.SNAPDB
```

| STATS_CACHE_SIZE | STATS_FABRICATIONS | SYNC_RUNSTATS | ASYNC_RUNSTATS | ... |
|------------------|--------------------|---------------|----------------|-----|
| 128 | 2 | 1 | 0 | ... |

| STATS_FABRICATE_TIME | SYNC_RUNSTATS_TIME |
|----------------------|--------------------|
| 10 | 100 |

1 registro(s) seleccionado(s).

SNAP_GET_DB_V95, función de tabla

La función de tabla SNAP_GET_DB_V95 devuelve la misma información que la vista administrativa SNAPDB.

La función de tabla SNAP_GET_DB_V91, utilizada con las funciones de tabla SNAP_GET_DB_MEMORY_POOL, SNAP_GET_DETAILLOG_V95,

SNAP_GET_HADR y SNAP_GET_STORAGE_PATHS, proporciona información que es equivalente a la que proporciona el mandato CLP GET SNAPSHOT FOR ALL DATABASES.

En la Tabla 142 en la página 558 encontrará una lista completa de la información que se devuelve.

Sintaxis

```
▶▶ SNAP_GET_DB_V95 (—nombrebd [ , númparticiónbd ] )
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique una serie vacía para tomar la instantánea desde la base de datos conectada actualmente. Especifique un valor NULL para tomar la instantánea desde todas las bases de datos que se encuentran en la misma instancia que la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_DB_V95 tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_DB_V95
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL

- SYSMANT
- SYSADM

Ejemplos

Ejemplo 1: Recuperar el estado, la plataforma, la ubicación y el tiempo de conexión como vista de agregación de todas las particiones de base de datos de la base de datos conectada actualmente.

```
SELECT SUBSTR(DB_NAME, 1, 20) AS DB_NAME, DB_STATUS, SERVER_PLATFORM,
       DB_LOCATION, DB_CONN_TIME FROM TABLE(SNAP_GET_DB_V95(' ', -2)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
DB_NAME      DB_STATUS    SERVER_PLATFORM ...
-----
SAMPLE      ACTIVE      AIX64          ...
```

1 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```
... DB_LOCATION DB_CONN_TIME
... -----
... LOCAL      2005-07-24-22.09.22.013196
```

Ejemplo 2: Recuperar el estado, la plataforma, la ubicación y el tiempo de conexión como vista de agregación de todas las particiones de base de datos para todas las bases de datos activas que se encuentran en la misma instancia que contiene la base de datos conectada actualmente.

```
SELECT SUBSTR(DB_NAME, 1, 20) AS DB_NAME, DB_STATUS, SERVER_PLATFORM,
       DB_LOCATION, DB_CONN_TIME
       FROM TABLE(SNAP_GET_DB_V95(CAST (NULL AS VARCHAR(128)), -2)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
DB_NAME      DB_STATUS    SERVER_PLATFORM ...
-----
TOOLSDB     ACTIVE      AIX64          ...
SAMPLE     ACTIVE      AIX64          ...
```

Datos de salida de esta consulta (continuación).

```
... DB_LOCATION DB_CONN_TIME
... -----
... LOCAL      2005-07-24-22.26.54.396335
... LOCAL      2005-07-24-22.09.22.013196
```

Ejemplo 3: Esta rutina se puede utilizar realizando la siguiente llamada en la línea de mandatos:

Cuando está conectado a una base de datos:

```
SELECT TOTAL_OLAP_FUNCS, OLAP_FUNC_OVERFLOWS, ACTIVE_OLAP_FUNCS
       FROM TABLE (SNAP_GET_DB_V95(' ', 0)) AS T
```

La salida se parecerá a la siguiente:

```
TOTAL_OLAP_FUNCS  OLAP_FUNC_OVERFLOWS  ACTIVE_OLAP_FUNCS
-----
                7                2                1
```

1 registro(s) seleccionado(s).

Ejemplo 4: Después de ejecutar una carga de trabajo, el usuario puede utilizar la siguiente consulta con la función de tabla.

```
SELECT STATS_CACHE_SIZE, STATS_FABRICATIONS, SYNC_RUNSTATS,
       ASYNC_RUNSTATS, STATS_FABRICATE_TIME, SYNC_RUNSTATS_TIME
FROM TABLE (SNAP_GET_DB_V95('mytestdb', -1)) AS SNAPDB
```

```
STATS_CACHE_SIZE  STATS_FABRICATIONS  SYNC_RUNSTATS  ASYNC_RUNSTATS  ...
-----
                200                    1                2                0  ...
```

Continuación

```
...STATS_FABRICATE_TIME  SYNC_RUNSTATS_TIME
-----
...                2                32
```

1 registro(s) seleccionado(s).

Metadatos de la vista administrativa SNAPDB y la función de tabla SNAP_GET_DB_V95

Tabla 142. Información devuelta por la vista administrativa SNAPDB y la función de tabla SNAP_GET_DB_V95

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| DB_PATH | VARCHAR(1024) | db_path - Vía de acceso de base de datos |
| INPUT_DB_ALIAS | VARCHAR(128) | input_db_alias - Alias de base de datos de entrada |
| DB_STATUS | VARCHAR(12) | db_status - Estado de base de datos. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • ACTIVE • QUIESCE_PEND • QUIESCED • ROLLFWD |
| CATALOG_PARTITION | SMALLINT | catalog_node - Número de nodo de catálogo |
| CATALOG_PARTITION_NAME | VARCHAR(128) | catalog_node_name - Nombre de red de nodo de catálogo |

Tabla 142. Información devuelta por la vista administrativa SNAPDB y la función de tabla SNAP_GET_DB_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|---|
| SERVER_PLATFORM | VARCHAR(12) | <p>server_platform - Sistema operativo del servidor. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes:</p> <ul style="list-style-type: none"> • AIX • AIX64 • AS400_DRDA • DOS • DYNIX • HP • HP64 • HPIA • HPIA64 • LINUX • LINUX390 • LINUXIA64 • LINUXPPC • LINUXPPC64 • LINUXX8664 • LINUXZ64 • MAC • MVS_DRDA • NT • NT64 • OS2 • OS390 • SCO • SGI • SNI • SUN • SUN64 • UNKNOWN • UNKNOWN_DRDA • VM_DRDA • VSE_DRDA • WINDOWS |
| DB_LOCATION | VARCHAR(12) | <p>db_location - Ubicación de la base de datos. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h, y es uno de los siguientes:</p> <ul style="list-style-type: none"> • LOCAL • REMOTE |
| DB_CONN_TIME | TIMESTAMP | <p>db_conn_time - Indicación de fecha y hora de activación de base de datos</p> |

Tabla 142. Información devuelta por la vista administrativa SNAPDB y la función de tabla SNAP_GET_DB_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|--|
| LAST_RESET | TIMESTAMP | last_reset - Indicación de fecha y hora de último restablecimiento |
| LAST_BACKUP | TIMESTAMP | last_backup - Indicación de fecha y hora de última copia de seguridad |
| CONNECTIONS_TOP | BIGINT | connections_top - Número máximo de conexiones simultáneas |
| TOTAL_CONS | BIGINT | total_cons - Conexiones desde la activación de base de datos |
| TOTAL_SEC_CONS | BIGINT | total_sec_cons - Conexiones secundarias |
| APPLS_CUR_CONS | BIGINT | appls_cur_cons - Aplicaciones conectadas actualmente |
| APPLS_IN_DB2 | BIGINT | appls_in_db2 - Aplicaciones que se ejecutan en la base de datos actualmente |
| NUM_ASSOC_AGENTS | BIGINT | num_assoc_agents - Número de agentes asociados |
| AGENTS_TOP | BIGINT | agents_top - Número de agentes creados |
| COORD_AGENTS_TOP | BIGINT | coord_agents_top - Número máximo de agentes de coordinación |
| LOCKS_HELD | BIGINT | locks_held - Bloqueos retenidos |
| LOCK_WAITS | BIGINT | lock_waits - Esperas de bloqueo |
| LOCK_WAIT_TIME | BIGINT | lock_wait_time - Tiempo esperado en bloqueos |
| LOCK_LIST_IN_USE | BIGINT | lock_list_in_use - Memoria total de lista de bloqueos en uso |
| DEADLOCKS | BIGINT | deadlocks - Puntos muertos detectados |
| LOCK_ESCALS | BIGINT | lock_escals - Número de escalamientos de bloqueo |
| X_LOCK_ESCALS | BIGINT | x_lock_escals - Escalas de bloqueo exclusivas |
| LOCKS_WAITING | BIGINT | locks_waiting - Agentes actuales en espera de bloqueos |
| LOCK_TIMEOUTS | BIGINT | lock_timeouts - Número de tiempos de espera de bloqueo |
| NUM_INDOUBT_TRANS | BIGINT | num_indoubt_trans - Número de transacciones dudosas |
| SORT_HEAP_ALLOCATED | BIGINT | sort_heap_allocated - Total de pila de clasificación asignada |
| SORT_SHRHEAP_ALLOCATED | BIGINT | sort_shrheap_allocated - Pila de compartimiento de clasificación asignada actualmente |
| SORT_SHRHEAP_TOP | BIGINT | sort_shrheap_top - Marca de límite superior de pila de compartimiento de clasificación |
| POST_SHRTHRESHOLD_SORTS | BIGINT | post_shrthreshold_sorts - Clasificaciones de umbral compartidas posteriores |
| TOTAL_SORTS | BIGINT | total_sorts - Clasificaciones totales |

Tabla 142. Información devuelta por la vista administrativa SNAPDB y la función de tabla SNAP_GET_DB_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|--|
| TOTAL_SORT_TIME | BIGINT | total_sort_time - Tiempo de clasificación total |
| SORT_OVERFLOWS | BIGINT | sort_overflows - Desbordamientos de clasificación |
| ACTIVE_SORTS | BIGINT | active_sorts - Clasificaciones activas |
| POOL_DATA_L_READS | BIGINT | pool_data_l_reads - Lecturas lógicas de datos de la agrupación de almacenamientos intermedios |
| POOL_DATA_P_READS | BIGINT | pool_data_p_reads - Lecturas físicas de datos de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_L_READS | BIGINT | pool_temp_data_l_reads - Lecturas lógicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_P_READS | BIGINT | pool_temp_data_p_reads - Lecturas físicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_READS | BIGINT | pool_async_data_reads - Lecturas de datos asíncronas de agrupación de almacenamientos intermedios |
| POOL_DATA_WRITES | BIGINT | pool_data_writes - Grabaciones de datos de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_WRITES | BIGINT | pool_async_data_writes - Grabaciones de datos asíncronas de agrupación de almacenamientos intermedios |
| POOL_INDEX_L_READS | BIGINT | pool_index_l_reads - Lecturas lógicas de índices de la agrupación de almacenamientos intermedios |
| POOL_INDEX_P_READS | BIGINT | pool_index_p_reads - Lecturas físicas de índices de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_L_READS | BIGINT | pool_temp_index_l_reads - Lecturas lógicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_P_READS | BIGINT | pool_temp_index_p_reads - Lecturas físicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_READS | BIGINT | pool_async_index_reads - Lecturas de índice asíncronas de agrupación de almacenamientos intermedios |
| POOL_INDEX_WRITES | BIGINT | pool_index_writes - Grabaciones de índices de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_WRITES | BIGINT | pool_async_index_writes - Grabaciones de índice asíncronas de agrupación de almacenamientos intermedios |

Tabla 142. Información devuelta por la vista administrativa SNAPDB y la función de tabla SNAP_GET_DB_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|----------------------------|---------------|--|
| POOL_XDA_P_READS | BIGINT | pool_xda_p_reads - Lecturas físicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_XDA_L_READS | BIGINT | pool_xda_l_reads - Lecturas lógicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_XDA_WRITES | BIGINT | pool_xda_writes - Grabaciones de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_XDA_READS | BIGINT | pool_async_xda_reads - Lecturas de datos XDA asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_XDA_WRITES | BIGINT | pool_async_xda_writes - Grabaciones de datos XDA asíncronas de agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_P_READS | BIGINT | pool_temp_xda_p_reads - Elemento del supervisor de las lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_L_READS | BIGINT | pool_temp_xda_l_reads - Lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |
| POOL_READ_TIME | BIGINT | pool_read_time - Tiempo total de lectura física de la agrupación de almacenamientos intermedios |
| POOL_WRITE_TIME | BIGINT | pool_write_time - Tiempo total de grabación física de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_READ_TIME | BIGINT | pool_async_read_time - Tiempo de lectura asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_WRITE_TIME | BIGINT | pool_async_write_time - Tiempo de grabación asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_READ_REQS | BIGINT | pool_async_data_read_reqs - Peticiones de lectura asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_READ_REQS | BIGINT | pool_async_index_read_reqs - Peticiones de lectura de índice asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_XDA_READ_REQS | BIGINT | pool_async_xda_read_reqs - Peticiones de lectura XDA asíncrona de agrupación de almacenamientos intermedios |
| POOL_NO_VICTIM_BUFFER | BIGINT | pool_no_victim_buffer - Almacenamiento intermedio sin víctimas de la agrupación de almacenamientos intermedios |

Tabla 142. Información devuelta por la vista administrativa SNAPDB y la función de tabla SNAP_GET_DB_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|--|
| POOL_LSN_GAP_CLNS | BIGINT | pool_lsn_gap_clns - Limpiadores de espacio de anotación cronológica de agrupación de almacenamientos intermedios activados |
| POOL_DRTY_PG_STEAL_CLNS | BIGINT | pool_drt_y_pg_steal_clns - Limpiadores de páginas víctima de agrupación de agrupación de almacenamientos intermedios activados |
| POOL_DRTY_PG_THRSH_CLNS | BIGINT | pool_drt_y_pg_thrsh_clns - Limpiadores de umbral de agrupación de almacenamientos intermedios activados |
| PREFETCH_WAIT_TIME | BIGINT | prefetch_wait_time - Tiempo esperado para captación previa |
| UNREAD_PREFETCH_PAGES | BIGINT | unread_prefetch_pages - Páginas de captación previa no leídas |
| DIRECT_READS | BIGINT | direct_reads - Lecturas directas de la base de datos |
| DIRECT_WRITES | BIGINT | direct_writes - Grabaciones directas en la base de datos |
| DIRECT_READ_REQS | BIGINT | direct_read_reqs - Peticiones de lectura directa |
| DIRECT_WRITE_REQS | BIGINT | direct_write_reqs - Peticiones de grabación directa |
| DIRECT_READ_TIME | BIGINT | direct_read_time - Tiempo de lectura directa |
| DIRECT_WRITE_TIME | BIGINT | direct_write_time - Tiempo de grabación directa |
| FILES_CLOSED | BIGINT | files_closed - Archivos de base de datos cerrados |
| ELAPSED_EXEC_TIME_S | BIGINT | elapsed_exec_time - Tiempo transcurrido de ejecución de sentencia |
| ELAPSED_EXEC_TIME_MS | BIGINT | elapsed_exec_time - Tiempo transcurrido de ejecución de sentencia |
| COMMIT_SQL_STMTS | BIGINT | commit_sql_stmts - Sentencias de confirmación intentadas |
| ROLLBACK_SQL_STMTS | BIGINT | rollback_sql_stmts - Sentencias de retrotracción intentadas |
| DYNAMIC_SQL_STMTS | BIGINT | dynamic_sql_stmts - Sentencias de SQL dinámico intentadas |
| STATIC_SQL_STMTS | BIGINT | static_sql_stmts - Sentencias de SQL estático intentadas |
| FAILED_SQL_STMTS | BIGINT | failed_sql_stmts - Operaciones de sentencia anómalas |
| SELECT_SQL_STMTS | BIGINT | select_sql_stmts - Sentencias Select de SQL ejecutadas |
| UID_SQL_STMTS | BIGINT | uid_sql_stmts - Sentencias UPDATE/INSERT/DELETE de SQL ejecutadas |

Tabla 142. Información devuelta por la vista administrativa SNAPDB y la función de tabla SNAP_GET_DB_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------|---------------|---|
| DDL_SQL_STMTS | BIGINT | ddl_sql_stmts - Sentencias de lenguaje de definición de datos (DDL) de SQL |
| INT_AUTO_REBINDS | BIGINT | int_auto_rebinds - Revinculaciones automáticas internas |
| INT_ROWS_DELETED | BIGINT | int_rows_deleted - Filas internas suprimidas |
| INT_ROWS_INSERTED | BIGINT | int_rows_inserted - Filas internas insertadas |
| INT_ROWS_UPDATED | BIGINT | int_rows_updated - Filas internas actualizadas |
| INT_COMMITS | BIGINT | int_commits - Confirmaciones internas |
| INT_ROLLBACKS | BIGINT | int_rollback - Retrotracciones internas |
| INT_DEADLOCK_ROLLBACKS | BIGINT | int_deadlock_rollback - Retrotracciones internas debido a punto muerto |
| ROWS_DELETED | BIGINT | rows_deleted - Filas suprimidas |
| ROWS_INSERTED | BIGINT | rows_inserted - Filas insertadas |
| ROWS_UPDATED | BIGINT | rows_updated - Filas actualizadas |
| ROWS_SELECTED | BIGINT | rows_selected - Filas seleccionadas |
| ROWS_READ | BIGINT | rows_read - Filas leídas |
| BINDS_PRECOMPILES | BIGINT | binds_precompiles - Vinculaciones/precompilaciones intentadas |
| TOTAL_LOG_AVAILABLE | BIGINT | total_log_available - Anotaciones cronológicas totales disponibles |
| TOTAL_LOG_USED | BIGINT | total_log_used - Espacio total de anotaciones cronológicas utilizado |
| SEC_LOG_USED_TOP | BIGINT | sec_log_used_top - Espacio máximo de anotaciones cronológicas secundarias utilizado |
| TOT_LOG_USED_TOP | BIGINT | tot_log_used_top - Espacio máximo de anotaciones cronológicas totales utilizado |
| SEC_LOGS_ALLOCATED | BIGINT | sec_logs_allocated - Anotaciones cronológicas secundarias asignadas actualmente |
| LOG_READS | BIGINT | log_reads - Número de páginas de anotaciones cronológicas leídas |
| LOG_READ_TIME_S | BIGINT | log_read_time - Tiempo de lectura de anotaciones cronológicas |
| LOG_READ_TIME_NS | BIGINT | log_read_time - Tiempo de lectura de anotaciones cronológicas |
| LOG_WRITES | BIGINT | log_writes - Número de páginas de anotaciones cronológicas escritas |
| LOG_WRITE_TIME_S | BIGINT | log_write_time - Tiempo de grabación de anotaciones cronológicas |
| LOG_WRITE_TIME_NS | BIGINT | log_write_time - Tiempo de grabación de anotaciones cronológicas |
| NUM_LOG_WRITE_IO | BIGINT | num_log_write_io - Número de grabaciones de anotaciones cronológicas |

Tabla 142. Información devuelta por la vista administrativa SNAPDB y la función de tabla SNAP_GET_DB_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------------|---------------|--|
| NUM_LOG_READ_IO | BIGINT | num_log_read_io - Número de lecturas de anotaciones cronológicas |
| NUM_LOG_PART_PAGE_IO | BIGINT | num_log_part_page_io - Número de grabaciones de páginas de anotaciones cronológicas parciales |
| NUM_LOG_BUFFER_FULL | BIGINT | num_log_buffer_full - Número de almacenamientos intermedios de anotaciones cronológicas llenos |
| NUM_LOG_DATA_FOUND_IN_BUFFER | BIGINT | num_log_data_found_in_buffer - Número de datos de anotaciones cronológicas encontrados en el almacenamiento intermedio |
| APPL_ID_OLDEST_XACT | BIGINT | appl_id_oldest_xact - Aplicación con la transacción más antigua |
| LOG_TO_REDO_FOR_RECOVERY | BIGINT | log_to_redo_for_recovery - Cantidad de anotaciones cronológicas a rehacer para la recuperación |
| LOG_HELD_BY_DIRTY_PAGES | BIGINT | log_held_by_dirty_pages - Cantidad de espacio de anotaciones cronológicas contabilizado por páginas sucias |
| PKG_CACHE_LOOKUPS | BIGINT | pkg_cache_lookups - Búsquedas de antememoria de paquete |
| PKG_CACHE_INSERTS | BIGINT | pkg_cache_inserts - Inserciones de antememoria de paquete |
| PKG_CACHE_NUM_OVERFLOWS | BIGINT | pkg_cache_num_overflows - Desbordamientos de antememoria de paquete |
| PKG_CACHE_SIZE_TOP | BIGINT | pkg_cache_size_top - Marca de límite superior de antememoria de paquete |
| APPL_SECTION_LOOKUPS | BIGINT | appl_section_lookups - Búsquedas de sección |
| APPL_SECTION_INSERTS | BIGINT | appl_section_inserts - Inserciones de sección |
| CAT_CACHE_LOOKUPS | BIGINT | cat_cache_lookups - Búsquedas de antememoria de catálogo |
| CAT_CACHE_INSERTS | BIGINT | cat_cache_inserts - Inserciones de antememoria de catálogo |
| CAT_CACHE_OVERFLOWS | BIGINT | cat_cache_overflows - Desbordamientos de antememoria de catálogo |
| CAT_CACHE_SIZE_TOP | BIGINT | cat_cache_size_top - Marca de límite superior de antememoria de catálogo |
| PRIV_WORKSPACE_SIZE_TOP | BIGINT | priv_workspace_size_top - Tamaño máximo de espacio de trabajo privado |
| PRIV_WORKSPACE_NUM_OVERFLOWS | BIGINT | priv_workspace_num_overflows - Desbordamientos de espacio de trabajo privado |

Tabla 142. Información devuelta por la vista administrativa SNAPDB y la función de tabla SNAP_GET_DB_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------------------|---------------|--|
| PRIV_WORKSPACE_SECTION_INSERTS | BIGINT | priv_workspace_section_inserts - Inserciones de sección de espacio de trabajo privado |
| PRIV_WORKSPACE_SECTION_LOOKUPS | BIGINT | priv_workspace_section_lookups - Búsquedas de sección de espacio de trabajo privado |
| SHR_WORKSPACE_SIZE_TOP | BIGINT | shr_workspace_size_top - Tamaño máximo de espacio de trabajo compartido |
| SHR_WORKSPACE_NUM_OVERFLOWS | BIGINT | shr_workspace_num_overflows - Desbordamientos de espacio de trabajo compartido |
| SHR_WORKSPACE_SECTION_INSERTS | BIGINT | shr_workspace_section_inserts - Inserciones de sección de espacio de trabajo compartido |
| SHR_WORKSPACE_SECTION_LOOKUPS | BIGINT | shr_workspace_section_lookups - Búsquedas de sección de espacio de trabajo compartido |
| TOTAL_HASH_JOINS | BIGINT | total_hash_joins - Total de uniones hash |
| TOTAL_HASH_LOOPS | BIGINT | total_hash_loops - Total de bucles hash |
| HASH_JOIN_OVERFLOWS | BIGINT | hash_join_overflows - Desbordamientos de uniones hash |
| HASH_JOIN_SMALL_OVERFLOWS | BIGINT | hash_join_small_overflows - Desbordamientos pequeños de uniones hash |
| POST_SHRTHRESHOLD_HASH_JOINS | BIGINT | post_shrthreshold_hash_joins - Uniones hash de umbral posteriores |
| ACTIVE_HASH_JOINS | BIGINT | active_hash_joins - Uniones hash activas |
| NUM_DB_STORAGE_PATHS | BIGINT | num_db_storage_paths - Número de vías de acceso de almacenamiento automático |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |
| SMALLEST_LOG_AVAIL_NODE | INTEGER | smallest_log_avail_node - Nodo con menos espacio de anotaciones cronológicas disponible |
| TOTAL_OLAP_FUNCS | BIGINT | Número total de funciones OLAP ejecutadas. |
| OLAP_FUNC_OVERFLOWS | BIGINT | Número de veces que los datos de función OLAP han superado el espacio de pila de clasificación disponible. |
| ACTIVE_OLAP_FUNCS | BIGINT | Número total de funciones OLAP que se están ejecutando actualmente y que están consumiendo memoria de pila de clasificación. |
| STATS_CACHE_SIZE | BIGINT | Tamaño de antememoria de estadísticas en bytes |

Tabla 142. Información devuelta por la vista administrativa SNAPDB y la función de tabla SNAP_GET_DB_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------------|---------------|--|
| STATS_FABRICATIONS | BIGINT | Número total de actividades de recopilación de estadísticas para que el sistema cree estadísticas sin exploración de tabla ni de índice. |
| SYNC_RUNSTATS | BIGINT | Número total de actividades de recopilación de estadísticas síncronas durante compilación de consultas. |
| ASYNC_RUNSTATS | BIGINT | Cambiaremos la salida de esta columna por el número total de actividades asíncronas satisfactorias de recopilación de estadísticas. |
| STATS_FABRICATE_TIME | BIGINT | Tiempo total que el sistema ha empleado en crear estadísticas sin exploración de tabla o índice durante la compilación de consultas en milisegundos. |
| SYNC_RUNSTATS_TIME | BIGINT | Tiempo total empleado en actividades síncronas de recopilación de estadísticas en milisegundos. |
| NUM_THRESHOLD_VIOLATIONS | BIGINT | Número de violaciones de umbral sucedidas en la base de datos. |

Vista administrativa SNAPDB_MEMORY_POOL y función de tabla SNAP_GET_DB_MEMORY_POOL – Recuperar información sobre el uso de la memoria en el nivel de la base de datos

La vista administrativa SNAPDB_MEMORY_POOL y la función de tabla SNAP_GET_DB_MEMORY_POOL devuelven información acerca del uso de la memoria en el nivel de la base de datos únicamente para plataformas de UNIX.

Vista administrativa SNAPDB_MEMORY_POOL

Esta vista administrativa permite recuperar la información sobre el uso de la memoria en el nivel de la base de datos para la base de datos conectada actualmente.

La vista administrativa SNAPDB_MEMORY_POOL, utilizada con las vistas administrativas SNAPDB, SNAPDETAILLOG, SNAPHADR y SNAPSTORAGE_PATHS, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR DATABASE ON alias de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 143 en la página 570 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPDB_MEMORY_POOL

- Privilegio CONTROL en la vista administrativa SNAPDB_MEMORY_POOL
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_DB_MEMORY_POOL
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMAINT
- SYSADM

Ejemplo

Recuperar una lista de agrupaciones de memoria y su tamaño actual para la base de datos conectada actualmente. SAMPLE.

```
SELECT POOL_ID,
POOL_CUR_SIZE FROM SYSIBMADM.SNAPDB_MEMORY_POOL
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| POOL_ID | POOL_CUR_SIZE |
|---------------|---------------|
| UTILITY | 32768 |
| PACKAGE_CACHE | 475136 |
| CAT_CACHE | 65536 |
| BP | 2097152 |
| BP | 1081344 |
| BP | 540672 |
| BP | 278528 |
| BP | 147456 |
| BP | 81920 |
| LOCK_MGR | 294912 |
| DATABASE | 3833856 |
| OTHER | 0 |

12 registro(s) seleccionado(s).

SNAP_GET_DB_MEMORY_POOL, función de tabla

La función de tabla SNAP_GET_DB_MEMORY_POOL devuelve la misma información que la vista administrativa SNAPDB_MEMORY_POOL, pero permite recuperar la información para una base de datos concreta en una partición específica de la base de datos, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_DB_MEMORY_POOL, utilizada con las funciones de tabla SNAP_GET_DB_V95, SNAP_GET_DETAILLOG_V91, SNAP_GET_HADR y SNAP_GET_STORAGE_PATHS, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR ALL DATABASES.

Consulte la Tabla 143 en la página 570 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
▶▶ SNAP_GET_DB_MEMORY_POOL ( ( nombrebd [ , númparticiónbd ] ) ) ▶▶
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique una serie vacía para tomar la instantánea desde la base de datos conectada actualmente. Especifique un valor NULL para tomar la instantánea desde todas las bases de datos que se encuentran en la misma instancia que la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_DB_MEMORY_POOL toma una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_DB_MEMORY_POOL
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar una lista de agrupaciones de memoria y su tamaño actual para todas las bases de datos.

```
SELECT SUBSTR(DB_NAME,1,8) AS DB_NAME, POOL_ID, POOL_CUR_SIZE
FROM TABLE(SNAPSHOT_GET_DB_MEMORY_POOL
(CAST(NULL AS VARCHAR(128)), -1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| DB_NAME | POOL_ID | POOL_CUR_SIZE |
|---------|---------------|---------------|
| TESTDB | UTILITY | 65536 |
| TESTDB | PACKAGE_CACHE | 851968 |
| TESTDB | CAT_CACHE | 65536 |
| TESTDB | BP | 35913728 |
| TESTDB | BP | 589824 |
| TESTDB | BP | 327680 |
| TESTDB | BP | 196608 |
| TESTDB | BP | 131072 |
| TESTDB | SHARED_SORT | 65536 |
| TESTDB | LOCK_MGR | 10092544 |
| TESTDB | DATABASE | 4980736 |
| TESTDB | OTHER | 196608 |
| SAMPLE | UTILITY | 65536 |
| SAMPLE | PACKAGE_CACHE | 655360 |
| SAMPLE | CAT_CACHE | 131072 |
| SAMPLE | BP | 4325376 |
| SAMPLE | BP | 589824 |
| SAMPLE | BP | 327680 |
| SAMPLE | BP | 196608 |
| SAMPLE | BP | 131072 |
| SAMPLE | SHARED_SORT | 0 |
| SAMPLE | LOCK_MGR | 655360 |
| SAMPLE | DATABASE | 4653056 |
| SAMPLE | OTHER | 196608 |

24 registro(s) seleccionado(s).

Información devuelta

Tabla 143. Información devuelta por la vista administrativa `SNAPDB_MEMORY_POOL` y la función de tabla `SNAP_GET_DB_MEMORY_POOL`

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |

Tabla 143. Información devuelta por la vista administrativa SNAPDB_MEMORY_POOL y la función de tabla SNAP_GET_DB_MEMORY_POOL (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|--|
| POOL_ID | VARCHAR(14) | pool_id - Identificador de agrupación de memoria. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • APP_GROUP • APPL_CONTROL • APPLICATION • BP • CAT_CACHE • DATABASE • DFM • FCMBP • IMPORT_POOL • LOCK_MGR • MONITOR • OTHER • PACKAGE_CACHE • QUERY • SHARED_SORT • SORT • STATEMENT • STATISTICS • UTILITY |
| POOL_SECONDARY_ID | VARCHAR(32) | pool_secondary_id - Identificador secundario de agrupación de memoria |
| POOL_CUR_SIZE | BIGINT | pool_cur_size - Tamaño actual de la agrupación de memoria |
| POOL_WATERMARK | BIGINT | pool_watermark - Marca límite de agrupación de memoria |
| POOL_CONFIG_SIZE | BIGINT | pool_config_size - Tamaño configurado de la agrupación de memoria |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa SNAPDBM y función de tabla SNAP_GET_DBM_V95 – Recuperar la información de instantáneas del grupo de datos lógicos dbm

La vista administrativa SNAPDBM y la función de tabla SNAP_GET_DBM_V95 devuelven información acerca de los grupos lógicos del gestor de bases de datos DB2 (dbm) y del supervisor de instantáneas.

Vista administrativa SNAPDBM

La vista administrativa SNAPDBM, utilizada con las vistas administrativas SNAPDBM_MEMORY_POOL, SNAPFCM, SNAPFCM_PART y SNAPSWITCHES, ofrece los datos equivalentes al mandato GET SNAPSHOT FOR DBM.

El esquema es SYSIBMADM.

Consulte la Tabla 144 en la página 574 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPDBM
- Privilegio CONTROL en la vista administrativa SNAPDBM
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_DBM_V95
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar estado del gestor de bases de datos e información sobre la conexión para todas las particiones de base de datos.

```
SELECT DB2_STATUS, DB2START_TIME, LAST_RESET, LOCAL_CONS, REM_CONS_IN,  
       (AGENTS_CREATED_EMPTY_POOL/AGENTS_FROM_POOL) AS AGENT_USAGE,  
       DBPARTITIONNUM FROM SYSIBMADM.SNAPDBM ORDER BY DBPARTITIONNUM
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| DB2_STATUS | DB2START_TIME | LAST_RESET | ... |
|------------|----------------------------|------------|-----|
| ACTIVE | 2006-01-06-14.59.59.059879 | - | ... |
| ACTIVE | 2006-01-06-14.59.59.097605 | - | ... |
| ACTIVE | 2006-01-06-14.59.59.062798 | - | ... |

3 registro(s) seleccionado(s). ...

Datos de salida de esta consulta (continuación).

| ... | LOCAL_CONS | REM_CONS_IN | AGENT_USAGE | DBPARTITIONNUM |
|-----|------------|-------------|-------------|----------------|
| ... | 1 | 1 | 0 | 0 |
| ... | 0 | 0 | 0 | 1 |
| ... | 0 | 0 | 0 | 2 |

SNAP_GET_DBM_V95, función de tabla

La función de tabla SNAP_GET_DBM_V95 devuelve la misma información que la vista administrativa SNAPDBM, pero le permite recuperar la información para una partición de base de datos específica, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_DBM_V95, utilizada con las funciones de tabla SNAP_GET_DBM_MEMORY_POOL, SNAP_GET_FCM, SNAP_GET_FCM_PART y SNAP_GET_SWITCHES, ofrece los datos equivalentes al mandato GET SNAPSHOT FOR DBM.

Consulte la Tabla 144 en la página 574 para ver una lista completa de la información que se puede devolver.

Sintaxis

```

▶▶ SNAP_GET_DBM_V95 ( [ númparticiónbd ] )

```

El esquema es SYSPROC.

Parámetro de la función de tabla

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si no se utiliza esta opción de entrada, se devolverán datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *núm_partición_bd* se establece en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_DBM_V95 toma la instantánea de la memoria.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_DBM_V95
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar la hora de inicio y el estado actual del número 2 de partición de base de datos.

```
SELECT DB2START_TIME, DB2_STATUS FROM TABLE(SNAP_GET_DBM_V95(2)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
DB2START_TIME          DB2_STATUS
-----
2006-01-06-14.59.59.062798 ACTIVE
```

Información devuelta

Tabla 144. Información que devuelve la vista administrativa SNAPDBM y la función de tabla SNAP_GET_DBM_V95

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| SORT_HEAP_ALLOCATED | BIGINT | sort_heap_allocated - Total de pila de clasificación asignada |
| POST_THRESHOLD_SORTS | BIGINT | post_threshold_sorts - Clasificaciones de umbral posteriores |
| PIPED_SORTS_REQUESTED | BIGINT | piped_sorts_requested - Clasificaciones mediante conexiones solicitadas |
| PIPED_SORTS_ACCEPTED | BIGINT | piped_sorts_accepted - Clasificaciones mediante conexiones aceptadas |
| REM_CONS_IN | BIGINT | rem_cons_in - Conexiones remotas con gestor de bases de datos |
| REM_CONS_IN_EXEC | BIGINT | rem_cons_in_exec - Conexiones remotas que se ejecutan en el elemento de supervisor de gestor de bases de datos |
| LOCAL_CONS | BIGINT | local_cons - Conexiones locales |
| LOCAL_CONS_IN_EXEC | BIGINT | local_cons_in_exec - Conexiones locales que se ejecutan en el elemento de supervisor de gestor de bases de datos |
| CON_LOCAL_DBASES | BIGINT | con_local_dbases - Bases de datos locales con conexiones actuales |
| AGENTS_REGISTERED | BIGINT | agents_registered - Agentes registrados |
| AGENTS_WAITING_ON_TOKEN | BIGINT | agents_waiting_on_token - Agentes en espera de una señal |

Tabla 144. Información que devuelve la vista administrativa SNAPDBM y la función de tabla SNAP_GET_DBM_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------------|---------------|--|
| DB2_STATUS | VARCHAR(12) | db2_status - Estado de instancia de DB2 Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • ACTIVE • QUIESCE_PEND • QUIESCED |
| AGENTS_REGISTERED_TOP | BIGINT | agents_registered_top - Número máximo de agentes registrados |
| AGENTS_WAITING_TOP | BIGINT | agents_waiting_top - Número máximo de agentes en espera |
| COMM_PRIVATE_MEM | BIGINT | comm_private_mem - Memoria privada confirmada |
| IDLE_AGENTS | BIGINT | idle_agents - Número de agentes desocupados |
| AGENTS_FROM_POOL | BIGINT | agents_from_pool - Agentes asignados de la agrupación |
| AGENTS_CREATED_EMPTY_POOL | BIGINT | agents_created_empty_pool - Agentes creados porque la agrupación de agentes está vacía |
| COORD_AGENTS_TOP | BIGINT | coord_agents_top - Número máximo de agentes de coordinación |
| MAX_AGENT_OVERFLOW | BIGINT | max_agent_overflows - Número máximo de desbordamientos de agente |
| AGENTS_STOLEN | BIGINT | agents_stolen - Agentes robados |
| GW_TOTAL_CONS | BIGINT | gw_total_cons - Número total de conexiones intentadas para DB2 Connect |
| GW_CUR_CONS | BIGINT | gw_cur_cons - Número actual de conexiones para DB2 Connect |
| GW_CONS_WAIT_HOST | BIGINT | gw_cons_wait_host - Número de conexiones en espera para que conteste el sistema principal |
| GW_CONS_WAIT_CLIENT | BIGINT | gw_cons_wait_client - Número de conexiones en espera para que el cliente envíe petición |
| POST_THRESHOLD_HASH_JOINS | BIGINT | post_threshold_hash_joins - Umbral de uniones hash |
| NUM_GW_CONN_SWITCHES | BIGINT | num_gw_conn_switches - Conmutadores de conexión |
| DB2START_TIME | TIMESTAMP | db2start_time - Indicación de fecha y hora de inicio de gestor de bases de datos |
| LAST_RESET | TIMESTAMP | last_reset - Indicación de fecha y hora de último restablecimiento |
| NUM_NODES_IN_DB2_INSTANCE | INTEGER | num_nodes_in_db2_instance - Número de nodos en la partición de base de datos |

Tabla 144. Información que devuelve la vista administrativa SNAPDBM y la función de tabla SNAP_GET_DBM_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------------|---------------|--|
| PRODUCT_NAME | VARCHAR(32) | product_name - Nombre del producto |
| SERVICE_LEVEL | VARCHAR(18) | service_level - Nivel de servicio |
| SORT_HEAP_TOP | BIGINT | sort_heap_top - Marca de límite superior de pila privada de clasificación |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |
| POST_THRESHOLD_OLAP_FUNCS | BIGINT | <p>Número de funciones OLAP que han solicitado una pila de clasificación una vez superado el umbral de la pila de clasificación.</p> <p>Las clasificaciones, las uniones de generación aleatoria y las funciones OLAP son ejemplos de operaciones que utilizan una pila de clasificación. Bajo condiciones normales, el gestor de bases de dato asignará pila de clasificación utilizando el valor especificado por el parámetro de configuración sortheap. Si la cantidad de memoria asignada a las pilas de clasificación supera el umbral de pila de clasificación (parámetro de configuración sheapthres), el gestor de bases de datos asignará las siguientes pilas de clasificación utilizando un valor menor que el especificado por el parámetro de configuración sortheap.</p> <p>Es posible que las funciones OLAP que empiezan después de que se alcance el umbral de pila de clasificación no reciban la cantidad óptima de memoria para su ejecución.</p> |

Vista administrativa SNAPDBM_MEMORY_POOL y función de tabla SNAP_GET_DBM_MEMORY_POOL – Recuperar información sobre el uso de la memoria en el nivel del gestor de bases de datos

La vista administrativa SNAPDBM_MEMORY_POOL y la tabla de función SNAP_GET_DBM_MEMORY_POOL devuelven información acerca del uso de la memoria en el gestor de bases de datos.

Vista administrativa SNAPDBM_MEMORY_POOL

La vista administrativa SNAPDBM_MEMORY_POOL, utilizada con las vistas administrativas SNAPDBM, SNAPFCM, SNAPFCM_PART y SNAPSWITCHES, ofrece los datos equivalentes al mandato GET SNAPSHOT FOR DBM.

El esquema es SYSIBMADM.

Consulte la Tabla 145 en la página 579 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPDBM_MEMORY_POOL
- Privilegio CONTROL en la vista administrativa SNAPDBM_MEMORY_POOL
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE en la función de tabla SNAP_GET_DBM_MEMORY_POOL
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMAINT
- SYSADM

Ejemplo

Recuperar una lista de agrupaciones de memoria y su tamaño actual para el gestor de bases de datos de la base de datos conectada.

```
SELECT POOL_ID, POOL_CUR_SIZE FROM SNAPDBM_MEMORY_POOL
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
POOL_ID          POOL_CUR_SIZE
-----
MONITOR                    65536
OTHER                    29622272
FCMBP                    57606144
...
```

SNAP_GET_DBM_MEMORY_POOL, función de tabla

La función de tabla SNAP_GET_DBM_MEMORY_POOL devuelve la misma información que la vista administrativa SNAPDBM_MEMORY_POOL, pero permite recuperar la información de una partición específica de la base de datos, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_DBM_MEMORY_POOL, utilizada con las funciones de tabla SNAP_GET_DBM_V95, SNAP_GET_FCM, SNAP_GET_FCM_PART y SNAP_GET_SWITCHES, ofrece los datos equivalentes al mandato GET SNAPSHOT FOR DBM.

Consulte la Tabla 145 en la página 579 para ver una lista completa de la información que se puede devolver.

Sintaxis

► SNAP_GET_DBM_MEMORY_POOL (núm-partición-bd) ◄

El esquema es SYSPROC.

Parámetro de la función de tabla

núm-partición-bd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si no se utiliza esta opción de entrada, se devolverán datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *núm-partición-bd* se establece en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_DBM_MEMORY_POOL toma una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE en la función de tabla SNAP_GET_DBM_MEMORY_POOL
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar una lista de agrupaciones de memoria y su tamaño actual para todas las particiones de base de datos del gestor de bases de datos, de la base de datos conectada.

```
SELECT POOL_ID, POOL_CUR_SIZE, DBPARTITIONNUM  
FROM TABLE(SYSPROC.SNAP_GET_DBM_MEMORY_POOL())  
AS T ORDER BY DBPARTITIONNUM
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| POOL_ID | POOL_CUR_SIZE | DBPARTITIONNUM |
|---------|---------------|----------------|
| MONITOR | 65536 | 0 |
| OTHER | 29622272 | 0 |
| FCMBP | 57606144 | 0 |
| MONITOR | 65536 | 1 |

| | | |
|---------|----------|---|
| OTHER | 29425664 | 1 |
| FCMBP | 57606144 | 1 |
| MONITOR | 65536 | 2 |
| OTHER | 29425664 | 2 |
| FCMBP | 57606144 | 2 |

Información devuelta

Tabla 145. Información que devuelve la vista administrativa SNAPDBM_MEMORY_POOL y la función de tabla SNAP_GET_DBM_MEMORY_POOL

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| POOL_ID | VARCHAR(14) | pool_id - Identificador de agrupación de memoria. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • APP_GROUP • APPL_CONTROL • APPLICATION • BP • CAT_CACHE • DATABASE • DFM • FCMBP • IMPORT_POOL • LOCK_MGR • MONITOR • OTHER • PACKAGE_CACHE • QUERY • SHARED_SORT • SORT • STATEMENT • STATISTICS • UTILITY |
| POOL_CUR_SIZE | BIGINT | pool_cur_size - Tamaño actual de la agrupación de memoria |
| POOL_WATERMARK | BIGINT | pool_watermark - Marca límite de agrupación de memoria |
| POOL_CONFIG_SIZE | BIGINT | pool_config_size - Tamaño configurado de la agrupación de memoria |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa SNAPDETAILLOG y función de tabla SNAP_GET_DETAILLOG_V91 - Recuperar información de instantánea del grupo de datos lógicos detail_log

La vista administrativa SNAPDETAILLOG y la función de tabla SNAP_GET_DETAILLOG_V91 devuelven información de la instantánea del grupo de datos lógicos detail_log.

SNAPDETAILLOG, vista administrativa

Esta vista administrativa le permite recuperar información de instantánea desde el grupo de datos lógicos detail_log para la base de datos conectada actualmente.

La vista administrativa SNAPDETAILLOG, que se utiliza junto con las vistas administrativas SNAPDB, SNAPDB_MEMORY_POOL, SNAPHDR y SNAPSTORAGE_PATHS, proporciona información equivalente a la que proporciona el mandato CLP GET SNAPSHOT FOR DATABASE ON alias de base de datos.

El esquema es SYSIBMADM.

En la Tabla 146 en la página 582 encontrará una lista completa de la información que se devuelve.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPDETAILLOG
- Privilegio CONTROL en la vista administrativa SNAPDETAILLOG
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_DETAILLOG_V91
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar información acerca de las anotaciones cronológicas para todas las particiones de base de datos para la base de datos conectada actualmente.

```
SELECT SUBSTR(DB_NAME, 1, 8) AS DB_NAME, FIRST_ACTIVE_LOG,  
       LAST_ACTIVE_LOG, CURRENT_ACTIVE_LOG, CURRENT_ARCHIVE_LOG,  
       DBPARTITIONNUM  
FROM SYSIBMADM.SNAPDETAILLOG ORDER BY DBPARTITIONNUM
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| DB_NAME | FIRST_ACTIVE_LOG | LAST_ACTIVE_LOG | ... |
|---------|------------------|-----------------|-----|
| TEST | 0 | 8 | ... |
| TEST | 0 | 8 | ... |
| TEST | 0 | 8 | ... |

3 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

| ... | CURRENT_ACTIVE_LOG | CURRENT_ARCHIVE_LOG | DBPARTITIONNUM |
|-----|--------------------|---------------------|----------------|
| ... | 0 | - | 0 |
| ... | 0 | - | 1 |
| ... | 0 | - | 2 |

Función de tabla SNAP_GET_DETAILLOG_V91

La función de tabla SNAP_GET_DETAILLOG_V91 devuelve la misma información que la vista administrativa SNAPDETAILLOG.

La función de tabla SNAP_GET_DETAILLOG, que se utiliza junto con las funciones de tabla SNAP_GET_DB_V95, SNAP_GET_DB_MEMORY_POOL, SNAP_GET_HADR y SNAP_GET_STORAGE_PATHS, proporciona información que es equivalente a la que proporciona el mandato CLP GET SNAPSHOT FOR ALL DATABASES.

En la Tabla 146 en la página 582 encontrará una lista completa de la información que se devuelve.

Sintaxis

```

▶▶ SNAP_GET_DETAILLOG_V91 (—nombrebd —————)
                                     |
                                     |, númparticiónbd
▶▶

```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique una serie vacía para tomar la instantánea desde la base de datos conectada actualmente. Especifique un valor NULL para tomar la instantánea desde todas las bases de datos que se encuentran en la misma instancia que la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_DETAILLOG_V91 tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_DETAILLOG_V91
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar información acerca de las anotaciones cronológicas para la partición de base de datos 1 para la base de datos conectada actualmente.

```
SELECT SUBSTR(DB_NAME, 1, 8) AS DB_NAME, FIRST_ACTIVE_LOG,
       LAST_ACTIVE_LOG, CURRENT_ACTIVE_LOG, CURRENT_ARCHIVE_LOG
FROM TABLE(SNAP_GET_DETAILLOG_V91(' ', 1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
DB_NAME FIRST_ACTIVE_LOG LAST_ACTIVE_LOG ...
-----
TEST          0          8 ...
```

1 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```
... CURRENT_ACTIVE_LOG CURRENT_ARCHIVE_LOG
... -----
...          0          -
```

Metadatos de la vista administrativa SNAPDETAILLOG y la función de tabla SNAP_GET_DETAILLOG_V91

Tabla 146. Información devuelta por la vista administrativa SNAPDETAILLOG y la función de tabla SNAP_GET_DETAILLOG_V91

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |

Tabla 146. Información devuelta por la vista administrativa SNAPDETAILLOG y la función de tabla SNAP_GET_DETAILLOG_V91 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------|---------------|---|
| FIRST_ACTIVE_LOG | BIGINT | first_active_log - Primer número de archivo de anotaciones cronológicas activo |
| LAST_ACTIVE_LOG | BIGINT | last_active_log - Último número de archivo de anotaciones cronológicas activo |
| CURRENT_ACTIVE_LOG | BIGINT | current_active_log - Número de archivo de anotaciones cronológicas activo actual |
| CURRENT_ARCHIVE_LOG | BIGINT | current_archive_log - Número de archivo de anotaciones cronológicas de archivado actual |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa SNAPDYN_SQL y función de tabla SNAP_GET_DYN_SQL_V95 – Recuperar información de instantáneas del grupo de datos lógicos dynsql

La “SNAPDYN_SQL, vista administrativa” y la “Función de tabla SNAP_GET_DYN_SQL_V95” en la página 584 devuelven información de instantánea desde el grupo de datos lógicos dynsql.

SNAPDYN_SQL, vista administrativa

Esta vista administrativa permite recuperar la información de instantáneas del grupo lógico dynsql para la base de datos conectada actualmente.

Esta vista proporciona información equivalente a la que proporciona el mandato CLP GET SNAPSHOT FOR DYNAMIC SQL ON alias de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 147 en la página 587 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPDYN_SQL
- Privilegio CONTROL en la vista administrativa SNAPDYN_SQL
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_DYN_SQL_V95
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMMAINT
- SYSADM

Ejemplo

Recuperar una lista de sentencias de SQL dinámico ejecutadas en todas las particiones de base de datos, ordenadas por el número de filas leídas.

```
SELECT PREP_TIME_WORST, NUM_COMPILATIONS, SUBSTR(STMT_TEXT, 1, 60)
AS STMT_TEXT, DBPARTITIONNUM
FROM SYSIBMADM.SNAPDYN_SQL ORDER BY ROWS_READ
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| PREP_TIME_WORST | NUM_COMPILATIONS | ... |
|-----------------|------------------|-----|
| 98 | 1 | ... |
| 9 | 1 | ... |
| 0 | 0 | ... |
| 0 | 1 | ... |
| 0 | 1 | ... |
| 0 | 1 | ... |
| 0 | 1 | ... |
| 0 | 1 | ... |
| 0 | 1 | ... |
| 40 | 1 | ... |

9 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```
... STMT_TEXT ...
... ----- ...
... select prep_time_worst, num_compilations, substr(stmt_text, ...
... select * from dbuser.employee ...
... SET CURRENT LOCALE LC_CTYPE = 'en_US' ...
... select prep_time_worst, num_compilations, substr(stmt_text, ...
... select prep_time_worst, num_compilations, substr(stmt_text, ...
... select * from dbuser.employee ...
... insert into dbuser.employee values(1) ...
... select * from dbuser.employee ...
... insert into dbuser.employee values(1) ...
```

Datos de salida de esta consulta (continuación).

```
... DBPARTITIONNUM
... -----
... 0
... 0
... 0
... 2
... 1
... 2
... 2
... 1
... 0
```

Función de tabla SNAP_GET_DYN_SQL_V95

La función de tabla SNAP_GET_DYN_SQL_V95 devuelve la misma información que la vista administrativa SNAPDYN_SQL, pero le permite recuperar la

información para una base de datos específica de una partición de base de datos específica, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

Esta función de tabla devuelve información equivalente a la que proporciona el mandato CLP GET SNAPSHOT FOR DYNAMIC SQL ON alias de base de datos.

Consulte la Tabla 147 en la página 587 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
▶▶ SNAP_GET_DYN_SQL_V95 ( ( nombrebd [ , numparticiónbd ] ) ) ▶▶
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique NULL o una serie vacía para tomar la instantánea de la base de datos conectada actualmente.

numparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *numparticiónbd* se establece en NULL, -1 se establece implícitamente para *numparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *numparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_DYN_SQL_V95 tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_DYN_SQL_V95
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON

- SYSCTRL
- SYSMANT
- SYSADM

Ejemplo

Recuperar una lista de sentencias de SQL dinámico ejecutadas en la partición de base de datos correspondiente a la base de datos conectada actualmente, ordenadas por el número de filas leídas.

```
SELECT PREP_TIME_WORST, NUM_COMPILATIONS, SUBSTR(STMT_TEXT, 1, 60)
AS STMT_TEXT FROM TABLE(SNAP_GET_DYN_SQL_V95('','-1)) as T
ORDER BY ROWS_READ
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
PREP_TIME_WORST      ...
-----
0 ...
3 ...
...
4 ...
...
4 ...
...
4 ...
...
3 ...
...
4 ...
...
```

Datos de salida de esta consulta (continuación).

```
... NUM_COMPILATIONS  STMT_TEXT
... -----
... 0 SET CURRENT LOCALE LC_CTYPE = 'en_US'
... 1 select rows_read, rows_written,
...      substr(stmt_text, 1, 40) as
... 1 select * from table
...      (snap_get_dyn_sqlv9('','-1)) as t
... 1 select * from table
...      (snap_getdetaillog9('','-1)) as t
... 1 select * from table
...      (snap_get_hadr('','-1)) as t
... 1 select prep_time_worst, num_compilations,
...      substr(stmt_text,
... 1 select prep_time_worst, num_compilations,
...      substr(stmt_text,
```

Después de ejecutar una carga de trabajo, el usuario puede utilizar la siguiente consulta con la función de tabla.

```
SELECT STATS_FABRICATE_TIME, SYNC_RUNSTATS_TIME
FROM TABLE (SNAP_GET_DYN_SQL_V95('mytestdb', -1))
AS SNAPDB
```

```
STATS_FABRICATE_TIME SYNC_RUNSTATS_TIME
-----
2 12
1 30
```

Para la vista basada en esta función de tabla:

```
SELECT STATS_FABRICATE_TIME,SYNC_RUNSTATS_TIME
FROM SYSIBMADM.SNAPDYN_SQL
```

```
STATS_FABRICATE_TIME SYNC_RUNSTATS_TIME
-----
                    5                10
                    3                20
```

2 registro(s) seleccionado(s).

Información devuelta

Tabla 147. Información devuelta por la vista administrativa SNAPDYN_SQL y la función de tabla SNAP_GET_DYN_SQL_V95

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| NUM_EXECUTIONS | BIGINT | num_executions - Ejecuciones de sentencia |
| NUM_COMPILATIONS | BIGINT | num_compilations - Compilaciones de sentencia |
| PREP_TIME_WORST | BIGINT | prep_time_worst - Peor tiempo de preparación de sentencia |
| PREP_TIME_BEST | BIGINT | prep_time_best - Mejor tiempo de preparación de sentencia |
| INT_ROWS_DELETED | BIGINT | int_rows_deleted - Filas internas suprimidas |
| INT_ROWS_INSERTED | BIGINT | int_rows_inserted - Filas internas insertadas |
| INT_ROWS_UPDATED | BIGINT | int_rows_updated - Filas internas actualizadas |
| ROWS_READ | BIGINT | rows_read - Filas leídas |
| ROWS_WRITTEN | BIGINT | rows_written - Filas grabadas |
| STMT_SORTS | BIGINT | stmt_sorts - Clasificaciones de sentencias |
| SORT_OVERFLOWS | BIGINT | sort_overflows - Desbordamientos de clasificación |
| TOTAL_SORT_TIME | BIGINT | total_sort_time - Tiempo de clasificación total |
| POOL_DATA_L_READS | BIGINT | pool_data_l_reads - Lecturas lógicas de datos de la agrupación de almacenamientos intermedios |
| POOL_DATA_P_READS | BIGINT | pool_data_p_reads - Lecturas físicas de datos de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_L_READS | BIGINT | pool_temp_data_l_reads - Lecturas lógicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_P_READS | BIGINT | pool_temp_data_p_reads - Lecturas físicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_INDEX_L_READS | BIGINT | pool_index_l_reads - Lecturas lógicas de índices de la agrupación de almacenamientos intermedios |
| POOL_INDEX_P_READS | BIGINT | pool_index_p_reads - Lecturas físicas de índices de la agrupación de almacenamientos intermedios |

Tabla 147. Información devuelta por la vista administrativa SNAPDYN_SQL y la función de tabla SNAP_GET_DYN_SQL_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|--|
| POOL_TEMP_INDEX_L_READS | BIGINT | pool_temp_index_l_reads - Lecturas lógicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_P_READS | BIGINT | pool_temp_index_p_reads - Lecturas físicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_XDA_L_READS | BIGINT | pool_xda_l_reads - Lecturas lógicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_XDA_P_READS | BIGINT | pool_xda_p_reads - Lecturas físicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_L_READS | BIGINT | pool_temp_xda_l_reads - Lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_P_READS | BIGINT | pool_temp_xda_p_reads - Elemento del supervisor de las lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |
| TOTAL_EXEC_TIME | BIGINT | total_exec_time - Tiempo transcurrido de ejecución de sentencia |
| TOTAL_EXEC_TIME_MS | BIGINT | total_exec_time - Tiempo transcurrido de ejecución de sentencia |
| TOTAL_USR_CPU_TIME | BIGINT | total_usr_cpu_time - Total de CPU de usuario para una sentencia |
| TOTAL_USR_CPU_TIME_MS | BIGINT | total_usr_cpu_time - Total de CPU de usuario para una sentencia |
| TOTAL_SYS_CPU_TIME | BIGINT | total_sys_cpu_time - Total de CPU de sistema para una sentencia |
| TOTAL_SYS_CPU_TIME_MS | BIGINT | total_sys_cpu_time - Total de CPU de sistema para una sentencia |
| STMT_TEXT | CLOB(2 M) | stmt_text - Texto de sentencia de SQL |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |
| STATS_FABRICATE_TIME | BIGINT | Tiempo total (en milisegundos) que ha empleado el sistema en crear las estadísticas necesarias sin exploración de tabla o de índice durante la compilación de consultas para una sentencia dinámica. |
| SYNC_RUNSTATS_TIME | BIGINT | Tiempo total (en milisegundos) empleado en actividades síncronas de recopilación de estadísticas durante la compilación de consultas para una sentencia dinámica. |

Vista administrativa SNAPFCM y función de tabla SNAP_GET_FCM – Recuperar la información de instantáneas del grupo de datos lógicos fcm

La vista administrativa SNAPFCM y la función de tabla SNAP_GET_FCM devuelven información acerca del gestor de comunicaciones rápidas (FCM) a partir de la instantánea de una base de datos, en concreto el grupo de datos lógicos fcm.

SNAPFCM, vista administrativa

La vista administrativa SNAPFCM, utilizada con las vistas administrativas SNAPDBM, SNAPDBM_MEMORY_POOL, SNAPFCM_PART y SNAPSWITCHES, ofrece los datos equivalentes al mandato GET SNAPSHOT FOR DBM.

El esquema es SYSIBMADM.

Consulte la Tabla 148 en la página 591 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPFCM
- Privilegio CONTROL en la vista administrativa SNAPFCM
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE para la función de tabla SNAP_GET_FCM
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar información sobre los almacenamientos intermedios de mensajes de Fast Communication Manager en todas las particiones de base de datos.

```
SELECT BUFF_FREE, BUFF_FREE_BOTTOM, DBPARTITIONNUM  
FROM SYSIBMADM.SNAPFCM ORDER BY DBPARTITIONNUM
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| BUFF_FREE | BUFF_FREE_BOTTOM | DBPARTITIONNUM |
|-----------|------------------|----------------|
| 5120 | 5100 | 0 |
| 5120 | 5100 | 1 |
| 5120 | 5100 | 2 |

SNAP_GET_FCM, función de tabla

La función de tabla SNAP_GET_FCM devuelve la misma información que la vista administrativa SNAPFCM, pero le permite recuperar la información para una partición de base de datos específica, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_FCM, utilizada con las funciones de tabla SNAP_GET_DBM_V95, SNAP_GET_DBM_MEMORY_POOL, SNAP_GET_FCM_PART y SNAP_GET_SWITCHES, ofrece los datos equivalentes al mandato GET SNAPSHOT FOR DBM.

Consulte la Tabla 148 en la página 591 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
▶▶ SNAP_GET_FCM ( [ núm-partición-bd ] ) ▶▶▶
```

El esquema es SYSPROC.

Parámetro de la función de tabla

núm-partición-bd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si no se utiliza esta opción de entrada, se devolverán datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *núm-partición-bd* se establece en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_FCM toma una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE para la función de tabla SNAP_GET_FCM
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar información sobre los almacenamientos intermedios de mensajes de Fast Communication Manager en la partición 1 de la base de datos.

```
SELECT BUFF_FREE, BUFF_FREE_BOTTOM, DBPARTITIONNUM  
FROM TABLE(SYSPROC.SNAP_GET_FCM( 1 )) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
BUFF_FREE          BUFF_FREE_BOTTOM    DBPARTITIONNUM  
-----  
                5120                5100                1
```

Información devuelta

Tabla 148. Información que devuelve la vista administrativa SNAPFCM y la función de tabla SNAP_GET_FCM

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| BUFF_FREE | BIGINT | buff_free - Almacenamientos intermedios FCM libres actualmente |
| BUFF_FREE_BOTTOM | BIGINT | buff_free_bottom - Almacenamientos intermedios FCM mínimos libres |
| CH_FREE | BIGINT | ch_free - Canales libres actualmente |
| CH_FREE_BOTTOM | BIGINT | ch_free_bottom - Número mínimo de canales libres |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa SNAPFCM_PART y función de tabla SNAP_GET_FCM_PART – Recuperar la información de instantáneas del grupo de datos lógicos fcm_node

La vista administrativa SNAPFCM_PART y la función de tabla SNAP_GET_FCM_PART devuelven información acerca del gestor de comunicaciones rápidas (FCM) a partir de la instantánea del gestor de bases de datos, en concreto el grupo de datos lógicos fcm_node.

SNAPFCM_PART, vista administrativa

La vista administrativa SNAPFCM_PART, utilizada con las vistas administrativas SNAPDBM, SNAPDBM_MEMORY_POOL, SNAPFCM y SNAPSWITCHES, ofrece los datos equivalentes al mandato GET SNAPSHOT FOR DBM.

El esquema es SYSIBMADM.

Consulte la Tabla 149 en la página 594 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPFCM_PART
- Privilegio CONTROL en la vista administrativa SNAPFCM_PART
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE en la función de tabla SNAP_GET_FCM_PART
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMMAINT
- SYSADM

Ejemplo

Recuperar información enviada y recibida de almacenamientos intermedios para el Fast Communication Manager.

```
SELECT CONNECTION_STATUS, TOTAL_BUFFERS_SENT, TOTAL_BUFFERS_RECEIVED
FROM SYSIBMADM.SNAPFCM_PART WHERE DBPARTITIONNUM = 0
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| CONNECTION_STATUS | TOTAL_BUFFERS_SENT | TOTAL_BUFFERS_RCVD |
|-------------------|--------------------|--------------------|
| INACTIVE | 2 | 1 |

1 registro(s) seleccionado(s).

SNAP_GET_FCM_PART, función de tabla

La función de tabla SNAP_GET_FCM_PART devuelve la misma información que la vista administrativa SNAPFCM_PART, pero le permite recuperar la información para una partición de base de datos específica, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_FCM_PART, utilizada con las funciones de tabla SNAP_GET_DBM_V95, SNAP_GET_DBM_MEMORY_POOL, SNAP_GET_FCM y SNAP_GET_SWITCHES, ofrece los datos equivalentes al mandato GET SNAPSHOT FOR DBM.

Consulte la Tabla 149 en la página 594 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
▶▶ SNAP_GET_FCM_PART ( [núm-partición-bd] ) ▶▶
```

El esquema es SYSPROC.

Parámetro de la función de tabla

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición actual o -2 para un agregado de todas las particiones de base de datos activas. Si no se utiliza esta opción de entrada, se devolverán datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *núm_partición_bd* se establece en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_FCM_PART toma una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE en la función de tabla SNAP_GET_FCM_PART
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar información enviada y recibida de almacenamientos intermedios para el Fast Communication Manager en todas las particiones de base de datos.

```
SELECT FCM_DBPARTITIONNUM, TOTAL_BUFFERS_SENT, TOTAL_BUFFERS_RCVD,  
       DBPARTITIONNUM FROM TABLE(SNAP_GET_FCM_PART()) AS T  
ORDER BY DBPARTITIONNUM
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| FCM_DBPARTITIONNUM | TOTAL_BUFFERS_SENT | TOTAL_BUFFERS_RCVD | DBPARTITIONNUM |
|--------------------|--------------------|--------------------|----------------|
| 0 | 305 | 305 | 0 |
| 1 | 5647 | 1664 | 0 |
| 2 | 5661 | 1688 | 0 |
| 0 | 19 | 19 | 1 |
| 1 | 305 | 301 | 1 |
| 2 | 1688 | 5661 | 1 |
| 0 | 1664 | 5647 | 2 |
| 1 | 10 | 10 | 2 |
| 2 | 301 | 305 | 2 |

Información devuelta

Tabla 149. Información que devuelven la vista administrativa SNAPFCM_PART y la función de tabla SNAP_GET_FCM_PART

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| CONNECTION_STATUS | VARCHAR(10) | connection_status - Estado de conexión. Esta interfaz devuelve un identificador de texto basándose en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none">• INACTIVE• ACTIVE• CONGESTED |
| TOTAL_BUFFERS_SENT | BIGINT | total_buffers_sent - Total de almacenamientos intermedios FCM enviados |
| TOTAL_BUFFERS_RCVD | BIGINT | total_buffers_rcvd - Total de almacenamientos intermedios FCM recibidos |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |
| FCM_DBPARTITIONNUM | SMALLINT | El número de la partición de base de datos a la que se han enviado o desde la que se han recibido los datos (según las columnas TOTAL_BUFFERS_SENT y TOTAL_BUFFERS_RCVD). |

Vista administrativa SNAPHADR y función de tabla SNAP_GET_HADR – Recuperar información de instantáneas del grupo de datos lógicos hadr

La vista administrativa SNAPHADR y la función de tabla SNAP_GET_HADR devuelven información sobre la recuperación de catástrofes de alta disponibilidad a partir de la instantánea de una base de datos, en concreto el grupo de datos lógicos hadr.

SNAPHADR, vista administrativa

Esta vista administrativa permite recuperar la información de instantáneas del grupo de datos lógicos hadr para la base de datos conectada actualmente. Esta vista sólo devuelve los datos si la base de datos es principal o de recuperación de catástrofes de alta disponibilidad (HADR) en espera.

La vista administrativa SNAPHADR, utilizada con las vistas administrativas SNAPDB, SNAPDB_MEMORY_POOL, SNAPDETAILLOG y SNAPSTORAGE_PATHS, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR DATABASE ON alias de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 150 en la página 597 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPHADR
- Privilegio CONTROL en la vista administrativa SNAPHADR
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_HADR
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMAINT
- SYSADM

Ejemplo

Recuperar la información sobre el estado y la configuración de HADR en la base de datos HADR principal.

```
SELECT SUBSTR(DB_NAME, 1, 8) AS DBNAME, HADR_ROLE, HADR_STATE,  
       HADR_SYNCMODE, HADR_CONNECT_STATUS  
FROM SYSIBMADM.SNAPHADR
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| DBNAME | HADR_ROLE | HADR_STATE | HADR_SYNCMODE | HADR_CONNECT_STATUS |
|--------|-----------|------------|---------------|---------------------|
| SAMPLE | PRIMARY | PEER | SYNC | CONNECTED |

1 registro(s) seleccionado(s).

SNAP_GET_HADR, función de tabla

La función de tabla SNAP_GET_HADR devuelve la misma información que la vista administrativa SNAPHADR, pero permite recuperar la información para una base de datos concreta en una partición específica de la base de datos, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_HADR, utilizada con las funciones de tabla SNAP_GET_DB_V95, SNAP_GET_DB_MEMORY_POOL, SNAP_GET_DETAILLOG_V91 y SNAP_GET_STORAGE_PATHS, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR ALL DATABASES.

Consulte la Tabla 150 en la página 597 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
▶▶ SNAP_GET_HADR ( [ nombrebd ] [ , númparticiónbd ] ) ▶▶
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique una serie vacía para tomar la instantánea desde la base de datos conectada actualmente. Especifique un valor NULL para tomar la instantánea desde todas las bases de datos que se encuentran en la misma instancia que la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_HADR toma una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_HADR
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar la información sobre el estado y la configuración de HADR en todas las bases de datos.

```
SELECT SUBSTR(DB_NAME, 1, 8) AS DBNAME, HADR_ROLE, HADR_STATE,
       HADR_SYNCMODE, HADR_CONNECT_STATUS
FROM TABLE (SNAP_GET_HADR (CAST (NULL as VARCHAR(128)), 0)) as T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| DBNAME | HADR_ROLE | HADR_STATE | HADR_SYNCMODE | HADR_CONNECT_STATUS |
|--------|-----------|--------------|---------------|---------------------|
| SAMPLE | PRIMARY | PEER | SYNC | CONNECTED |
| TESTDB | PRIMARY | DISCONNECTED | NEARSYNC | DISCONNECTED |

2 registro(s) seleccionado(s).

Información devuelta

Tabla 150. Información devuelta por la vista administrativa SNAPHADR y la función de tabla SNAP_GET_HADR

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| HADR_ROLE | VARCHAR(10) | hadr_role - Rol HADR. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> PRIMARY STANDARD STANDBY |
| HADR_STATE | VARCHAR(14) | hadr_state - Estado HADR. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> DISCONNECTED LOCAL_CATCHUP PEER REM_CATCH_PEN REM_CATCHUP |
| HADR_SYNCMODE | VARCHAR(10) | hadr_syncmode - Modalidad de sincronización HADR. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> ASYNCR NEARSYNCR SYNCR |

Tabla 150. Información devuelta por la vista administrativa SNAPHADR y la función de tabla SNAP_GET_HADR (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------|---------------|--|
| HADR_CONNECT_STATUS | VARCHAR(12) | hadr_connect_status - Estado de conexión HADR. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • CONGESTED • CONNECTED • DISCONNECTED |
| HADR_CONNECT_TIME | TIMESTAMP | hadr_connect_time - Hora de conexión de HADR |
| HADR_HEARTBEAT | INTEGER | hadr_heartbeat - Latido de HADR |
| HADR_LOCAL_HOST | VARCHAR(255) | hadr_local_host - Sistema principal local de HADR |
| HADR_LOCAL_SERVICE | VARCHAR(40) | hadr_local_service - Servicio local de HADR |
| HADR_REMOTE_HOST | VARCHAR(255) | hadr_remote_host - Sistema principal remoto de HADR |
| HADR_REMOTE_SERVICE | VARCHAR(40) | hadr_remote_service - Servicio remoto de HADR |
| HADR_REMOTE_INSTANCE | VARCHAR(128) | hadr_remote_instance - Instancia remota de HADR |
| HADR_TIMEOUT | BIGINT | hadr_timeout - Tiempo de espera de HADR |
| HADR_PRIMARY_LOG_FILE | VARCHAR(255) | hadr_primary_log_file - Archivo de anotaciones cronológicas primario de HADR |
| HADR_PRIMARY_LOG_PAGE | BIGINT | hadr_primary_log_page - Página de anotaciones cronológicas primarias de HADR |
| HADR_PRIMARY_LOG_LSN | BIGINT | hadr_primary_log_lsn - LSN de anotaciones cronológicas primarias de HADR |
| HADR_STANDBY_LOG_FILE | VARCHAR(255) | hadr_standby_log_file - Archivo de anotaciones cronológica en espera de HADR |
| HADR_STANDBY_LOG_PAGE | BIGINT | hadr_standby_log_page - Página de anotaciones cronológicas en espera de HADR |
| HADR_STANDBY_LOG_LSN | BIGINT | hadr_standby_log_lsn - LSN de anotaciones cronológicas en espera de HADR |
| HADR_LOG_GAP | BIGINT | hadr_log_gap - Distancia de anotación cronológica de HADR |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa SNAPLOCK y función de tabla SNAP_GET_LOCK – Recuperar información de instantáneas del grupo de datos lógicos lock

La vista administrativa SNAPLOCK y la función de tabla SNAP_GET_LOCK devuelven información de instantáneas acerca de los bloqueos, en concreto el grupo de datos lógicos lock.

SNAPLOCK, vista administrativa

Esta vista administrativa permite recuperar la información de instantáneas del grupo de datos lógicos lock para la base de datos conectada actualmente.

La vista administrativa SNAPLOCK, utilizada con la vista administrativa SNAPLOCKWAIT, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR LOCKS ON alias de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 151 en la página 601 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPLOCK
- Privilegio CONTROL en la vista administrativa SNAPLOCK
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_LOCK
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar información de bloqueo de la partición 0 de la base de datos conectada actualmente.

```
SELECT AGENT_ID, LOCK_OBJECT_TYPE, LOCK_MODE, LOCK_STATUS
FROM SYSIBMADM.SNAPLOCK WHERE DBPARTITIONNUM = 0
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| AGENT_ID | LOCK_OBJECT_TYPE | LOCK_MODE | LOCK_STATUS |
|----------|------------------|-----------|-------------|
| 7 | TABLE | IX | GRNT |

1 registro(s) seleccionado(s).

SNAP_GET_LOCK, función de tabla

La función de tabla SNAP_GET_LOCK devuelve la misma información que la vista administrativa SNAPLOCK, pero permite recuperar la información para una base de datos concreta en una partición específica de la base de datos, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_LOCK, utilizada con la función de tabla SNAP_GET_LOCKWAIT, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR LOCKS ON alias de base de datos.

Consulte la Tabla 151 en la página 601 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
▶▶ SNAP_GET_LOCK ( ( nombrebd [ , númparticiónbd ] ) ) ▶▶
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique un valor nulo o una serie vacía para tomar la instantánea desde la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_LOCK toma una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_LOCK
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMANT
- SYSADM

Ejemplo

Recuperar información de bloqueo de la partición actual de la base de datos conectada actualmente.

```
SELECT AGENT_ID, LOCK_OBJECT_TYPE, LOCK_MODE, LOCK_STATUS
FROM TABLE(SNAP_GET_LOCK('1',-1)) as T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
AGENT_ID      LOCK_OBJECT_TYPE  LOCK_MODE  LOCK_STATUS
-----
          680 INTERNALV_LOCK      S          GRNT
          680 INTERNALP_LOCK      S          GRNT
```

2 registro(s) seleccionado(s).

Información devuelta

Tabla 151. Información que devuelve la vista administrativa SNAPLOCK y la función de tabla SNAP_GET_LOCK

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| AGENT_ID | BIGINT | agent_id - Descriptor de contexto de aplicación (ID de agente) |
| TAB_FILE_ID | BIGINT | table_file_id - Identificación de archivo de tabla |

Tabla 151. Información que devuelve la vista administrativa SNAPLOCK y la función de tabla SNAP_GET_LOCK (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|--|
| LOCK_OBJECT_TYPE | VARCHAR(18) | <p>lock_object_type - Tipo de objeto de bloqueo esperado. Esta interfaz devuelve un identificador de texto basándose en las sentencias define de sqlmon.h y es uno de los siguientes:</p> <ul style="list-style-type: none"> • AUTORESIZE_LOCK • AUTOSTORAGE_LOCK • BLOCK_LOCK • EOT_LOCK • INPLACE_REORG_LOCK • INTERNAL_LOCK • INTERNALB_LOCK • INTERNALC_LOCK • INTERNALJ_LOCK • INTERNALL_LOCK • INTERNALO_LOCK • INTERNALQ_LOCK • INTERNALP_LOCK • INTERNALS_LOCK • INTERNALT_LOCK • INTERNALV_LOCK • KEYVALUE_LOCK • ROW_LOCK • SYSBOOT_LOCK • TABLE_LOCK • TABLE_PART_LOCK • TABLESPACE_LOCK • XML_PATH_LOCK |
| LOCK_MODE | VARCHAR(10) | <p>lock_mode - Modalidad de bloqueo. Esta interfaz devuelve un identificador de texto basándose en las sentencias define de sqlmon.h y es uno de los siguientes:</p> <ul style="list-style-type: none"> • IN • IS • IX • NON (si no existe bloqueo) • NS • NW • S • SIX • U • X • Z |

Tabla 151. Información que devuelve la vista administrativa SNAPLOCK y la función de tabla SNAP_GET_LOCK (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|---|
| LOCK_STATUS | VARCHAR(10) | lock_status - Estado de bloqueo. Esta interfaz devuelve un identificador de texto basándose en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • CONV • GRNT |
| LOCK_ESCALATION | SMALLINT | lock_escalation - Escalamiento de bloqueos |
| TABNAME | VARCHAR(128) | table_name - Nombre de tabla |
| TABSCHEMA | VARCHAR(128) | table_schema - Nombre de esquema de tabla |
| TBSP_NAME | VARCHAR(128) | tablespace_name - Nombre de espacio de tablas |
| LOCK_ATTRIBUTES | VARCHAR(128) | lock_attributes - Atributos de bloqueo. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h. Si no hay bloqueos, el identificador de texto es NONE; de lo contrario, es cualquier combinación de los elementos siguientes separados por un signo '+': <ul style="list-style-type: none"> • ALLOW_NEW • DELETE_IN_BLOCK • ESCALATED • INSERT • NEW_REQUEST • RR • RR_IN_BLOCK • UPDATE_DELETE • WAIT_FOR_AVAIL |
| LOCK_COUNT | BIGINT | lock_count - Número de bloqueos |

Tabla 151. Información que devuelve la vista administrativa SNAPLOCK y la función de tabla SNAP_GET_LOCK (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| LOCK_CURRENT_MODE | VARCHAR(10) | lock_current_mode - Modalidad de bloqueo original antes de la conversión. Esta interfaz devuelve un identificador de texto basándose en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • IN • IS • IX • NON (si no existe bloqueo) • NS • NW • S • SIX • U • X • Z |
| LOCK_HOLD_COUNT | BIGINT | lock_hold_count - Número de bloqueos retenidos |
| LOCK_NAME | VARCHAR(32) | lock_name - Nombre de bloqueo |
| LOCK_RELEASE_FLAGS | BIGINT | lock_release_flags - Distintivos de liberación de bloqueo |
| DATA_PARTITION_ID | INTEGER | data_partition_id - Identificador de la partición de datos. Para una tabla no particionada, este elemento será NULL. |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa SNAPLOCKWAIT y función de tabla SNAP_GET_LOCKWAIT – Recuperar información de instantáneas del grupo de datos lógicos lockwait

La vista administrativa SNAPLOCKWAIT y la función de tabla SNAP_GET_LOCKWAIT devuelven información de instantáneas acerca de las esperas de bloqueo, en concreto el grupo de datos lógicos lockwait.

SNAPLOCKWAIT, vista administrativa

Esta vista administrativa permite recuperar la información de instantáneas del grupo de datos lógicos lockwait para la base de datos conectada actualmente.

La vista administrativa SNAPLOCKWAIT, utilizada con la vista administrativa SNAPLOCK, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR LOCKS ON alias de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 152 en la página 607 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPLOCKWAIT
- Privilegio CONTROL en la vista administrativa SNAPLOCKWAIT
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_LOCKWAIT
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMAINT
- SYSADM

Ejemplo

Recuperar información acerca de las esperas de bloqueo en la partición de base de datos 0 para la base de datos conectada actualmente.

```
SELECT AGENT_ID, LOCK_MODE, LOCK_OBJECT_TYPE, AGENT_ID_HOLDING_LK,  
       LOCK_MODE_REQUESTED FROM SYSIBMADM.SNAPLOCKWAIT  
WHERE DBPARTITIONNUM = 0
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
AGENT_ID    LOCK_MODE  LOCK_OBJECT_TYPE  ...  
-----  
          7 IX          TABLE          ...
```

1 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```
... AGENT_ID_HOLDING_LK  LOCK_MODE_REQUESTED  
... -----  
...                   12 IS
```

SNAP_GET_LOCKWAIT, función de tabla

La función de tabla SNAP_GET_LOCKWAIT devuelve la misma información que la vista administrativa SNAPLOCKWAIT, pero permite recuperar la información para una base de datos concreta en una partición específica de la base de datos, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_LOCKWAIT, utilizada con la función de tabla SNAP_GET_LOCK, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR LOCKS ON alias de base de datos.

Consulte la Tabla 152 en la página 607 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
▶▶ SNAP_GET_LOCKWAIT ( nombrebd [ , númparticiónbd ] ) ▶▶
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique un valor nulo o una serie vacía para tomar la instantánea desde la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_LOCKWAIT tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_LOCKWAIT
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT

- SYSADM

Ejemplo

Recuperar información acerca de las esperas de bloqueo en la partición actual de la base de datos conectada actualmente.

```
SELECT AGENT_ID, LOCK_MODE, LOCK_OBJECT_TYPE, AGENT_ID_HOLDING_LK,
       LOCK_MODE_REQUESTED FROM TABLE(SNAP_GET_LOCKWAIT('',-1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
AGENT_ID      LOCK_MODE  LOCK_OBJECT_TYPE  ...
-----
          12 X          ROW_LOCK          ...
```

1 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```
... AGENT_ID_HOLDING_LK  LOCK_MODE_REQUESTED
... -----
...                      7 X
```

Nota sobre uso

Para ver información de esperas de bloqueo, primero deberá activar el conmutador de supervisor LOCK por omisión en la configuración del gestor de bases de datos. Para que el cambio tenga efecto inmediatamente, conéctese explícitamente a la instancia utilizando CLP y luego vuelva a emitir el mandato CLP:

```
UPDATE DATABASE MANAGER CONFIGURATION CLP USING DFT_MON_LOCK ON
```

Los valores por omisión también se puede activar mediante el procedimiento almacenado ADMIN_CMD. Por ejemplo:

```
CALL SYSPROC.ADMIN_CMD('update dbm cfg using DFT_MON_LOCK ON')
```

Si se utiliza el procedimiento almacenado ADMIN_CMD o si se utiliza el mandato clp sin conectarse previamente a una instancia, la instancia deberá reciclarse antes de que el cambio tiene efecto.

Información devuelta

Tabla 152. Información que devuelve la vista administrativa SNAPLOCKWAIT y la función de tabla SNAP_GET_LOCKWAIT

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| AGENT_ID | BIGINT | agent_id - Descriptor de contexto de aplicación (ID de agente) |
| SUBSECTION_NUMBER | BIGINT | ss_number - Número de subsección |

Tabla 152. Información que devuelve la vista administrativa SNAPLOCKWAIT y la función de tabla SNAP_GET_LOCKWAIT (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|---|
| LOCK_MODE | VARCHAR(10) | lock_mode - Modalidad de bloqueo. Esta interfaz devuelve un identificador de texto basándose en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • IN • IS • IX • NON (si no existe bloqueo) • NS • NW • S • SIX • U • X • Z |
| LOCK_OBJECT_TYPE | VARCHAR(18) | lock_object_type - Tipo de objeto de bloqueo esperado. Esta interfaz devuelve un identificador de texto basándose en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • AUTORESIZE_LOCK • AUTOSTORAGE_LOCK • BLOCK_LOCK • EOT_LOCK • INPLACE_REORG_LOCK • INTERNAL_LOCK • INTERNALB_LOCK • INTERNALC_LOCK • INTERNALJ_LOCK • INTERNALL_LOCK • INTERNALO_LOCK • INTERNALQ_LOCK • INTERNALP_LOCK • INTERNALS_LOCK • INTERNALT_LOCK • INTERNALV_LOCK • KEYVALUE_LOCK • ROW_LOCK • SYSBOOT_LOCK • TABLE_LOCK • TABLE_PART_LOCK • TABLESPACE_LOCK • XML_PATH_LOCK |

Tabla 152. Información que devuelve la vista administrativa SNAPLOCKWAIT y la función de tabla SNAP_GET_LOCKWAIT (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|----------------------|---------------|--|
| AGENT_ID_HOLDING_LK | BIGINT | agent_id_holding_lock - ID de agente que mantiene el bloqueo |
| LOCK_WAIT_START_TIME | TIMESTAMP | lock_wait_start_time - Indicación de fecha y hora de inicio de espera de bloqueo |
| LOCK_MODE_REQUESTED | VARCHAR(10) | lock_mode_requested - Modalidad de bloqueo solicitada. Esta interfaz devuelve un identificador de texto basándose en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • IN • IS • IX • NON (si no existe bloqueo) • NS • NW • S • SIX • U • X • Z |
| LOCK_ESCALATION | SMALLINT | lock_escalation - Escalamiento de bloqueos |
| TABNAME | VARCHAR(128) | table_name - Nombre de tabla |
| TABSHEMA | VARCHAR(128) | table_schema - Nombre de esquema de tabla |
| TBSP_NAME | VARCHAR(128) | tablespace_name - Nombre de espacio de tablas |
| APPL_ID_HOLDING_LK | VARCHAR(128) | appl_id_holding_lk - ID de aplicación que mantiene el bloqueo |

Tabla 152. Información que devuelve la vista administrativa SNAPLOCKWAIT y la función de tabla SNAP_GET_LOCKWAIT (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| LOCK_ATTRIBUTES | VARCHAR(128) | lock_attributes - Atributos de bloqueo. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h. Si no hay bloqueos, el identificador de texto es NONE; de lo contrario, es cualquier combinación de los elementos siguientes separados por un signo '+': <ul style="list-style-type: none"> • ALLOW_NEW • DELETE_IN_BLOCK • ESCALATED • INSERT • NEW_REQUEST • RR • RR_IN_BLOCK • UPDATE_DELETE • WAIT_FOR_AVAIL |
| LOCK_CURRENT_MODE | VARCHAR(10) | lock_current_mode - Modalidad de bloqueo original antes de la conversión. Esta interfaz devuelve un identificador de texto basándose en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • IN • IS • IX • NON (si no existe bloqueo) • NS • NW • S • SIX • U • X • Z |
| LOCK_NAME | VARCHAR(32) | lock_name - Nombre de bloqueo |
| LOCK_RELEASE_FLAGS | BIGINT | lock_release_flags - Distintivos de liberación de bloqueo. |
| DATA_PARTITION_ID | INTEGER | data_partition_id - Identificador de la partición de datos. Para una tabla no particionada, este elemento será NULL. |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa SNAPSTMT y función de tabla SNAP_GET_STMT – Recuperar información de instantáneas de sentencias

La vista administrativa SNAPSTMT y la función de tabla SNAP_GET_STMT devuelven información sobre sentencias SQL o XQuery desde una instantánea de la aplicación.

SNAPSTMT, vista administrativa

Esta vista administrativa le permite recuperar información acerca de la instantánea de sentencia para la base de datos conectada actualmente.

La vista administrativa SNAPSTMT, utilizada con las vistas administrativas SNAPAGENT, SNAPAGENT_MEMORY_POOL, SNAPAPPL, SNAPAPPL_INFO y SNAPSUBSECTION, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR APPLICATIONS ON alias de base de datos, pero recupera datos de todas las particiones de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 153 en la página 613 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPSTMT
- Privilegio CONTROL en la vista administrativa SNAPSTMT
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_STMT
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMAINT
- SYSADM

Ejemplo

Recuperar filas leídas, grabadas y aquellas en que se han efectuado operaciones, para las sentencias ejecutadas en la base de datos de partición única conectada actualmente.

```
SELECT SUBSTR(STMT_TEXT,1,30) AS STMT_TEXT, ROWS_READ, ROWS_WRITTEN,  
       STMT_OPERATION FROM SYSIBMADM.SNAPSTMT
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| STMT_TEXT | ROWS_READ | ROWS_WRITTEN | STMT_OPERATION |
|-----------|-----------|--------------|-----------------|
| - | | 0 | 0 FETCH |
| - | | 0 | 0 STATIC_COMMIT |

2 registro(s) seleccionado(s).

SNAP_GET_STMT, función de tabla

La función de tabla SNAP_GET_STMT devuelve la misma información que la vista administrativa SNAP_STMT, pero permite recuperar la información para una base de datos concreta en una partición específica de la base de datos, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_STMT, utilizada con las funciones de tablas SNAP_GET_AGENT, SNAP_GET_AGENT_MEMORY_POOL, SNAP_GET_APPL_V95, SNAP_GET_APPL_INFO_V95 y SNAP_GET_SUBSECTION, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR ALL APPLICATIONS, pero recupera datos de todas las particiones de base de datos.

Consulte la Tabla 153 en la página 613 para ver una lista completa de la información que se puede devolver.

Sintaxis

```

▶▶ SNAP_GET_STMT ( ( nombrebd [ , númparticiónbd ] ) )

```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique una serie vacía para tomar la instantánea desde la base de datos conectada actualmente. Especifique un valor NULL para tomar la instantánea desde todas las bases de datos que se encuentran en la misma instancia que la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es

posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_STMT toma una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_STMT
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMANT
- SYSADM

Ejemplo

Recuperar filas leídas, grabadas y aquellas en que se han efectuado operaciones, para las sentencias ejecutadas en la partición actual de la base de datos conectada actualmente.

```
SELECT SUBSTR(STMT_TEXT,1,30) AS STMT_TEXT, ROWS_READ,
       ROWS_WRITTEN, STMT_OPERATION FROM TABLE(SNAP_GET_STMT(' ', -1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
STMT_TEXT                ROWS_READ    ...
-----
update t set a=3         0 ...
SELECT SUBSTR(STMT_TEXT,1,30) 0 ...
-                        0 ...
-                        0 ...
update t set a=2        9 ...
...
5 registro(s) seleccionado(s).    ...
```

Datos de salida de esta consulta (continuación).

```
... ROWS_WRITTEN    STMT_OPERATION
... -----
...                0 EXECUTE_IMMEDIATE
...                0 FETCH
...                0 NONE
...                0 NONE
...                1 EXECUTE_IMMEDIATE
...
...
```

Información devuelta

Tabla 153. Información que devuelve la vista administrativa SNAPSTMT y la función de tabla SNAP_GET_STMT

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |

Tabla 153. Información que devuelve la vista administrativa SNAPSTMT y la función de tabla SNAP_GET_STMT (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|--|
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| AGENT_ID | BIGINT | agent_id - Descriptor de contexto de aplicación (ID de agente) |
| ROWS_READ | BIGINT | rows_read - Filas leídas |
| ROWS_WRITTEN | BIGINT | rows_written - Filas grabadas |
| NUM_AGENTS | BIGINT | num_agents - Número de agentes trabajando en una sentencia |
| AGENTS_TOP | BIGINT | agents_top - Número de agentes creados |
| STMT_TYPE | VARCHAR(20) | stmt_type - Tipo de sentencia. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • DYNAMIC • NON_STMT • STATIC • STMT_TYPE_UNKNOWN |

Tabla 153. Información que devuelve la vista administrativa SNAPSTMT y la función de tabla SNAP_GET_STMT (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------|---------------|---|
| STMT_OPERATION | VARCHAR(20) | stmt_operation/operation - Operación de sentencia. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • CALL • CLOSE • COMPILE • DESCRIBE • EXECUTE • EXECUTE_IMMEDIATE • FETCH • FREE_LOCATOR • GETAA • GETNEXTCHUNK • GETTA • NONE • OPEN • PREP_COMMIT • PREP_EXEC • PREP_OPEN • PREPARE • REBIND • REDIST • REORG • RUNSTATS • SELECT • SET • STATIC_COMMIT • STATIC_ROLLBACK |
| SECTION_NUMBER | BIGINT | section_number - Número de sección |
| QUERY_COST_ESTIMATE | BIGINT | query_cost_estimate - Estimación de coste de consulta |
| QUERY_CARD_ESTIMATE | BIGINT | query_card_estimate - Estimación de consulta de número de filas |
| DEGREE_PARALLELISM | BIGINT | degree_parallelism - Grado de paralelismo |
| STMT_SORTS | BIGINT | stmt_sorts - Clasificaciones de sentencias |
| TOTAL_SORT_TIME | BIGINT | total_sort_time - Tiempo de clasificación total |
| SORT_OVERFLOW | BIGINT | sort_overflows - Desbordamientos de clasificación |

Tabla 153. Información que devuelve la vista administrativa SNAPSTMT y la función de tabla SNAP_GET_STMT (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|----------------------|---------------|---|
| INT_ROWS_DELETED | BIGINT | int_rows_deleted - Filas internas suprimidas |
| INT_ROWS_UPDATED | BIGINT | int_rows_updated - Filas internas actualizadas |
| INT_ROWS_INSERTED | BIGINT | int_rows_inserted - Filas internas insertadas |
| FETCH_COUNT | BIGINT | fetch_count - Número de captaciones satisfactorias |
| STMT_START | TIMESTAMP | stmt_start - Indicación de fecha y hora de inicio de operación de sentencia |
| STMT_STOP | TIMESTAMP | stmt_stop - Indicación de fecha y hora de detención de operación de sentencia |
| STMT_USR_CPU_TIME_S | BIGINT | stmt_usr_cpu_time - Tiempo de CPU de usuario utilizado por sentencia |
| STMT_USR_CPU_TIME_MS | BIGINT | stmt_usr_cpu_time - Tiempo de CPU de usuario utilizado por sentencia |
| STMT_SYS_CPU_TIME_S | BIGINT | stmt_sys_cpu_time - Tiempo de CPU de sistema utilizado por sentencia |
| STMT_SYS_CPU_TIME_MS | BIGINT | stmt_sys_cpu_time - Tiempo de CPU de sistema utilizado por sentencia |
| STMT_ELAPSED_TIME_S | BIGINT | stmt_elapsed_time - Tiempo transcurrido de sentencia más reciente |
| STMT_ELAPSED_TIME_MS | BIGINT | stmt_elapsed_time - Tiempo transcurrido de sentencia más reciente |
| BLOCKING_CURSOR | SMALLINT | blocking_cursor - Cursor de bloqueo |
| STMT_NODE_NUMBER | SMALLINT | stmt_node_number - Nodo de sentencia |
| CURSOR_NAME | VARCHAR(128) | cursor_name - Nombre de cursor |
| CREATOR | VARCHAR(128) | creator - Creador de aplicación |
| PACKAGE_NAME | VARCHAR(128) | package_name - Nombre de paquete |
| STMT_TEXT | CLOB(16 M) | stmt_text - Texto de sentencia de SQL |
| CONSISTENCY_TOKEN | VARCHAR(128) | consistency_token - Señal de coherencia del paquete |
| PACKAGE_VERSION_ID | VARCHAR(128) | package_version_id - Versión del paquete |

Tabla 153. Información que devuelve la vista administrativa SNAPSTMT y la función de tabla SNAP_GET_STMT (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|--|
| POOL_DATA_L_READS | BIGINT | pool_data_l_reads - Lecturas lógicas de datos de la agrupación de almacenamientos intermedios |
| POOL_DATA_P_READS | BIGINT | pool_data_p_reads - Lecturas físicas de datos de la agrupación de almacenamientos intermedios |
| POOL_INDEX_L_READS | BIGINT | pool_index_l_reads - Lecturas lógicas de índices de la agrupación de almacenamientos intermedios |
| POOL_INDEX_P_READS | BIGINT | pool_index_p_reads - Lecturas físicas de índices de la agrupación de almacenamientos intermedios |
| POOL_XDA_L_READS | BIGINT | pool_xda_l_reads - Elemento del supervisor de las lecturas lógicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_XDA_P_READS | BIGINT | pool_xda_p_reads - Elemento del supervisor de las lecturas físicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_L_READS | BIGINT | pool_temp_data_l_reads - Lecturas lógicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_P_READS | BIGINT | pool_temp_data_p_reads - Lecturas físicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_L_READS | BIGINT | pool_temp_index_l_reads - Lecturas lógicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_P_READS | BIGINT | pool_temp_index_p_reads - Lecturas físicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_L_READS | BIGINT | pool_temp_xda_l_reads - Lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_P_READS | BIGINT | pool_temp_xda_p_reads - Elemento del supervisor de las lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa **SNAPSTORAGE_PATHS** y función de tabla **SNAP_GET_STORAGE_PATHS_V97** - Recuperar información de vía de acceso de almacenamiento automático

La vista administrativa **SNAPSTORAGE_PATHS** y la función de tabla **SNAP_GET_STORAGE_PATHS_V97** devuelven una lista de vías de acceso del almacenamiento automático para la base de datos, incluida la información del sistema de archivos correspondiente a cada vía de acceso del almacenamiento, concretamente del grupo de datos lógicos `db_storage_group`.

SNAPSTORAGE_PATHS, vista administrativa

Esta vista administrativa permite recuperar la información de vía de acceso del almacenamiento automático para la base de datos conectada actualmente.

La vista administrativa **SNAPSTORAGE_PATHS**, utilizada con las vistas administrativas **SNAPDB**, **SNAPDETAILLOG**, **SNAPHADR** y **SNAPDB_MEMORY_POOL**, ofrece información equivalente al mandato `CLP GET SNAPSHOT FOR DATABASE ON` alias de base de datos.

El esquema es `SYSIBMADM`.

Consulte la Tabla 154 en la página 620 para ver una lista completa de la información que se puede devolver.

Autorización

- Autorización `SYSMON`
- Privilegio `SELECT` o `CONTROL` para la vista administrativa **SNAPSTORAGE_PATHS** y privilegio `EXECUTE` para la función de tabla **SNAP_GET_STORAGE_PATHS_V97**.

Ejemplo

Recuperar la vía de acceso de almacenamiento para la base de datos de una sola partición conectada actualmente.

```
SELECT SUBSTR(DB_NAME,1,8) AS DB_NAME, SUBSTR(DB_STORAGE_PATH,1,8)
      AS DB_STORAGE_PATH, SUBSTR(HOSTNAME,1,10) AS HOSTNAME
FROM SYSIBMADM.SNAPSTORAGE_PATHS
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
DB_NAME  DB_STORAGE_PATH  HOSTNAME
-----  -
STOPATH  d:                   JESSICAE
```

1 registro(s) seleccionado(s).

SNAP_GET_STORAGE_PATHS_V97, función de tabla

La función de tabla **SNAP_GET_STORAGE_PATHS_V97** devuelve la misma información que la vista administrativa **SNAPSTORAGE_PATHS_V**, pero le permite recuperar la información para una base de datos específica de una partición de base de datos específica, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_STORAGE_PATHS_V97, utilizada con las funciones de tabla SNAP_GET_DB_V95, SNAP_GET_DETAILLOG_V91, SNAP_GET_HADR y SNAP_GET_DB_MEMORY_POOL, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR ALL DATABASES.

Consulte la Tabla 154 en la página 620 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
▶▶ SNAP_GET_STORAGE_PATHS_V97 ( ( nombre-bd [ , númer-partición-bd ] ) ) ▶▶
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique una serie vacía para tomar la instantánea desde la base de datos conectada actualmente. Especifique un valor NULL para tomar la instantánea desde todas las bases de datos que se encuentran en la misma instancia que la base de datos conectada actualmente.

númerparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *númerparticiónbd* se establece en NULL, -1 se establece implícitamente para *númerparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númerparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_STORAGE_PATHS_V97 tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP_GET_STORAGE_PATHS_V97.

Ejemplos

Recuperar la información de vía de acceso de almacenamiento para todas las bases de datos activas.

```
SELECT SUBSTR(DB_NAME,1,8) AS DB_NAME, DB_STORAGE_PATH
FROM TABLE(SNAP_GET_STORAGE_PATHS_V97(CAST (NULL AS VARCHAR(128)), -1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
DB_NAME  DB_STORAGE_PATH
-----
STOPATH  /home/jessicae/sdb
MYDB     /home/jessicae/mdb
```

2 registro(s) seleccionado(s)

Información devuelta

Para que se devuelva la información del sistema de archivos, debe activarse el conmutador de supervisor BUFFERPOOL.

Tabla 154. Información devuelta por la vista administrativa SNAPSTORAGE_PATHS y la función de tabla SNAP_GET_STORAGE_PATHS_V97

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| DB_STORAGE_PATH | VARCHAR(256) | db_storage_path - Vía de acceso de almacenamiento automático |
| DB_STORAGE_PATH_WITH_DPE | VARCHAR(256) | Vía de acceso de almacenamiento automático que contiene la expresión de partición de base de datos (DPE) sin evaluar. Devuelve NULL si la vía de acceso de almacenamiento no contiene una DPE. |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |
| DB_STORAGE_PATH_STATE | VARCHAR(16) | Estado de la vía de acceso de almacenamiento automático (actualmente, los valores pueden ser "IN_USE", "NOT_IN_USE", "DROP_PENDING"). |
| FS_ID | VARCHAR(22) | fs_id - Número de identificación del sistema de archivos exclusivo |
| FS_TOTAL_SIZE | BIGINT | fs_total_size - Tamaño total de un sistema de archivos |
| FS_USED_SIZE | BIGINT | fs_used_size - Cantidad de espacio utilizada por un sistema de archivos |
| STO_PATH_FREE_SIZE | BIGINT | sto_path_free_sz - Espacio libre para la vía de acceso de almacenamiento automático |

Vista administrativa SNAPSUBSECTION y función de tabla SNAP_GET_SUBSECTION – Recuperar información de instantáneas del grupo de supervisor lógico de la subsección

La vista administrativa SNAPSUBSECTION y la función de tabla SNAP_GET_SUBSECTION devuelven información sobre las subsecciones de aplicaciones, concretamente la agrupación del supervisor lógico de la subsección.

SNAPSUBSECTION, vista administrativa

Esta vista administrativa permite recuperar la información de instantáneas del grupo de supervisor lógico de la subsección para la base de datos conectada actualmente.

La vista administrativa SNAPSUBSECTION, utilizada con las vistas administrativas SNAPAGENT, SNAPAGENT_MEMORY_POOL, SNAPAPPL, SNAPAPPL_INFO y SNAPSTMT, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR APPLICATIONS ON alias de base de datos, pero recupera datos de todas las particiones de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 155 en la página 623 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPSUBSECTION
- Privilegio CONTROL en la vista administrativa SNAPSUBSECTION
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_SUBSECTION
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMAINT
- SYSADM

Ejemplo

Obtener el estado de las subsecciones que se ejecutan en todas las particiones de base de datos.

```
SELECT DB_NAME, STMT_TEXT, SS_STATUS, DBPARTITIONNUM
      FROM SYSIBMADM.SNAPSUBSECTION
      ORDER BY DB_NAME, SS_STATUS, DBPARTITIONNUM
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| DB_NAME | STMT_TEXT | SS_STATUS | DBPARTITIONNUM |
|---------|------------------------|-----------|----------------|
| SAMPLE | select * from EMPLOYEE | EXEC | 0 |
| SAMPLE | select * from EMPLOYEE | EXEC | 1 |

SNAP_GET_SUBSECTION, función de tabla

La función de tabla SNAP_GET_SUBSECTION devuelve la misma información que la vista administrativa SNAPSUBSECTION, pero permite recuperar la información de una base de datos concreta en una partición específica de la base de datos, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

Consulte la Tabla 155 en la página 623 para ver una lista completa de la información que se puede devolver.

La función de tabla SNAP_GET_SUBSECTION, utilizada con las funciones de tablas SNAP_GET_AGENT, SNAP_GET_AGENT_MEMORY_POOL, SNAP_GET_APPL_V95, SNAP_GET_APPL_INFO_V95 y SNAP_GET_STMT, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR ALL APPLICATIONS, pero recupera datos de todas las particiones de base de datos.

Sintaxis

```

▶▶ SNAP_GET_SUBSECTION ( ( nombrebd [ , númparticiónbd ] ) )

```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique una serie vacía para tomar la instantánea desde la base de datos conectada actualmente. Especifique un valor NULL para tomar la instantánea desde todas las bases de datos que se encuentran en la misma instancia que la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla

SNAP_GET_SUBSECTION tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_SUBSECTION
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Obtener el estado de las subsecciones que se ejecutan en todas las particiones de base de datos.

```
SELECT DB_NAME, STMT_TEXT, SS_STATUS, DBPARTITIONNUM
FROM TABLE(SYSPROC.SNAP_GET_SUBSECTION( ' ', 0 )) as T
ORDER BY DB_NAME, SS_STATUS, DBPARTITIONNUM
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
DB_NAME      STMT_TEXT                SS_STATUS      DBPARTITIONNUM
-----
SAMPLE      select * from EMPLOYEE   EXEC           0
SAMPLE      select * from EMPLOYEE   EXEC           1
```

Información devuelta

Tabla 155. Información que devuelve la vista administrativa SNAPSUBSECTION y la función de tabla SNAP_GET_SUBSECTION

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| STMT_TEXT | CLOB(16 M) | stmt_text - Texto de sentencia de SQL |
| SS_EXEC_TIME | BIGINT | ss_exec_time - Tiempo transcurrido de ejecución de subsección |
| TQ_TOT_SEND_SPILLS | BIGINT | tq_tot_send_spills - Número total de almacenamientos intermedios de colas de tabla desbordados |
| TQ_CUR_SEND_SPILLS | BIGINT | tq_cur_send_spills - Número actual de almacenamientos intermedios de colas de tabla desbordados |

Tabla 155. Información que devuelve la vista administrativa SNAPSUBSECTION y la función de tabla SNAP_GET_SUBSECTION (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| TQ_MAX_SEND_SPILLS | BIGINT | tq_max_send_spills - Número máximo de desbordamientos de almacenamientos intermedios de colas de tabla |
| TQ_ROWS_READ | BIGINT | tq_rows_read - Número de filas leídas de las colas de tabla |
| TQ_ROWS_WRITTEN | BIGINT | tq_rows_written - Número de filas grabadas en las colas de tabla |
| ROWS_READ | BIGINT | rows_read - Filas leídas |
| ROWS_WRITTEN | BIGINT | rows_written - Filas grabadas |
| SS_USR_CPU_TIME_S | BIGINT | ss_usr_cpu_time - Tiempo de CPU de usuario utilizado por subsección |
| SS_USR_CPU_TIME_MS | BIGINT | ss_usr_cpu_time - Tiempo de CPU de usuario utilizado por subsección |
| SS_SYS_CPU_TIME_S | BIGINT | ss_sys_cpu_time - Tiempo de CPU de sistema utilizado por subsección |
| SS_SYS_CPU_TIME_MS | BIGINT | ss_sys_cpu_time - Tiempo de CPU de sistema utilizado por subsección |
| SS_NUMBER | INTEGER | ss_number - Número de subsección |
| SS_STATUS | VARCHAR(20) | ss_status - Estado de subsección. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • EXEC • TQ_WAIT_TO_RCV • TQ_WAIT_TO_SEND • COMPLETED |
| SS_NODE_NUMBER | SMALLINT | ss_node_number - Número de nodo de subsección |
| TQ_NODE_WAITED_FOR | SMALLINT | tq_node_waited_for - Nodo esperado en una cola de tabla |
| TQ_WAIT_FOR_ANY | INTEGER | tq_wait_for_any - En espera de cualquier nodo a enviar a una cola de tabla |
| TQ_ID_WAITING_ON | INTEGER | tq_id_waiting_on - Nodo en que se ha esperado en una cola de tabla |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa SNAPSWITCHES y función de tabla SNAP_GET_SWITCHES - Recuperar información de estado de conmutador de instantáneas de base de datos

La vista administrativa SNAPSWITCHES y la función de tabla SNAP_GET_SWITCHES devuelven información sobre el estado del conmutador de instantánea de base de datos.

SNAPSWITCHES, vista administrativa

Esta vista proporciona datos que son equivalentes a los que proporciona el mandato CLP GET DBM MONITOR SWITCHES.

El esquema es SYSIBMADM.

Consulte la Tabla 156 en la página 627 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPSWITCHES
- Privilegio CONTROL en la vista administrativa SNAPSWITCHES
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE en la función de tabla SNAP_GET_SWITCHES
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMMAINT
- SYSADM

Ejemplo

Recuperar información acerca del estado de los conmutadores del supervisor DBM para todas las particiones de base de datos.

```
SELECT UOW_SW_STATE, STATEMENT_SW_STATE, TABLE_SW_STATE, BUFFPOOL_SW_STATE,  
       LOCK_SW_STATE, SORT_SW_STATE, TIMESTAMP_SW_STATE,  
       DBPARTITIONNUM FROM SYSIBMADM.SNAPSWITCHES
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
UOW_SW_STATE STATEMENT_SW_STATE TABLE_SW_STATE BUFFPOOL_SW_STATE ...  
-----  
           0                0                0                0 ...  
           0                0                0                0 ...  
           0                0                0                0 ...  
                                     ...
```

3 registros seleccionados.

Datos de salida de esta consulta (continuación).

| ... | LOCK_SW_STATE | SORT_SW_STATE | TIMESTAMP_SW_STATE | DBPARTITIONNUM |
|-----|---------------|---------------|--------------------|----------------|
| ... | 1 | 0 | 1 | 0 |
| ... | 1 | 0 | 1 | 1 |
| ... | 1 | 0 | 1 | 2 |

SNAP_GET_SWITCHES, función de tabla

La función de tabla SNAP_GET_SWITCHES devuelve la misma información que la vista administrativa SNAPSWITCHES, pero le permite recuperar la información para una partición de base de datos específica, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

Esta función de tabla proporciona datos que son equivalentes a los que proporciona el mandato CLP GET DBM MONITOR SWITCHES.

Consulte la Tabla 156 en la página 627 para ver una lista completa de la información que se puede devolver.

Sintaxis

```

▶▶ SNAP_GET_SWITCHES ( (      ) )
                        [núm-partición-bd]

```

El esquema es SYSPROC.

Parámetro de la función de tabla

núm-partición-bd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si no se utiliza esta opción de entrada, se devolverán datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *núm-partición-bd* se establece en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_SWITCHES tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE en la función de tabla SNAP_GET_SWITCHES
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON

- SYSCTRL
- SYSMANT
- SYSADM

Ejemplos

Recuperar información acerca del estado de los conmutadores del supervisor DBM para la partición de base de datos actual.

```
SELECT UOW_SW_STATE, STATEMENT_SW_STATE, TABLE_SW_STATE,
       BUFFPOOL_SW_STATE, LOCK_SW_STATE, SORT_SW_STATE, TIMESTAMP_SW_STATE
FROM TABLE(SNAP_GET_SWITCHES(-1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
UOW_SW_STATE STATEMENT_SW_STATE TABLE_SW_STATE...
-----
                1                1                1...
                ...
```

1 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```
... BUFFPOOL_SW_STATE LOCK_SW_STATE SORT_SW_STATE TIMESTAMP_SW_STATE
... -----
...                1                1                0                1
```

Información devuelta

Tabla 156. Información que devuelve la vista administrativa SNAPSWITCHES y la función de tabla SNAP_GET_SWITCHES

| Nombre de columna | Tipo de datos | Descripción |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| UOW_SW_STATE | SMALLINT | Estado del conmutador de registro del supervisor de unidad de trabajo (0 ó 1). |
| UOW_SW_TIME | TIMESTAMP | Si el conmutador de registro del supervisor de unidad de trabajo está activado, la fecha y la hora a la que se activó este conmutador. |
| STATEMENT_SW_STATE | SMALLINT | Estado del conmutador de registro del supervisor de sentencia de SQL (0 ó 1). |
| STATEMENT_SW_TIME | TIMESTAMP | Si el conmutador de registro del supervisor de sentencia de SQL está activado, la fecha y la hora a la que se activó este conmutador. |
| TABLE_SW_STATE | SMALLINT | Estado del conmutador de registro del supervisor de actividad de tabla (0 ó 1). |
| TABLE_SW_TIME | TIMESTAMP | Si el conmutador de registro del supervisor de actividad de tabla está activado, la fecha y la hora a la que se activó este conmutador. |

Tabla 156. Información que devuelve la vista administrativa SNAPSWITCHES y la función de tabla SNAP_GET_SWITCHES (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|--------------------|---------------|---|
| BUFFPOOL_SW_STATE | SMALLINT | Estado del conmutador de registro del supervisor de actividad de agrupación de almacenamientos intermedios (0 ó 1). |
| BUFFPOOL_SW_TIME | TIMESTAMP | Si el conmutador de registro del supervisor de actividad de agrupación de almacenamientos intermedios está activado, la fecha y la hora a la que se activó este conmutador. |
| LOCK_SW_STATE | SMALLINT | Estado del conmutador de registro del supervisor de bloqueo (0 ó 1). |
| LOCK_SW_TIME | TIMESTAMP | Si el conmutador de registro del supervisor de bloqueo está activado, la fecha y la hora a la que se activó este conmutador. |
| SORT_SW_STATE | SMALLINT | Estado del conmutador de registro del supervisor de clasificación (0 ó 1). |
| SORT_SW_TIME | TIMESTAMP | Si el conmutador de registro del supervisor de clasificación está activado, la fecha y la hora a la que se activó este conmutador. |
| TIMESTAMP_SW_STATE | SMALLINT | Estado del conmutador de registro del supervisor de indicación de fecha y hora (0 ó 1) |
| TIMESTAMP_SW_TIME | TIMESTAMP | Si el conmutador de registro del supervisor de indicación de fecha y hora está activado, la fecha y la hora a la que se activó este conmutador. |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa SNAPTAB y función de tabla SNAP_GET_TAB_V91 – Recuperar información de instantáneas del grupo de datos lógicos table

La vista administrativa SNAPTAB y la función de tabla SNAP_GET_TAB_V91 devuelven información de la instantánea del grupo de datos lógicos table.

SNAPTAB, vista administrativa

Esta vista administrativa permite recuperar la información de instantáneas del grupo de datos lógicos table para la base de datos conectada actualmente.

La vista administrativa SNAPTAB, utilizada con la vista administrativa SNAPTAB_REORG, devuelve información equivalente a la que proporciona el mandato CLP GET SNAPSHOT FOR TABLES ON alias de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 157 en la página 631 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPTAB
- Privilegio CONTROL en la vista administrativa SNAPTAB
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_TAB_V91
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar el esquema y el nombre de todas las tablas activas.

```
SELECT SUBSTR(TABSHEMA,1,8), SUBSTR(TABNAME,1,15) AS TABNAME, TAB_TYPE,  
       DBPARTITIONNUM FROM SYSIBMADM.SNAPTAB
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| TABSHEMA | TABNAME | TAB_TYPE | DBPARTITIONNUM |
|----------|---------------|------------|----------------|
| SYSTOOLS | HMON_ATM_INFO | USER_TABLE | 0 |

1 registro(s) seleccionado(s).

SNAP_GET_TAB_V91, función de tabla

La función de tabla SNAP_GET_TAB_V91 devuelve la misma información que la vista administrativa SNAPTAB, pero le permite recuperar la información para una base de datos específica de una partición de base de datos específica, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_TAB_V91, utilizada con la función de tabla SNAP_GET_TAB_REORG, devuelve información equivalente a la que proporciona el mandato CLP GET SNAPSHOT FOR TABLES ON alias de base de datos.

Consulte la Tabla 157 en la página 631 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
▶▶ SNAP_GET_TAB_V91 (—nombrebd— [ , númparticiónbd ] )▶▶
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique NULL o una serie vacía para tomar la instantánea de la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_TAB_V91 tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_TAB_V91
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar una lista de tablas activas como vista de agregación para la base de datos conectada actualmente.

```
SELECT SUBSTR(TABSHEMA,1,8) AS TABSHEMA, SUBSTR(TABNAME,1,15) AS TABNAME,
TAB_TYPE, DBPARTITIONNUM FROM TABLE(SNAP_GET_TAB(' ', -2)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| TABSHEMA | TABNAME | TAB_TYPE | DBPARTITIONNUM |
|----------|---------------|------------|----------------|
| SYSTOOLS | HMON_ATM_INFO | USER_TABLE | - |
| JESSICAE | EMPLOYEE | USER_TABLE | - |

Información devuelta

Tabla 157. Información devuelta por la vista administrativa SNAPTAB y la función de tabla SNAP_GET_TAB_V91

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| TABSHEMA | VARCHAR(128) | table_schema - Nombre de esquema de tabla |
| TABNAME | VARCHAR(128) | table_name - Nombre de tabla |
| TAB_FILE_ID | BIGINT | table_file_id - Identificación de archivo de tabla |
| TAB_TYPE | VARCHAR(14) | table_type - Tipo de tabla. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • USER_TABLE • DROPPED_TABLE • TEMP_TABLE • CATALOG_TABLE • REORG_TABLE |
| DATA_OBJECT_PAGES | BIGINT | data_object_pages - Páginas de objeto de datos |
| INDEX_OBJECT_PAGES | BIGINT | index_object_pages - Páginas de objeto de índice |
| LOB_OBJECT_PAGES | BIGINT | lob_object_pages - Páginas de objeto LOB |
| LONG_OBJECT_PAGES | BIGINT | long_object_pages - Páginas de objeto largo |
| XDA_OBJECT_PAGES | BIGINT | xda_object_pages - Páginas de objeto XDA |
| ROWS_READ | BIGINT | rows_read - Filas leídas |
| ROWS_WRITTEN | BIGINT | rows_written - Filas grabadas |
| OVERFLOW_ACCESSES | BIGINT | overflow_accesses - Accesos a registros desbordados |
| PAGE_REORGS | BIGINT | page_reorgs - Reorganizaciones de página |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Tabla 157. Información devuelta por la vista administrativa SNAPTAB y la función de tabla SNAP_GET_TAB_V91 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|--|
| TBSP_ID | BIGINT | tablespace_id - Identificación de espacio de tablas |
| DATA_PARTITION_ID | INTEGER | data_partition_id - Identificador de la partición de datos. Para una tabla no particionada, este elemento será NULL. |

Vista administrativa SNAPTAB_REORG y función de tabla SNAP_GET_TAB_REORG - Recuperar información sobre instantáneas de reorganización de tabla

La vista administrativa SNAPTAB_REORG y la tabla de función SNAP_GET_TAB_REORG devuelven información acerca de la reorganización de tabla. Si no se ha reorganizado ninguna tabla, se devolverán 0 filas.

SNAPTAB_REORG, vista administrativa

Esta vista administrativa le permite recuperar información acerca de la instantánea de reorganización de tabla para la base de datos conectada actualmente.

La vista administrativa SNAPTAB_REORG, utilizada con la vista administrativa SNAPTAB, proporciona datos que son equivalentes a los que proporciona el mandato CLP GET SNAPSHOT FOR TABLES ON alias de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 158 en la página 635 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPTAB_REORG
- Privilegio CONTROL en la vista administrativa SNAPTAB_REORG
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_TAB_REORG
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Seleccionar información detallada acerca de las operaciones de reorganización para todas las particiones de base de datos de la base de datos conectada actualmente.

```
SELECT SUBSTR(TABNAME, 1, 15) AS TAB_NAME, SUBSTR(TABSHEMA, 1, 15)
      AS TAB_SCHEMA, REORG_PHASE, SUBSTR(REORG_TYPE, 1, 20) AS REORG_TYPE,
      REORG_STATUS, REORG_COMPLETION, DBPARTITIONNUM
FROM SYSIBMADM.SNAPTAB_REORG ORDER BY DBPARTITIONNUM
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| TAB_NAME | TAB_SCHEMA | REORG_PHASE | ... |
|----------|------------|-------------|-----|
| EMPLOYEE | DBUSER | REPLACE | ... |
| EMPLOYEE | DBUSER | REPLACE | ... |
| EMPLOYEE | DBUSER | REPLACE | ... |
| | | | ... |

3 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

| ... | REORG_TYPE | REORG_STATUS | REORG_COMPLETION | DBPARTITIONNUM |
|-----|----------------------|--------------|------------------|----------------|
| ... | RECLAIM+OFFLINE+ALLO | COMPLETED | SUCCESS | 0 |
| ... | RECLAIM+OFFLINE+ALLO | COMPLETED | SUCCESS | 1 |
| ... | RECLAIM+OFFLINE+ALLO | COMPLETED | SUCCESS | 2 |

Seleccionar toda la información sobre una operación de reorganización para reclamar extensiones de una tabla de clúster multidimensional (MDC) desde la vista administrativa SNAPTAB_REORG.

```
db2 -v "select * from sysibmadm.snaptab_reorg"
```

| TABNAME | REORG_PHASE | REORG_MAX_PHASE | REORG_TYPE |
|---------|-------------|-----------------|-----------------------------|
| T1 | RELEASE | 3 | RECLAIM_EXTENTS+ALLOW_WRITE |

| REORG_STATUS | REORG_COMPLETION | REORG_START | REORG_END |
|--------------|------------------|----------------------------|----------------------------|
| COMPLETED | SUCCESS | 2008-09-24-14.35.30.734741 | 2008-09-24-14.35.31.460674 |

Función de tabla SNAP_GET_TAB_REORG

La función de tabla SNAP_GET_TAB_REORG devuelve la misma información que la vista administrativa SNAPTAB_REORG, pero le permite recuperar la información para una base de datos específica de una partición de base de datos específica, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_TAB_REORG, utilizada con la función de tabla SNAP_GET_TAB, proporciona datos que son equivalentes a los que proporciona el mandato CLP GET SNAPSHOT FOR TABLES ON alias de base de datos.

Consulte la Tabla 158 en la página 635 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
▶▶ SNAP_GET_TAB_REORG ( (—nombrebd—) (—, númparticiónbd—) ) ▶▶
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique NULL o una serie vacía para tomar la instantánea de la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número de partición de base de datos válido. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_TAB_REORG tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_TAB_REORG
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Seleccionar información detallada acerca de las operaciones de reorganización para la partición de base de datos 1 de la base de datos conectada actualmente.

```
SELECT SUBSTR(TABNAME, 1, 15) AS TAB_NAME, SUBSTR(TABSHEMA, 1, 15)
AS TAB_SCHEMA, REORG_PHASE, SUBSTR(REORG_TYPE, 1, 20) AS REORG_TYPE,
REORG_STATUS, REORG_COMPLETION, DBPARTITIONNUM
FROM TABLE( SNAP_GET_TAB_REORG('', 1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```

TAB_NAME      TAB_SCHEMA    REORG_PHASE    REORG_TYPE      ...
-----
EMPLOYEE      DBUSER        REPLACE        RECLAIM+OFFLINE+ALLO ...
1 registro(s) seleccionado(s).

```

Datos de salida de esta consulta (continuación).

```

... REORG_STATUS REORG_COMPLETION DBPARTITIONNUM
... -----
... COMPLETED   SUCCESS                               1
...

```

Seleccionar toda la información sobre una operación de reorganización para reclamar extensiones de una tabla de clúster multidimensional (MDC) utilizando la función de tabla SNAP_GET_TAB_REORG.

```
db2 -v "select * from table(snap_get_tab_reorg(''))"
```

```

TABNAME  REORG_PHASE    REORG_MAX_PHASE  REORG_TYPE
-----
T1       RELEASE        3                RECLAIM_EXTENTS+ALLOW_WRITE

REORG_STATUS REORG_COMPLETION REORG_START          REORG_END
-----
COMPLETED   SUCCESS          2008-09-24-14.35.30.734741 2008-09-24-14.35.31.460674

```

Información devuelta

Tabla 158. Información devuelta por la vista administrativa SNAPTAB_REORG y la función de tabla SNAP_GET_TAB_REORG

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| TABNAME | VARCHAR(128) | table_name - Nombre de tabla |
| TABSCHEMA | VARCHAR(128) | table_schema - Nombre de esquema de tabla |
| PAGE_REORGS | BIGINT | page_reorgs - Reorganizaciones de página |
| REORG_PHASE | VARCHAR(16) | reorg_phase - Fase de reorganización de tabla. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> BUILD DICT_SAMPLE INDEX_RECREATE REPLACE SORT SCAN DRAIN RELEASE o SORT+DICT_SAMPLE. |
| REORG_MAX_PHASE | INTEGER | reorg_max_phase - Fase máxima de reorganización de tabla |

Tabla 158. Información devuelta por la vista administrativa SNAPTAB_REORG y la función de tabla SNAP_GET_TAB_REORG (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------|---------------|---|
| REORG_CURRENT_COUNTER | BIGINT | reorg_current_counter - Progreso de reorganización de tabla |
| REORG_MAX_COUNTER | BIGINT | reorg_max_counter - Cantidad total de reorganización de tabla |

Tabla 158. Información devuelta por la vista administrativa SNAPTAB_REORG y la función de tabla SNAP_GET_TAB_REORG (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|------------------|--|
| REORG_TYPE | VARCHAR (128) | <p>reorg_type - Atributos de reorganización de tabla. Esta interfaz devuelve un identificador de texto utilizando una combinación de los siguientes identificadores separados mediante '+':</p> <p>Uno de los siguientes:</p> <ul style="list-style-type: none"> • RECLAIM • RECLUSTER • RECLAIM_EXTS <p>y uno de los siguientes:</p> <ul style="list-style-type: none"> • +OFFLINE • +ONLINE <p>Si se especifica la modalidad de acceso, es uno de los siguientes:</p> <ul style="list-style-type: none"> • +ALLOW_NONE • +ALLOW_READ • +ALLOW_WRITE <p>Si está fuera de línea y con la opción RECLUSTER, uno de los siguientes:</p> <ul style="list-style-type: none"> • +INDEXSCAN • +TABLESCAN <p>Si está fuera de línea, uno de los siguientes:</p> <ul style="list-style-type: none"> • +LONGLOB • +DATAONLY <p>Si está fuera de línea, y se especifica una opción, uno de los siguientes:</p> <ul style="list-style-type: none"> • +CHOOSE_TEMP • +KEEPDICTIONARY • +RESETDICTIONARY <p>Si está en línea, y se especifica una opción:</p> <ul style="list-style-type: none"> • +NOTRUNCATE <p>Ejemplo 1: Si se ha ejecutado REORG TABLE TEST.EMPLOYEE, se visualizaría lo siguiente: RECLAIM+OFFLINE+ALLOW_READ+DATAONLY+KEEPDICTIONARY</p> <p>Ejemplo 2: Si se ha ejecutado REORG TABLE TEST.EMPLOYEE INDEX EMPIDX INDEXSCAN, se visualizaría lo siguiente: RECLUSTER+OFFLINE+ALLOW_READ+INDEXSCAN+DATAONLY+KEEPDICTIONARY</p> |

Tabla 158. Información devuelta por la vista administrativa SNAPTAB_REORG y la función de tabla SNAP_GET_TAB_REORG (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|----------------------|---------------|--|
| REORG_STATUS | VARCHAR (10) | reorg_status - Estado de reorganización de tabla. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • COMPLETED • PAUSED • STARTED • STOPPED • TRUNCATE |
| REORG_COMPLETION | VARCHAR (10) | reorg_completion - Distintivo de finalización de reorganización de tabla. Esta interfaz devuelve un identificador de texto, basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • FAIL • SUCCESS |
| REORG_START | TIMESTAMP | reorg_start - Hora de inicio de reorganización de tabla |
| REORG_END | TIMESTAMP | reorg_end - Hora de finalización de reorganización de tabla |
| REORG_PHASE_START | TIMESTAMP | reorg_phase_start - Hora de inicio de fase de reorganización de tabla |
| REORG_INDEX_ID | BIGINT | reorg_index_id - Índice que se utiliza para reorganizar la tabla |
| REORG_TBSPC_ID | BIGINT | reorg_tbspc_id - Espacio de tablas donde se reorganiza la tabla |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |
| DATA_PARTITION_ID | INTEGER | data_partition_id - Identificador de la partición de datos. Para una tabla no particionada, este elemento será NULL. |
| REORG_ROWSCOMPRESSED | BIGINT | reorg_rows_compressed - Filas comprimidas |
| REORG_ROWSREJECTED | BIGINT | reorg_rows_rejected_for_compression - Filas rechazadas para compresión |
| REORG_LONG_TBSPC_ID | BIGINT | reorg_long_tbspc_id - Espacio de tabla donde se reorganizan objetos grandes |

Vista administrativa SNAPTBSP y función de tabla SNAP_GET_TBSP_V91 – Recuperar información de instantáneas del grupo de datos lógicos de espacio de tablas

La vista administrativa SNAPTBSP y la función de tabla SNAP_GET_TBSP_V91 devuelven información de la instantánea del grupo de datos lógicos de espacio de tablas.

SNAPTBSP, vista administrativa

Esta vista administrativa permite recuperar la información de instantáneas del grupo de datos lógicos de espacio de tablas para la base de datos conectada actualmente.

La vista administrativa SNAPTBSP, utilizada con las vistas administrativas SNAPTBSP_PART, SNAPTBSP QUIESCER, SNAPTBSP_RANGE y SNAPCONTAINER, devuelve información equivalente a la que proporciona el mandato CLP GET SNAPSHOT FOR TABLESPACES ON alias de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 159 en la página 641 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPTBSP
- Privilegio CONTROL en la vista administrativa SNAPTBSP
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_TBSP_V91
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMAINT
- SYSADM

Ejemplo

Recuperar una lista de los espacios de tablas de la partición de base de datos de catálogo para la base de datos conectada actualmente.

```
SELECT SUBSTR(TBSP_NAME,1,30) AS TBSP_NAME, TBSP_ID, TBSP_TYPE,  
       TBSP_CONTENT_TYPE FROM SYSIBMADM.SNAPTBSP WHERE DBPARTITIONNUM = 1
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| TBSP_NAME | TBSP_ID | TBSP_TYPE | TBSP_CONTENT_TYPE |
|------------|---------|-----------|-------------------|
| TEMPSPACE1 | 1 | SMS | SYSTEMP |
| USERSPACE1 | 2 | DMS | LONG |

2 registro(s) seleccionado(s).

SNAP_GET_TBSP_V91, función de tabla

La función de tabla SNAP_GET_TBSP_V91 devuelve la misma información que la vista administrativa SNAPTbsp, pero le permite recuperar la información para una base de datos específica de una partición de base de datos específica, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_TBSP_V91, utilizada con las funciones de tabla SNAP_GET_TBSP_PART_V91, SNAP_GET_TBSP QUIESCER, SNAP_GET_TBSP_RANGE y SNAP_GET_CONTAINER_V91, devuelve información que es equivalente a la que proporciona el mandato CLP GET SNAPSHOT FOR TABLESPACES ON alias de base de datos.

Consulte la Tabla 159 en la página 641 para ver una lista completa de la información que se puede devolver.

Sintaxis

```

▶▶ SNAP_GET_TBSP_V91 (—nombrebd— [ , númparticiónbd ] )

```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique NULL o una serie vacía para tomar la instantánea de la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_TBSP_V91 tomará una instantánea del número de partición de base de

datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_TBSP_V91
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMAINT
- SYSADM

Ejemplo

Recuperar una lista de los espacios de tablas de todas las particiones de base de datos para la base de datos conectada actualmente.

```
SELECT SUBSTR(TBSP_NAME,1,10) AS TBSP_NAME, TBSP_ID, TBSP_TYPE,
       TBSP_CONTENT_TYPE, DBPARTITIONNUM FROM TABLE(SNAP_GET_TBSP_V91('')) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| TBSP_NAME | TBSP_ID | TBSP_TYPE | TBSP_CONTENT_TYPE | DBPARTITIONNUM |
|------------|---------|-----------|-------------------|----------------|
| TEMPSPACE1 | 1 | SMS | SYSTEMP | 1 |
| USERSPACE1 | 2 | DMS | LONG | 1 |
| SYSCATSPAC | 0 | DMS | ANY | 0 |
| TEMPSPACE1 | 1 | SMS | SYSTEMP | 0 |
| USERSPACE1 | 2 | DMS | LONG | 0 |
| SYSTOOLSPA | 3 | DMS | LONG | 0 |
| TEMPSPACE1 | 1 | SMS | SYSTEMP | 2 |
| USERSPACE1 | 2 | DMS | LONG | 2 |

8 registro(s) seleccionado(s).

Información devuelta

Tabla 159. Información devuelta por la vista administrativa SNAPTBSP y la función de tabla SNAP_GET_TBSP_V91

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| TBSP_NAME | VARCHAR(128) | tablespace_name - Nombre de espacio de tablas |
| TBSP_ID | BIGINT | tablespace_id - Identificación de espacio de tablas |
| TBSP_TYPE | VARCHAR(10) | tablespace_type - Tipo de espacio de tablas. Esta interfaz devuelve un identificador de texto basado en sentencias define en sqlutil.h y es uno de los siguientes: <ul style="list-style-type: none"> • DMS • SMS |

Tabla 159. Información devuelta por la vista administrativa SNAPTBSP y la función de tabla SNAP_GET_TBSP_V91 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------|---------------|---|
| TBSP_CONTENT_TYPE | VARCHAR(10) | tablespace_content_type - Tipo de contenido de espacio de tablas. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • ANY • LARGE • SYSTEMP • USRTEMP |
| TBSP_PAGE_SIZE | BIGINT | tablespace_page_size - Tamaño de página de espacio de tablas |
| TBSP_EXTENT_SIZE | BIGINT | tablespace_extent_size - Tamaño de extensión de espacio de tablas |
| TBSP_PREFETCH_SIZE | BIGINT | tablespace_prefetch_size - Tamaño de captación previa de espacio de tablas |
| TBSP_CUR_POOL_ID | BIGINT | tablespace_cur_pool_id - Agrupación de almacenamiento intermedio utilizada actualmente |
| TBSP_NEXT_POOL_ID | BIGINT | tablespace_next_pool_id - Agrupación de almacenamiento intermedio que se utilizará en el siguiente arranque |
| FS_CACHING | SMALLINT | fs_caching - Antememoria del sistema de archivos |
| POOL_DATA_L_READS | BIGINT | pool_data_l_reads - Lecturas lógicas de datos de la agrupación de almacenamientos intermedios |
| POOL_DATA_P_READS | BIGINT | pool_data_p_reads - Lecturas físicas de datos de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_L_READS | BIGINT | pool_temp_data_l_reads - Lecturas lógicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_P_READS | BIGINT | pool_temp_data_p_reads - Lecturas físicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_READS | BIGINT | pool_async_data_reads - Lecturas de datos asíncronas de agrupación de almacenamientos intermedios |
| POOL_DATA_WRITES | BIGINT | pool_data_writes - Grabaciones de datos de la agrupación de almacenamientos intermedios |

Tabla 159. Información devuelta por la vista administrativa SNAPTBSP y la función de tabla SNAP_GET_TBSP_V91 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|---|
| POOL_ASYNC_DATA_WRITES | BIGINT | pool_async_data_writes - Grabaciones de datos asíncronas de agrupación de almacenamientos intermedios |
| POOL_INDEX_L_READS | BIGINT | pool_index_l_reads - Lecturas lógicas de índices de la agrupación de almacenamientos intermedios |
| POOL_INDEX_P_READS | BIGINT | pool_index_p_reads - Lecturas físicas de índices de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_L_READS | BIGINT | pool_temp_index_l_reads - Lecturas lógicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_P_READS | BIGINT | pool_temp_index_p_reads - Lecturas físicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_READS | BIGINT | pool_async_index_reads - Lecturas de índice asíncronas de agrupación de almacenamientos intermedios |
| POOL_INDEX_WRITES | BIGINT | pool_index_writes - Grabaciones de índices de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_WRITES | BIGINT | pool_async_index_writes - Grabaciones de índice asíncronas de agrupación de almacenamientos intermedios |
| POOL_XDA_L_READS | BIGINT | pool_xda_l_reads - Lecturas lógicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_XDA_P_READS | BIGINT | pool_xda_p_reads - Lecturas físicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_XDA_WRITES | BIGINT | pool_xda_writes - Grabaciones de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_XDA_READS | BIGINT | pool_async_xda_reads - Lecturas de datos XDA asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_XDA_WRITES | BIGINT | pool_async_xda_writes - Grabaciones de datos XDA asíncronas de agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_L_READS | BIGINT | pool_temp_xda_l_reads - Lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |

Tabla 159. Información devuelta por la vista administrativa SNAPT BSP y la función de tabla SNAP_GET_TBSP_V91 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|----------------------------|---------------|--|
| POOL_TEMP_XDA_P_READS | BIGINT | pool_temp_xda_p_reads - Elemento del supervisor de las lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |
| POOL_READ_TIME | BIGINT | pool_read_time - Tiempo total de lectura física de la agrupación de almacenamientos intermedios |
| POOL_WRITE_TIME | BIGINT | pool_write_time - Tiempo total de grabación física de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_READ_TIME | BIGINT | pool_async_read_time - Tiempo de lectura asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_WRITE_TIME | BIGINT | pool_async_write_time - Tiempo de grabación asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_READ_REQS | BIGINT | pool_async_data_read_reqs - Peticiones de lectura asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_READ_REQS | BIGINT | pool_async_index_read_reqs - Peticiones de lectura de índice asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_XDA_READ_REQS | BIGINT | pool_async_xda_read_reqs - Peticiones de lectura XDA asíncrona de agrupación de almacenamientos intermedios |
| POOL_NO_VICTIM_BUFFER | BIGINT | pool_no_victim_buffer - Almacenamiento intermedio sin víctimas de la agrupación de almacenamientos intermedios |
| DIRECT_READS | BIGINT | direct_reads - Lecturas directas de la base de datos |
| DIRECT_WRITES | BIGINT | direct_writes - Grabaciones directas en la base de datos |
| DIRECT_READ_REQS | BIGINT | direct_read_reqs - Peticiones de lectura directa |
| DIRECT_WRITE_REQS | BIGINT | direct_write_reqs - Peticiones de grabación directa |
| DIRECT_READ_TIME | BIGINT | direct_read_time - Tiempo de lectura directa |
| DIRECT_WRITE_TIME | BIGINT | direct_write_time - Tiempo de grabación directa |
| FILES_CLOSED | BIGINT | files_closed - Archivos de base de datos cerrados |

Tabla 159. Información devuelta por la vista administrativa SNAPTbsp y la función de tabla SNAP_GET_TBSP_V91 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------------|---------------|--|
| UNREAD_PREFETCH_PAGES | BIGINT | unread_prefetch_pages - Páginas de captación previa no leídas |
| TBSP_REBALANCER_MODE | VARCHAR(10) | tablespace_rebalancer_mode - Modalidad de reequilibrador. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h, y es uno de los siguientes: <ul style="list-style-type: none"> • NO_REBAL • FWD_REBAL • REV_REBAL |
| TBSP_USING_AUTO_STORAGE | SMALLINT | tablespace_using_auto_storage - Espacio de tablas habilitado para el almacenamiento automático |
| TBSP_AUTO_RESIZE_ENABLED | SMALLINT | tablespace_auto_resize_enabled - Espacio de tablas habilitado para la modificación de tamaño automática |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa SNAPTbsp_PART y función de tabla SNAP_GET_TBSP_PART_V97 – Recuperar información de instantáneas del grupo de datos lógicos tablespace_nodeinfo

La vista administrativa SNAPTbsp_PART y la función de tabla SNAP_GET_TBSP_PART_V97 devuelven información de la instantánea del grupo de datos lógicos tablespace_nodeinfo.

Vista administrativa SNAPTbsp_PART

Esta vista administrativa permite recuperar la información de instantáneas del grupo de datos lógicos tablespace_nodeinfo para la base de datos conectada actualmente.

La vista administrativa SNAPTbsp_PART, utilizada con las vistas administrativas SNAPTbsp, SNAPTbsp_QUIESCER, SNAPTbsp_RANGE y SNAPCONTAINER, devuelve información equivalente a la que proporciona el mandato CLP GET SNAPSHOT FOR TABLESPACES ON alias de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 160 en la página 647 para ver una lista completa de la información que se puede devolver.

Autorización

- Autorización SYSMON

- Privilegio SELECT o CONTROL para la vista administrativa SNAPTBSP_PART y privilegio EXECUTE para la función de tabla SNAP_GET_TBSP_PART_V97.

Ejemplo

Recuperar una lista de los espacios de tablas, y el estado de éstos, de todas las particiones de base de datos de la base de datos conectada actualmente.

```
SELECT SUBSTR(TBSP_NAME,1,30) AS TBSP_NAME, TBSP_ID,
       SUBSTR(TBSP_STATE,1,30) AS TBSP_STATE, DBPARTITIONNUM
FROM SYSIBMADM.SNAPTbsp_PART
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| TBSP_NAME | TBSP_ID | TBSP_STATE | DBPARTITIONNUM |
|-------------|---------|------------|----------------|
| SYSCATSPACE | 0 | NORMAL | 0 |
| TEMPSPACE1 | 1 | NORMAL | 0 |
| USERSPACE1 | 2 | NORMAL | 0 |
| TEMPSPACE1 | 1 | NORMAL | 1 |
| USERSPACE1 | 2 | NORMAL | 1 |

5 registro(s) seleccionado(s).

SNAP_GET_TBSP_PART_V97, función de tabla

La función de tabla SNAP_GET_TBSP_PART_V97 devuelve la misma información que la vista administrativa SNAPTBSP_PART, pero le permite recuperar la información para una base de datos específica de una partición de base de datos específica, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_TBSP_PART_V97, utilizada con las funciones de tabla SNAP_GET_TBSP_V97, SNAP_GET_TBSP_QUIESCER y SNAP_GET_TBSP_RANGE, SNAP_GET_CONTAINER_V91, devuelve información que es equivalente a la que proporciona el mandato CLP GET SNAPSHOT FOR TABLESPACES ON alias-base-datos.

Consulte la Tabla 160 en la página 647 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
▶▶ SNAP_GET_TBSP_PART_V97 ( ( nombrebd , numparticiónbd ) ) ▶▶
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique NULL o una serie vacía para tomar la instantánea de la base de datos conectada actualmente.

numparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *numparticiónbd* se establece en NULL, -1 se establece implícitamente para *numparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *numparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_TBSP_PART_V97 tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP_GET_TBSP_PART_V97.

Ejemplo

Recuperar una lista de los espacios de tablas, y el estado de éstos, de la partición de base de datos conectada de la base de datos conectada.

```
SELECT SUBSTR(TBSP_NAME,1,30) AS TBSP_NAME, TBSP_ID,  
       SUBSTR(TBSP_STATE,1,30) AS TBSP_STATE  
FROM TABLE(SNAP_GET_TBSP_PART_V97(CAST(NULL AS VARCHAR(128)),-1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| TBSP_NAME | TBSP_ID | TBSP_STATE |
|------------------|---------|------------|
| SYSCATSPACE | | 0 NORMAL |
| TEMPSPACE1 | | 1 NORMAL |
| USERSPACE1 | | 2 NORMAL |
| SYSTOOLSPACE | | 3 NORMAL |
| SYSTOOLSTMPSPACE | | 4 NORMAL |

5 registro(s) seleccionado(s).

Información devuelta

Tabla 160. Información devuelta por la vista administrativa SNAP_TBSP_PART y la función de tabla SNAP_GET_TBSP_PART_V97

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| TBSP_NAME | VARCHAR(128) | tablespace_name - Nombre de espacio de tablas |
| TBSP_ID | BIGINT | tablespace_id - Identificación de espacio de tablas |

Tabla 160. Información devuelta por la vista administrativa SNAPTBSP_PART y la función de tabla SNAP_GET_TBSP_PART_V97 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------------|------------------|--|
| TBSP_STATE | VARCHAR (256) | <p>tablespace_state - Estado de espacio de tablas. Esta interfaz devuelve un identificador de texto basado en las sentencias define en sqlutil.h y es una combinación de los siguientes elementos separados por un signo '+':</p> <ul style="list-style-type: none"> • BACKUP_IN_PROGRESS • BACKUP_PENDING • DELETE_PENDING • DISABLE_PENDING • DROP_PENDING • LOAD_IN_PROGRESS • LOAD_PENDING • NORMAL • OFFLINE • PSTAT_CREATION • PSTAT_DELETION • QUIESCED_EXCLUSIVE • QUIESCED_SHARE • QUIESCED_UPDATE • REBAL_IN_PROGRESS • REORG_IN_PROGRESS • RESTORE_IN_PROGRESS • RESTORE_PENDING • ROLLFORWARD_IN_PROGRESS • ROLLFORWARD_PENDING • STORDEF_ALLOWED • STORDEF_CHANGED • STORDEF_FINAL_VERSION • STORDEF_PENDING • SUSPEND_WRITE |
| TBSP_PREFETCH_SIZE | BIGINT | tablespace_prefetch_size - Tamaño de captación previa de espacio de tablas |
| TBSP_NUM QUIESCERS | BIGINT | tablespace_num_quiescers - Número de inmovilizadores |
| TBSP_STATE_CHANGE_OBJECT_ID | BIGINT | tablespace_state_change_object_id - Identificación de objeto de cambio de estado |
| TBSP_STATE_CHANGE_TBSP_ID | BIGINT | tablespace_state_change_ts_id - Identificación de espacio de tablas de cambio de estado |
| TBSP_MIN_RECOVERY_TIME | TIMESTAMP | tablespace_min_recovery_time - Tiempo mínimo de recuperación para la recuperación en avance |
| TBSP_TOTAL_PAGES | BIGINT | tablespace_total_pages - Total de páginas en espacio de tablas |

Tabla 160. Información devuelta por la vista administrativa SNAPTBSP_PART y la función de tabla SNAP_GET_TBSP_PART_V97 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------------|---------------|---|
| TBSP_USABLE_PAGES | BIGINT | tablespace_usable_pages - Páginas utilizables de espacio de tablas |
| TBSP_USED_PAGES | BIGINT | tablespace_used_pages - Páginas utilizadas en espacio de tablas |
| TBSP_FREE_PAGES | BIGINT | tablespace_free_pages - Páginas libres en espacio de tablas |
| TBSP_PENDING_FREE_PAGES | BIGINT | tablespace_pending_free_pages - Páginas libres pendientes en espacio de tablas |
| TBSP_PAGE_TOP | BIGINT | tablespace_page_top - Nivel límite de espacio de tablas |
| REBALANCER_MODE | VARCHAR (30) | tablespace_rebalancer_mode - Modalidad de reequilibrador. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • FWD_REBAL • NO_REBAL • REV_REBAL • FWD_REBAL_OF_2PASS • REV_REBAL_OF_2PASS |
| REBALANCER_EXTENTS_REMAINING | BIGINT | tablespace_rebalancer_extents_remaining - Número total de extensiones que el reequilibrador debe procesar |
| REBALANCER_EXTENTS_PROCESSED | BIGINT | tablespace_rebalancer_extents_processed - Número de extensiones que el reequilibrador ha procesado |
| REBALANCER_PRIORITY | BIGINT | tablespace_rebalancer_priority - Prioridad de reequilibrador actual |
| REBALANCER_START_TIME | TIMESTAMP | tablespace_rebalancer_start_time - Hora de inicio de reequilibrador |
| REBALANCER_RESTART_TIME | TIMESTAMP | tablespace_rebalancer_restart_time - Hora de reinicio de reequilibrador |
| REBALANCER_LAST_EXTENT_MOVED | BIGINT | tablespace_rebalancer_last_extents_moved - Última extensión movida por el reequilibrador |
| TBSP_NUM_RANGES | BIGINT | tablespace_num_ranges - Número de rangos de la correlación de espacio de tablas |
| TBSP_NUM_CONTAINERS | BIGINT | tablespace_num_containers - Número de contenedores de espacio de tablas |
| TBSP_INITIAL_SIZE | BIGINT | tablespace_initial_size - Tamaño inicial de espacio de tablas |
| TBSP_CURRENT_SIZE | BIGINT | tablespace_current_size - Tamaño actual de espacio de tablas |

Tabla 160. Información devuelta por la vista administrativa `SNAPTbsp_Part` y la función de tabla `SNAP_Get_Tbsp_Part_V97` (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---|---------------|--|
| <code>Tbsp_Max_Size</code> | BIGINT | <code>tablespace_max_size</code> - Tamaño máximo de espacio de tablas |
| <code>Tbsp_Increase_Size</code> | BIGINT | <code>tablespace_increase_size</code> - Aumento de tamaño en bytes |
| <code>Tbsp_Increase_Size_Percent</code> | SMALLINT | <code>tablespace_increase_size_percent</code> - Aumento de tamaño en porcentaje |
| <code>Tbsp_Last_Resize_Time</code> | TIMESTAMP | <code>tablespace_last_resize_time</code> - Hora del último redimensionamiento satisfactorio |
| <code>Tbsp_Last_Resize_Failed</code> | SMALLINT | <code>tablespace_last_resize_failed</code> - Último intento de redimensionamiento fallido |
| <code>Tbsp_Paths_Dropped</code> | SMALLINT | Indica que el espacio de tablas reside en una o varias vías de almacenamiento que se han descartado (0 - No, 1 - Sí) |
| <code>DBPARTITIONNUM</code> | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa `SNAPTbsp_QUIESCER` y función de tabla `SNAP_Get_Tbsp_QUIESCER` – Recuperar información acerca de la instantánea de espacio de tablas de inmovilizador

La vista administrativa `SNAPTbsp_QUIESCER` y la función de tabla `SNAP_Get_Tbsp_QUIESCER` devuelven información sobre inmovilizadores desde una instantánea del espacio de tablas.

`SNAPTbsp_QUIESCER`, vista administrativa

Esta vista administrativa le permite recuperar información acerca de la instantánea de espacio de tablas de inmovilizador para la base de datos conectada actualmente.

La vista administrativa `SNAPTbsp_QUIESCER`, utilizada con las vistas administrativas `SNAPTbsp`, `SNAPTbsp_Part`, `SNAPTbsp_Range`, `SNAPCONTAINER`, proporciona información que es equivalente a la información que proporciona el mandato `CLP GET SNAPSHOT FOR TABLESPACES ON` alias de base de datos.

El esquema es `SYSIBMADM`.

Consulte la Tabla 161 en la página 654 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio `SELECT` en la vista administrativa `SNAPTbsp_QUIESCER`
- Privilegio `CONTROL` en la vista administrativa `SNAPTbsp_QUIESCER`

- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE para la función de tabla SNAP_GET_TBSP QUIESCER
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar información acerca de los espacios de tablas inmovilizados para todas las particiones de base de datos de la base de datos conectada actualmente.

```
SELECT SUBSTR(TBSP_NAME, 1, 10) AS TBSP_NAME, QUIESCER_TS_ID,
       QUIESCER_OBJ_ID, QUIESCER_AUTH_ID, QUIESCER_AGENT_ID,
       QUIESCER_STATE, DBPARTITIONNUM
FROM SYSIBMADM.SNAPTbsp QUIESCER ORDER BY DBPARTITIONNUM
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| TBSP_NAME | QUIESCER_TS_ID | QUIESCER_OBJ_ID | QUIESCER_AUTH_ID | .. |
|------------|----------------|-----------------|------------------|----|
| USERSPACE1 | 2 | 5 | SWALKTY | .. |
| USERSPACE1 | 2 | 5 | SWALKTY | .. |

2 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

| ... QUIESCER_AGENT_ID | QUIESCER_STATE | DBPARTITIONNUM |
|-----------------------|-----------------|----------------|
| ... | 0 EXCLUSIVE | 0 |
| ... | 65983 EXCLUSIVE | 1 |

Ejemplo: determinar los nombres de las tablas particionadas de rangos

Si la tabla está particionada por rangos se mantiene en estado inmovilizado, los diferentes valores del ID de espacio de tablas y del ID de tabla se representan en SYSCAT.TABLES. Estos ID se mostrarán como descripciones breves sin signo. Para encontrar el nombre de la tabla inmovilizada, quizás necesite encontrar primero la descripción breve con signo calculando para ello el ID de espacio de tablas que resta 65536 (el valor máximo) de QUIESCER_TS_ID y, a continuación, utilizar este ID de espacio de tablas para localizar las tablas inmovilizadas. (El ID de espacio de tablas real se encuentra en SYSCAT.DATAPARTITIONS para cada partición de rango de la tabla).

```
SELECT SUBSTR(TBSP_NAME, 1, 10) AS TBSP_NAME,
       CASE WHEN QUIESCER_TS_ID = 65530
            THEN QUIESCER_TS_ID - 65536
            ELSE QUIESCER_TS_ID END as tbspaceid,
       CASE WHEN QUIESCER_TS_ID = 65530
            THEN QUIESCER_OBJ_ID - 65536
            ELSE QUIESCER_OBJ_ID END as tableid
FROM SYSIBMADM.SNAPTbsp QUIESCER
ORDER BY DBPARTITIONNUM
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| TBSP_NAME | TBSPACEID | TABLEID |
|-----------|-----------|---------|
| TABDATA | -6 | -32768 |
| DATAMART | -6 | -32765 |
| SMALL | 5 | 17 |

3 registro(s) seleccionado(s).

Utilice los valores de TBSPACEID y TABLEID proporcionados por la consulta anterior para encontrar el nombre y el esquema de tabla desde SYSCAT.TABLES.

```
SELECT CHAR(tabschema, 10)tabschema, CHAR(tabname,15)tabname
FROM SYSCAT.TABLES
WHERE tbspaceid = -6 AND tableid in (-32768,-32765)
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| TABSCHEMA | TABNAME |
|-----------|--------------|
| TPCD | ORDERS_RP |
| TPCD | ORDERS_DMART |

2 registro(s) seleccionado(s).

```
SELECT CHAR(tabschema, 10)tabschema, CHAR(tabname,15)tabname
FROM SYSCAT.TABLES
WHERE tbspaceid = 5 AND tableid = 17
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| TABSCHEMA | TABNAME |
|-----------|---------|
| TPCD | NATION |

1 registro(s) seleccionado(s).

Función de tabla SNAP_GET_TBSP QUIESCER

La función de tabla SNAP_GET_TBSP QUIESCER devuelve la misma información que la vista administrativa SNAPTbsp QUIESCER, pero le permite recuperar la información para una base de datos específica de una partición de una base de datos específica, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_TBSP QUIESCER, utilizada con las funciones de tabla SNAP_GET_TBSP_V91, SNAP_GET_TBSP_PART_V91, SNAP_GET_TBSP_RANGE, SNAP_GET_CONTAINER_V91, proporciona información que es equivalente a la información que proporciona el mandato CLP GET SNAPSHOT FOR TABLESPACES ON alias de base de datos.

Consulte la Tabla 161 en la página 654 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
▶▶ SNAP_GET_TBSP QUIESCER (—nombrebd —————) —————▶▶
                        |, númparticiónbd|
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique NULL o una serie vacía para tomar la instantánea de la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_TBSP QUIESCER tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_TBSP QUIESCER
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar información acerca de los espacios de tablas inmovilizados para la partición de base de datos 1 de la base de datos conectada actualmente.

```
SELECT SUBSTR(TBSP_NAME, 1, 10) AS TBSP_NAME, QUIESCER_TS_ID,  
       QUIESCER_OBJ_ID, QUIESCER_AUTH_ID, QUIESCER_AGENT_ID,  
       QUIESCER_STATE, DBPARTITIONNUM  
FROM TABLE( SYSPROC.SNAP_GET_TBSP QUIESCER( ' ', 1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```

Tbsp_name  Quiescer_ts_id  Quiescer_obj_id  Quiescer_auth_id  ...
-----
USERSPACE1                2                5 SWALKTY                ...

```

1 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```

... Quiescer_agent_id  Quiescer_state  DBPARTITIONNUM
... -----
...                65983 EXCLUSIVE                1

```

Información devuelta

Tabla 161. Información devuelta por la vista administrativa `SNAPTbsp_QUIESCER` y la función de tabla `SNAP_GET_Tbsp_QUIESCER`

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| Tbsp_name | VARCHAR(128) | tablespace_name - Nombre de espacio de tablas |
| Quiescer_ts_id | BIGINT | quiescer_ts_id - Identificación del espacio de tablas de inmovilizador |
| Quiescer_obj_id | BIGINT | quiescer_obj_id - Identificación del objeto de inmovilizador |
| Quiescer_auth_id | VARCHAR(128) | quiescer_auth_id - Identificación de autorización de usuario de inmovilizador |
| Quiescer_agent_id | BIGINT | quiescer_agent_id - Identificación de agente de inmovilizador |
| Quiescer_state | VARCHAR(14) | quiescer_state - Estado de inmovilizador. Esta interfaz devuelve un identificador de texto basado en las sentencias define de <code>sqlutil.h</code> y es uno de los siguientes: <ul style="list-style-type: none"> EXCLUSIVE UPDATE SHARE |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa `SNAPTbsp_RANGE` y función de tabla `SNAP_GET_Tbsp_RANGE` – Recuperar información acerca de la instantánea de rango

La vista administrativa `SNAPTbsp_RANGE` y la función de tabla `SNAP_GET_Tbsp_RANGE` devuelven información desde una instantánea de rango.

SNAPTbsp_RANGE, vista administrativa

Esta vista administrativa le permite recuperar información acerca de la instantánea de rango para la base de datos conectada actualmente.

La vista administrativa SNAPTbsp_RANGE, utilizada con las vistas administrativas SNAPTbsp, SNAPTbsp_PART, SNAPTbsp_QUIESCER y SNAPCONTAINER, proporciona información que es equivalente a la información que proporciona el mandato CLP GET SNAPSHOT FOR TABLESPACES ON alias de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 162 en la página 658 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPTbsp_RANGE
- Privilegio CONTROL en la vista administrativa SNAPTbsp_RANGE
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_TBSP_RANGE
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCtrl
- SYSMAINT
- SYSADM

Ejemplo

Seleccionar información acerca de los rangos de espacio de tablas para todas las particiones de la base de datos conectada actualmente.

```
SELECT TBSP_ID, SUBSTR(TBSP_NAME, 1, 15) AS TBSP_NAME, RANGE_NUMBER,  
       RANGE_STRIPE_SET_NUMBER, RANGE_OFFSET, RANGE_MAX_PAGE,  
       RANGE_MAX_EXTENT, RANGE_START_STRIPE, RANGE_END_STRIPE,  
       RANGE_ADJUSTMENT, RANGE_NUM_CONTAINER, RANGE_CONTAINER_ID,  
       DBPARTITIONNUM FROM SYSIBMADM.SNAPTbsp_RANGE  
ORDER BY DBPARTITIONNUM
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| TBSP_ID | TBSP_NAME | RANGE_NUMBER | RANGE_STRIPE_SET_NUMBER | ... |
|---------|--------------|--------------|-------------------------|-----|
| 0 | SYSCATSPACE | 0 | 0 | ... |
| 2 | USERSPACE1 | 0 | 0 | ... |
| 3 | SYSTOOLSPACE | 0 | 0 | ... |
| 2 | USERSPACE1 | 0 | 0 | ... |
| 2 | USERSPACE1 | 0 | 0 | ... |

5 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

| ... | RANGE_OFFSET | RANGE_MAX_PAGE | RANGE_MAX_EXTENT | ... |
|-----|--------------|----------------|------------------|-----|
| ... | 0 | 11515 | 2878 | ... |
| ... | 0 | 479 | 14 | ... |
| ... | 0 | 251 | 62 | ... |
| ... | 0 | 479 | 14 | ... |
| ... | 0 | 479 | 14 | ... |

Datos de salida de esta consulta (continuación).

| ... | RANGE_START_STRIPE | RANGE_END_STRIPE | RANGE_ADJUSTMENT | ... |
|-----|--------------------|------------------|------------------|-----|
| ... | 0 | 2878 | 0 | ... |
| ... | 0 | 14 | 0 | ... |
| ... | 0 | 62 | 0 | ... |
| ... | 0 | 14 | 0 | ... |
| ... | 0 | 14 | 0 | ... |

Datos de salida de esta consulta (continuación).

| ... | RANGE_NUM_CONTAINER | RANGE_CONTAINER_ID | DBPARTITIONNUM |
|-----|---------------------|--------------------|----------------|
| ... | 1 | 0 | 0 |
| ... | 1 | 0 | 0 |
| ... | 1 | 0 | 0 |
| ... | 1 | 0 | 1 |
| ... | 1 | 0 | 2 |

Función de tabla SNAP_GET_TBSP_RANGE

La función de tabla SNAP_GET_TAB_RANGE devuelve la misma información que la vista administrativa SNAPTAB_RANGE, pero le permite recuperar la información para una base de datos específica de una partición de base de datos específica, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_TBSP_RANGE, utilizada con las funciones de tabla SNAP_GET_TBSP_V91, SNAP_GET_TBSP_PART_V91, SNAP_GET_TBSP QUIESCER y SNAP_GET_CONTAINER_V91, proporciona información que es equivalente a la información que proporciona el mandato CLP GET SNAPSHOT FOR TABLESPACES ON alias de base de datos.

Consulte la Tabla 162 en la página 658 para ver una lista completa de la información que se puede devolver.

Sintaxis

```

▶▶ SNAP_GET_TBSP_RANGE ( ( nombrebd [ , númparticiónbd ] ) )

```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST

DATABASE DIRECTORY. Especifique NULL o una serie vacía para tomar la instantánea de la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_TBSP_RANGE tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_TBSP_RANGE
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMANT
- SYSADM

Ejemplos

Seleccionar información acerca del rango de espacio de tablas para el espacio de tablas con *tbasp_id* = 2 de la partición de base de datos conectada actualmente.

```
SELECT TBSP_ID, SUBSTR(TBSP_NAME, 1, 15) AS TBSP_NAME, RANGE_NUMBER,
       RANGE_STRIPE_SET_NUMBER, RANGE_OFFSET, RANGE_MAX_PAGE, RANGE_MAX_EXTENT,
       RANGE_START_STRIPE, RANGE_END_STRIPE, RANGE_ADJUSTMENT,
       RANGE_NUM_CONTAINER, RANGE_CONTAINER_ID
FROM TABLE(SNAP_GET_TBSP_RANGE(' ', -1)) AS T WHERE TBSP_ID = 2
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| TBSP_ID | TBSP_NAME | RANGE_NUMBER | ... |
|---------|------------|--------------|-----|
| 2 | USERSPACE1 | 0 | ... |

1 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

| ... | RANGE_STRIPE_SET_NUMBER | RANGE_OFFSET | RANGE_MAX_PAGE | ... |
|-----|-------------------------|--------------|----------------|-----|
| ... | 0 | 0 | 3967 | ... |

Datos de salida de esta consulta (continuación).

```

... RANGE_MAX_EXTENT    RANGE_START_STRIPE    RANGE_END_STRIPE    ...
... -----
...                    123                            0                    123 ...

```

Datos de salida de esta consulta (continuación).

```

... RANGE_ADJUSTMENT    RANGE_NUM_CONTAINER    RANGE_CONTAINER_ID
... -----
...                    0                            1                    0

```

Información devuelta

Tabla 162. Información devuelta por la vista administrativa SNAPTBSP_RANGE y la función de tabla SNAP_GET_TBSP_RANGE

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| TBSP_ID | BIGINT | tablespace_id - Identificación de espacio de tablas |
| TBSP_NAME | VARCHAR(128) | tablespace_name - Nombre de espacio de tablas |
| RANGE_NUMBER | BIGINT | range_number - Número de rango |
| RANGE_STRIPE_SET_NUMBER | BIGINT | range_stripe_set_number - Número de conjunto de bandas |
| RANGE_OFFSET | BIGINT | range_offset - Desplazamiento de rango |
| RANGE_MAX_PAGE | BIGINT | range_max_page_number - Página máxima en rango |
| RANGE_MAX_EXTENT | BIGINT | range_max_extent - Extensión máxima en rango |
| RANGE_START_STRIPE | BIGINT | range_start_stripe - Banda inicial |
| RANGE_END_STRIPE | BIGINT | range_end_stripe - Banda final |
| RANGE_ADJUSTMENT | BIGINT | range_adjustment - Ajuste de rango |
| RANGE_NUM_CONTAINER | BIGINT | range_num_containers - Número de contenedores en rango |
| RANGE_CONTAINER_ID | BIGINT | range_container_id - Contenedor de rango |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa SNAPUTIL y función de tabla SNAP_GET_UTIL - Recuperar información de instantánea de grupo de datos lógicos utility_info

La vista administrativa SNAPUTIL y la función de tabla SNAP_GET_UTIL devuelven información de instantáneas acerca de los programas de utilidad desde el grupo de datos lógicos utility_info.

SNAPUTIL, vista administrativa

La vista administrativa SNAPUTIL, que se utiliza junto con la vista administrativa SNAPUTIL_PROGRESS, proporciona la misma información que proporciona el mandato CLP LIST UTILITIES SHOW DETAIL.

El esquema es SYSIBMADM.

Consulte la Tabla 163 en la página 661 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPUTIL
- Privilegio CONTROL en la vista administrativa SNAPUTIL
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE para la función de tabla SNAP_GET_UTIL
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMAINT
- SYSADM

Ejemplo

Recuperar una lista de programas de utilidad, y los estados de éstos, de todas las particiones de base de datos para todas las bases de datos activas en la instancia que contiene la base de datos conectada.

```
SELECT UTILITY_TYPE, UTILITY_PRIORITY, SUBSTR(UTILITY_DESCRIPTION, 1, 72)
       AS UTILITY_DESCRIPTION, SUBSTR(UTILITY_DBNAME, 1, 17) AS
       UTILITY_DBNAME, UTILITY_STATE, UTILITY_INVOKER_TYPE, DBPARTITIONNUM
FROM SYSIBMADM.SNAPUTIL ORDER BY DBPARTITIONNUM
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
UTILITY_TYPE    UTILITY_PRIORITY ...
-----
LOAD            - ...
LOAD            - ...
LOAD            - ...
```

3 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```
... UTILITY_DESCRIPTION ...
... -----
... ONLINE LOAD DEL AUTOMATIC INDEXING INSERT COPY NO TEST .LOADTEST ...
... ONLINE LOAD DEL AUTOMATIC INDEXING INSERT COPY NO TEST .LOADTEST ...
... ONLINE LOAD DEL AUTOMATIC INDEXING INSERT COPY NO TEST .LOADTEST ...
```

Datos de salida de esta consulta (continuación).

| UTILITY_DBNAME | UTILITY_STATE | UTILITY_INVOKER_TYPE | DBPARTITIONNUM |
|----------------|---------------|----------------------|----------------|
| SAMPLE | EXECUTE | USER | 0 |
| SAMPLE | EXECUTE | USER | 1 |
| SAMPLE | EXECUTE | USER | 2 |

SNAP_GET_UTIL, función de tabla

La función de tabla SNAP_GET_UTIL devuelve la misma información que la vista administrativa SNAPUTIL, pero le permite recuperar la información para una partición de base de datos específica, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_UTIL, que se utiliza junto con la función de tabla SNAP_GET_UTIL_PROGRESS, proporciona la misma información que el mandato CLP LIST UTILITIES SHOW DETAIL.

Consulte la Tabla 163 en la página 661 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
▶▶ SNAP_GET_UTIL ( [ núm-partición-bd ] ) ▶▶
```

El esquema es SYSPROC.

Parámetro de la función de tabla

núm-partición-bd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si no se utiliza esta opción de entrada, se devolverán datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *núm-partición-bd* se establece en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_UTIL tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE para la función de tabla SNAP_GET_UTIL
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMAINT
- SYSADM

Ejemplo

Recuperar una lista de los ID de programa de utilidad, con su tipo y estado, para la partición de base de datos conectada actualmente en la base de datos SAMPLE.

```
SELECT UTILITY_ID, UTILITY_TYPE, STATE
FROM TABLE(SNAP_GET_UTIL(-1)) AS T WHERE UTILITY_DBNAME='SAMPLE'
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| UTILITY_ID | UTILITY_TYPE | STATE |
|------------|--------------|---------|
| 1 | BACKUP | EXECUTE |

1 registro(s) seleccionado(s).

Información devuelta

Tabla 163. Información que devuelve la vista administrativa SNAPUTIL y la función de tabla SNAP_GET_UTIL

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| UTILITY_ID | INTEGER | utility_id - ID de programa de utilidad. Exclusivo de una partición de base de datos. |
| UTILITY_TYPE | VARCHAR(26) | utility_type - Tipo de programa de utilidad. Esta interfaz devuelve un identificador de texto basándose en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • ASYNC_INDEX_CLEANUP • BACKUP • CRASH_RECOVERY • LOAD • REBALANCE • REDISTRIBUTE • REORG • RESTART_RECREATE_INDEX • RESTORE • ROLLFORWARD_RECOVERY • RUNSTATS |
| UTILITY_PRIORITY | INTEGER | utility_priority - Prioridad del programa de utilidad. Prioridad si el programa de utilidad da soporte a la disminución; de lo contrario, nulo. |

Tabla 163. Información que devuelve la vista administrativa SNAPUTIL y la función de tabla SNAP_GET_UTIL (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------------|---------------|---|
| UTILITY_DESCRIPTION | VARCHAR(2048) | utility_description - Descripción del programa de utilidad. Puede ser nulo. |
| UTILITY_DBNAME | VARCHAR(128) | utility_dbname - Base de datos en la que trabaja el programa de utilidad |
| UTILITY_START_TIME | TIMESTAMP | utility_start_time - Hora de inicio del programa de utilidad |
| UTILITY_STATE | VARCHAR(10) | utility_state - Estado del programa de utilidad. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • ERROR • EXECUTE • WAIT |
| UTILITY_INVOKER_TYPE | VARCHAR(10) | utility_invoker_type - Tipo de invocador del programa de utilidad. Esta interfaz devuelve un identificador de texto basándose en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • AUTO • USER |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |
| PROGRESS_LIST_ATTR | VARCHAR(10) | progress_list_attr - Atributos de la lista de progresos actual |
| PROGRESS_LIST_CUR_SEQ_NUM | INTEGER | progress_list_current_seq_num - Número de secuencia de lista de progresos actual |

Vista administrativa SNAPUTIL_PROGRESS y función de tabla SNAP_GET_UTIL_PROGRESS - Recuperar información de instantánea de grupo de datos lógicos de progreso

La vista administrativa SNAPUTIL_PROGRESS y la función de tabla SNAP_GET_UTIL_PROGRESS devuelven información de instantáneas acerca del progreso del programa de utilidad, en concreto el grupo de datos lógicos de progreso.

SNAPUTIL_PROGRESS, vista administrativa

La vista administrativa SNAPUTIL_PROGRESS, que se utiliza junto con la vista administrativa SNAPUTIL, proporciona la misma información que el mandato CLP LIST UTILITIES SHOW DETAIL.

El esquema es SYSIBMADM.

Consulte la Tabla 164 en la página 665 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPUTIL_PROGRESS
- Privilegio CONTROL en la vista administrativa SNAPUTIL_PROGRESS
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE para la función de tabla SNAP_GET_UTIL_PROGRESS
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMMAINT
- SYSADM

Ejemplo

Recuperar información detallada acerca de las unidades de progreso totales y completadas por ID de programa de utilidad.

```
SELECT SELECT UTILITY_ID, PROGRESS_TOTAL_UNITS, PROGRESS_COMPLETED_UNITS,  
        DBPARTITIONNUM FROM SYSIBMADM.SNAPUTIL_PROGRESS
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| UTILITY_ID | PROGRESS_TOTAL_UNITS | PROGRESS_COMPLETED_UNITS | DBPARTITIONNUM |
|------------|----------------------|--------------------------|----------------|
| 7 | 10 | 5 | 0 |
| 9 | 10 | 5 | 1 |

1 registro(s) seleccionado(s).

SNAP_GET_UTIL_PROGRESS, función de tabla

La función de tabla SNAP_GET_UTIL_PROGRESS devuelve la misma información que la vista administrativa SNAPUTIL_PROGRESS, pero le permite recuperar la información para una base de datos específica de una partición de base de datos específica, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_UTIL_PROGRESS, que se utiliza junto con la función de tabla SNAP_GET_UTIL, proporciona la misma información que el mandato CLP LIST UTILITIES SHOW DETAIL.

Consulte la Tabla 164 en la página 665 para ver una lista completa de la información que se puede devolver.

Sintaxis

►► SNAP_GET_UTIL_PROGRESS (núm-partición-bd) ►►

El esquema es SYSPROC.

Parámetro de la función de tabla

núm-partición-bd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si no se utiliza esta opción de entrada, se devolverán datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *núm-partición-bd* se establece en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_UTIL_PROGRESS tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE para la función de tabla SNAP_GET_UTIL_PROGRESS
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMANT
- SYSADM

Ejemplo

Recuperar información detallada acerca del progreso de los programas de utilidad de la partición conectada actualmente.

```
SELECT UTILITY_ID, PROGRESS_TOTAL_UNITS, PROGRESS_COMPLETED_UNITS,  
       DBPARTITIONNUM FROM TABLE(SNAP_GET_UTIL_PROGRESS(-1)) as T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
UTILITY_ID PROGRESS_TOTAL_UNITS PROGRESS_COMPLETED_UNITS DBPARTITIONNUM  
-----  
          7                10                5                0
```

1 registro(s) seleccionado(s).

Información devuelta

Tabla 164. Información devuelta por la vista administrativa SNAPUTIL_PROGRESS y la función de tabla SNAP_GET_UTIL_PROGRESS

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| UTILITY_ID | INTEGER | utility_id - ID de programa de utilidad. Exclusivo de una partición de base de datos. |
| PROGRESS_SEQ_NUM | INTEGER | progress_seq_num - Número de secuencia de progreso. Si es serie, el número de la fase. Si es simultáneo, puede ser NULL. |
| UTILITY_STATE | VARCHAR(16) | utility_state - Estado del programa de utilidad. Esta interfaz devuelve un identificador de texto basándose en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • ERROR • EXECUTE • WAIT |
| PROGRESS_DESCRIPTION | VARCHAR(2048) | progress_description - Descripción del progreso |
| PROGRESS_START_TIME | TIMESTAMP | progress_start_time - Hora de inicio del progreso. Hora de inicio si la fase se ha iniciado; de lo contrario, NULL. |
| PROGRESS_WORK_METRIC | VARCHAR(16) | progress_work_metric - Métrica de trabajo en curso. Esta interfaz devuelve un identificador de texto basándose en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • NOT_SUPPORT • BYTES • EXTENTS • INDEXES • PAGES • ROWS • TABLES |
| PROGRESS_TOTAL_UNITS | BIGINT | progress_total_units - Total de unidades de trabajo en curso |
| PROGRESS_COMPLETED_UNITS | BIGINT | progress_completed_units - Unidades de trabajo en curso completadas |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Procedimiento SNAP_WRITE_FILE

El procedimiento SNAP_WRITE_FILE graba datos de instantánea del sistema en un archivo ubicado en el subdirectorio tmp del directorio de la instancia.

Sintaxis

```
▶▶—SNAP_WRITE_FILE—(—tipo-petición—,—nombre-bd—,—númparticiónbd—)————▶▶
```

El esquema es SYSPROC.

Parámetros del procedimiento

tipo-petición

Argumento de entrada de tipo VARCHAR (32) que especifica un tipo de petición de instantánea válido. Los tipos de petición posibles son identificadores de texto basados en las sentencias define de sqlmon.h y son uno de los siguientes:

- APPL_ALL
- BUFFERPOOLS_ALL
- DB2
- DBASE_ALL
- DBASE_LOCKS
- DBASE_TABLES
- DBASE_TABLESPACES
- DYNAMIC_SQL

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada en el momento en que se llama a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique NULL o una serie vacía para tomar la instantánea de la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica un valor nulo, se establece -1 de forma implícita.

Autorización

Para ejecutar el procedimiento, el usuario debe tener autorización SYSADM, SYSCTRL, SYSMANT o SYSMON. Los usuarios que no tengan autorización SYSADM, SYSCTRL, SYSMANT o SYSMON podrán leer la instantánea guardada pasando valores nulos como entrada a funciones de tabla de la instantánea.

Ejemplo

Tomar una instantánea de la información del gestor de bases de datos especificando un tipo de petición de 'DB2' (que corresponde a SQLMA_DB2) y adoptando los valores por omisión de la base de datos conectada actualmente y la partición de base de datos actual.

```
CALL SYSPROC.SNAP_WRITE_FILE ('DB2', '', -1)
```

Esto dará como resultado que se graben datos de instantánea en el directorio temporal de instancia, que es `sql1ib/tmp/SQLMA_DB2.dat` en los sistemas operativos UNIX y `sql1ib\DB2\tmp\SQLMA_DB2.dat` en un sistema operativo Windows.

Notas de uso

Si se proporciona un parámetro de entrada no reconocido, se devuelve el error siguiente: SQL2032N El parámetro "REQUEST_TYPE" no es válido.

Vista administrativa SNAPAGENT y función de tabla SNAP_GET_AGENT – Recuperar información de instantáneas de la aplicación del grupo de datos lógicos agent

La vista administrativa SNAPAGENT y la función de tabla SNAP_GET_AGENT devuelven información sobre agentes de una instantánea de la aplicación, en concreto el grupo de datos lógicos agent.

Vista administrativa SNAPAGENT

Esta vista administrativa permite recuperar la información de instantáneas de la aplicación del grupo de datos lógicos agent para la base de datos conectada actualmente.

La vista administrativa SNAPAGENT, utilizada con las vistas administrativas SNAPAGENT_MEMORY_POOL, SNAPAPPL, SNAPAPPL_INFO, SNAPSTMT y SNAPSUBSECTION, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR APPLICATIONS ON alias de base de datos, pero recupera datos de todas las particiones de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 135 en la página 518 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPAGENT
- Privilegio CONTROL en la vista administrativa SNAPAGENT
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE en la función de tabla SNAP_GET_AGENT
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMMAINT
- SYSADM

Ejemplo

Recuperar toda la información de instantáneas de la aplicación para la base de datos conectada actualmente desde el grupo de datos lógicos agent.

```
SELECT * FROM SYSIBMADM.SNAPAGENT
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| SNAPSHOT_TIMESTAMP | DB_NAME | AGENT_ID | ... |
|----------------------------|---------|----------|-----|
| 2005-07-19-11.03.26.740423 | SAMPLE | 101 | ... |
| 2005-07-19-11.03.26.740423 | SAMPLE | 49 | ... |

2 registro(s) seleccionado(s). ...

Datos de salida de esta consulta (continuación).

| AGENT_PID | LOCK_TIMEOUT_VAL | DBPARTITIONNUM |
|-----------|------------------|----------------|
| 11980 | -1 | 0 |
| 15940 | -1 | 0 |

SNAP_GET_AGENT, función de tabla

La función de tabla SNAP_GET_AGENT devuelve la misma información que la vista administrativa SNAPAGENT, pero permite recuperar la información para una base de datos concreta en una partición específica de la base de datos, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_AGENT, utilizada con las funciones de tabla SNAP_GET_AGENT_MEMORY_POOL, SNAP_GET_APPL_V95, SNAP_GET_APPL_INFO_V95, SNAP_GET_STMT y SNAP_GET_SUBSECTION, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR ALL APPLICATIONS, pero recupera datos de todas las particiones de base de datos.

Consulte la Tabla 135 en la página 518 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
▶▶ SNAP_GET_AGENT (—nombre-bd— [ , númparticiónbd ] ) ▶▶
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de

base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique una serie vacía para tomar la instantánea desde la base de datos conectada actualmente. Especifique un valor NULL para tomar la instantánea desde todas las bases de datos que se encuentran en la misma instancia que la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_AGENT toma una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE en la función de tabla SNAP_GET_AGENT
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMANT
- SYSADM

Ejemplo

Recuperar toda la información de instantáneas de la aplicación para todas las aplicaciones en todas las bases de datos activas.

```
SELECT * FROM TABLE(SNAP_GET_AGENT(CAST(NULL AS VARCHAR(128)), -1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| SNAPSHOT_TIMESTAMP | DB_NAME | AGENT_ID | ... |
|----------------------------|---------|----------|-----|
| 2006-01-03-17.21.38.530785 | SAMPLE | 48 | ... |
| 2006-01-03-17.21.38.530785 | SAMPLE | 47 | ... |
| 2006-01-03-17.21.38.530785 | SAMPLE | 46 | ... |
| 2006-01-03-17.21.38.530785 | TESTDB | 30 | ... |

```

2006-01-03-17.21.38.530785 TESTDB          29 ...
2006-01-03-17.21.38.530785 TESTDB          28 ...

```

6 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```

... AGENT_PID          LOCK_TIMEOUT_VAL      DBPARTITIONNUM
... -----
...          7696              -1              0
...          8536              -1              0
...          6672              -1              0
...          2332              -1              0
...          8360              -1              0
...          6736              -1              0
...

```

Información devuelta

Tabla 165. Información devuelta por la vista administrativa SNAPAGENT y la función de tabla SNAP_GET_AGENT

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| AGENT_ID | BIGINT | agent_id - Descriptor de contexto de aplicación (ID de agente) |
| AGENT_PID | BIGINT | agent_pid - Unidad susceptible de envío de motor (EDU) |
| LOCK_TIMEOUT_VAL | BIGINT | lock_timeout_val - Tiempo de espera excedido de bloqueo (segundos) |
| DBPARTITIONNUM | SMALLINT | La partición de base de datos desde la que se han recuperado los datos para la fila. |

Vista administrativa SNAPAGENT_MEMORY_POOL y función de tabla SNAP_GET_AGENT_MEMORY_POOL – Recuperar información de instantáneas del grupo de datos lógicos memory_pool

La vista administrativa SNAPAGENT_MEMORY_POOL y la tabla de función SNAP_GET_AGENT_MEMORY_POOL devuelven información acerca del uso de la memoria en el nivel de agente.

Vista administrativa SNAPAGENT_MEMORY_POOL

Esta vista administrativa permite recuperar la información de instantáneas del grupo de datos lógicos memory_pool sobre el uso de la memoria en el nivel del agente para la base de datos conectada actualmente.

La vista administrativa SNAPAGENT_MEMORY_POOL, utilizada con las vistas administrativas SNAPAGENT, SNAPAPPL, SNAPAPPL_INFO, SNAPSTMT y

SNAPSUBSECTION, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR APPLICATIONS ON alias de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 136 en la página 521 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPAGENT_MEMORY_POOL
- Privilegio CONTROL en la vista administrativa SNAPAGENT_MEMORY_POOL
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_AGENT_MEMORY_POOL
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMMAINT
- SYSADM

Ejemplo

Recuperar una lista de agrupaciones de memoria y su tamaño actual.

```
SELECT AGENT_ID, POOL_ID, POOL_CUR_SIZE FROM SYSIBMADM.SNAPAGENT_MEMORY_POOL
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| AGENT_ID | POOL_ID | POOL_CUR_SIZE |
|----------|--------------|---------------|
| 48 | APPLICATION | 65536 |
| 48 | OTHER | 65536 |
| 48 | APPL_CONTROL | 65536 |
| 47 | APPLICATION | 65536 |
| 47 | OTHER | 131072 |
| 47 | APPL_CONTROL | 65536 |
| 46 | OTHER | 327680 |
| 46 | APPLICATION | 262144 |
| 46 | APPL_CONTROL | 65536 |

9 registro(s) seleccionado(s).

SNAP_GET_AGENT_MEMORY_POOL, función de tabla

La función de tabla SNAP_GET_AGENT_MEMORY_POOL devuelve la misma información que la vista administrativa SNAPAGENT_MEMORY_POOL, pero permite recuperar la información para una base de datos concreta en una partición específica de la base de datos, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_AGENT_MEMORY_POOL, utilizada con las funciones de tabla SNAP_GET_AGENT, SNAP_GET_APPL_V95, SNAP_GET_APPL_INFO_V95, SNAP_GET_STMT y SNAP_GET_SUBSECTION, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR ALL APPLICATIONS.

Consulte la Tabla 136 en la página 521 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
▶▶ SNAP_GET_AGENT_MEMORY_POOL ( ( nombrebd [ , numparticiónbd ] ) ) ▶▶
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique una serie vacía para tomar la instantánea desde la base de datos conectada actualmente. Especifique un valor NULL para tomar la instantánea desde todas las bases de datos que se encuentran en la misma instancia que la base de datos conectada actualmente.

numparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *numparticiónbd* se establece en NULL, -1 se establece implícitamente para *numparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *numparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_AGENT_MEMORY_POOL toma una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_AGENT_MEMORY_POOL
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMMAINT
- SYSADM

Ejemplo

Recuperar una lista de agrupaciones de memoria y su tamaño actual para todas las bases de datos.

```
SELECT SUBSTR(DB_NAME,1,8) AS DB_NAME, AGENT_ID, POOL_ID, POOL_CUR_SIZE
FROM TABLE(SNAP_GET_AGENT_MEMORY_POOL(CAST (NULL AS VARCHAR(128)), -1))
AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| DB_NAME | AGENT_ID | POOL_ID | POOL_CUR_SIZE |
|---------|----------|--------------|---------------|
| SAMPLE | 48 | APPLICATION | 65536 |
| SAMPLE | 48 | OTHER | 65536 |
| SAMPLE | 48 | APPL_CONTROL | 65536 |
| SAMPLE | 47 | APPLICATION | 65536 |
| SAMPLE | 47 | OTHER | 131072 |
| SAMPLE | 47 | APPL_CONTROL | 65536 |
| SAMPLE | 46 | OTHER | 327680 |
| SAMPLE | 46 | APPLICATION | 262144 |
| SAMPLE | 46 | APPL_CONTROL | 65536 |
| TESTDB | 30 | APPLICATION | 65536 |
| TESTDB | 30 | OTHER | 65536 |
| TESTDB | 30 | APPL_CONTROL | 65536 |
| TESTDB | 29 | APPLICATION | 65536 |
| TESTDB | 29 | OTHER | 131072 |
| TESTDB | 29 | APPL_CONTROL | 65536 |
| TESTDB | 28 | OTHER | 327680 |
| TESTDB | 28 | APPLICATION | 65536 |
| TESTDB | 28 | APPL_CONTROL | 65536 |

18 registro(s) seleccionado(s).

Información devuelta

Tabla 166. Información devuelta por la vista administrativa SNAPAGENT_MEMORY_POOL y la función de tabla SNAP_GET_AGENT_MEMORY_POOL

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| AGENT_ID | BIGINT | agent_id - Descriptor de contexto de aplicación (ID de agente) |
| AGENT_PID | BIGINT | agent_pid - Unidad susceptible de envío de motor (EDU) |

Tabla 166. Información devuelta por la vista administrativa SNAPAGENT_MEMORY_POOL y la función de tabla SNAP_GET_AGENT_MEMORY_POOL (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|--|
| POOL_ID | VARCHAR(14) | pool_id - Identificador de agrupación de memoria. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • APP_GROUP • APPL_CONTROL • APPLICATION • BP • CAT_CACHE • DATABASE • DFM • FCMBP • IMPORT_POOL • LOCK_MGR • MONITOR • OTHER • PACKAGE_CACHE • QUERY • SHARED_SORT • SORT • STATEMENT • STATISTICS • UTILITY |
| POOL_CUR_SIZE | BIGINT | pool_cur_size - Tamaño actual de la agrupación de memoria |
| POOL_WATERMARK | BIGINT | pool_watermark - Marca límite de agrupación de memoria |
| POOL_CONFIG_SIZE | BIGINT | pool_config_size - Tamaño configurado de la agrupación de memoria |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa SNAPAPPL_INFO y función de tabla SNAP_GET_APPL_INFO_V95 – Recuperar información de instantáneas del grupo de datos lógicos appl_info

La vista administrativa SNAPAPPL_INFO y la función de tabla SNAP_GET_APPL_INFO_V95 devuelven información acerca de las aplicaciones a partir de una instantánea de la aplicación, en concreto del grupo de datos lógico appl_info.

SNAPAPPL_INFO, vista administrativa

Esta vista administrativa permite recuperar la información de instantáneas del grupo de datos lógicos appl_info para la base de datos conectada actualmente.

La vista administrativa SNAPAPPL_INFO, utilizada con las vistas administrativas SNAPAGENT, SNAPAGENT_MEMORY_POOL, SNAPAPPL, SNAPSTMT y SNAPSUBSECTION, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR APPLICATIONS ON alias de base de datos, pero recupera datos de todas las particiones de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 137 en la página 526 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPAPPL_INFO
- Privilegio CONTROL en la vista administrativa SNAPAPPL_INFO
- Autorización DATAACCESS

Además, se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE en la función de tabla SNAP_GET_APPL_INFO_V95
- Autorización DATAACCESS

Además, se necesita una de las autorizaciones siguientes:

- SYSMON
- SYSMAINT
- SYSCTRL
- SYSADM

Ejemplo

Recuperar el estado de las aplicaciones conectadas a la base de datos actual.

```
SELECT AGENT_ID, SUBSTR(APPL_NAME,1,10) AS APPL_NAME, APPL_STATUS
FROM SYSIBMADM.SNAPAPPL_INFO
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| AGENT_ID | APPL_NAME | APPL_STATUS |
|----------|-----------|-------------|
| 101 | db2bp.exe | UOWEXEC |
| 49 | db2bp.exe | CONNECTED |

2 registro(s) seleccionado(s).

SNAP_GET_APPL_INFO_V95, función de tabla

La función de tabla SNAP_GET_APPL_INFO_V95 devuelve la misma información que la vista administrativa SNAPAPPL_INFO, pero permite recuperar la información para una base de datos concreta en una partición específica de la base de datos, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_APPL_INFO_V95, utilizada con las funciones de tabla SNAP_GET_AGENT, SNAP_GET_AGENT_MEMORY_POOL, SNAP_GET_APPL_V95, SNAP_GET_STMT y SNAP_GET_SUBSECTION, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR ALL APPLICATIONS, pero recupera datos de todas las particiones de base de datos.

Consulte la Tabla 137 en la página 526 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
▶▶ SNAP_GET_APPL_INFO_V95 (—nombrebd— [ , númparticiónbd ] ) ▶▶
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique una serie vacía para tomar la instantánea desde la base de datos conectada actualmente. Especifique un valor NULL para tomar la instantánea desde todas las bases de datos que se encuentran en la misma instancia que la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_APPL_INFO_V95 tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE en la función de tabla SNAP_GET_APPL_INFO_V95
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYCTRL
- SYMAINT
- SYSADM

Ejemplos

Recuperar el estado de todas las aplicaciones en la partición de base de datos conectada.

```
SELECT SUBSTR(DB_NAME,1,8) AS DB_NAME, AGENT_ID,
       SUBSTR(APPL_NAME,1,10) AS APPL_NAME, APPL_STATUS
FROM TABLE(SNAP_GET_APPL_INFO_V95(CAST(NULL AS VARCHAR(128)),-1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
DB_NAME  AGENT_ID          APPL_NAME  APPL_STATUS
-----
TOOLSDB          14 db2bp.exe  CONNECTED
SAMPLE           15 db2bp.exe  UOWEXEC
SAMPLE           8  javaw.exe CONNECTED
SAMPLE           7  db2bp.exe UOWWAIT
```

4 registro(s) seleccionado(s).

A continuación se muestra lo que se obtiene al ejecutar SELECT sobre el resultado de la función de tabla.

```
SELECT SUBSTR(DB_NAME,1,8) AS DB_NAME, AUTHORITY_LVL
FROM TABLE(SNAP_GET_APPL_INFO_V95(CAST(NULL AS VARCHAR(128)),-1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
DB_NAME  AUTHORITY_LVL
-----
TESTDB   SYSADM(GROUP) + DBADM(USER) + CREATETAB(USER, GROUP) +
        BINDADD(USER, GROUP) + CONNECT(USER, GROUP) +
        CREATE_NOT_FENC(USER) + IMPLICIT_SCHEMA(USER, GROUP) +
        LOAD(USER) + CREATE_EXT_RT(USER) + QUIESCE_CONN(USER)
TESTDB   SYSADM(GROUP) + DBADM(USER) + CREATETAB(USER, GROUP) +
        BINDADD(USER, GROUP) + CONNECT(USER, GROUP) +
        CREATE_NOT_FENC(USER) + IMPLICIT_SCHEMA(USER, GROUP) +
        LOAD(USER) + CREATE_EXT_RT(USER) + QUIESCE_CONN(USER)
TESTDB   SYSADM(GROUP) + DBADM(USER) + CREATETAB(USER, GROUP) +
        BINDADD(USER, GROUP) + CONNECT(USER, GROUP) +
        CREATE_NOT_FENC(USER) + IMPLICIT_SCHEMA(USER, GROUP) +
        LOAD(USER) + CREATE_EXT_RT(USER) + QUIESCE_CONN(USER)
```

3 registro(s) seleccionado(s).

Información devuelta

Tabla 167. Información devuelta por la vista administrativa SNAPAPPL_INFO y la función de tabla SNAP_GET_APPL_INFO_V95

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| AGENT_ID | BIGINT | agent_id - Descriptor de contexto de aplicación (ID de agente) |

Tabla 167. Información devuelta por la vista administrativa SNAPAPPL_INFO y la función de tabla SNAP_GET_APPL_INFO_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|--|
| APPL_STATUS | VARCHAR(22) | appl_status - Estado de aplicación. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • BACKUP • COMMIT_ACT • COMP • CONNECTED • CONNECTPEND • CREATE_DB • DECOUPLED • DISCONNECTPEND • INTR • IOERROR_WAIT • LOAD • LOCKWAIT • QUIESCE_TABLESPACE • RECOMP • REMOTE_RQST • RESTART • RESTORE • ROLLBACK_ACT • ROLLBACK_TO_SAVEPOINT • TEND • THABRT • THCOMT • TPREP • UNLOAD • UOWEXEC • UOWWAIT • WAITFOR_REMOTE |
| CODEPAGE_ID | BIGINT | codepage_id - ID de página de códigos utilizada por aplicación |
| NUM_ASSOC_AGENTS | BIGINT | num_assoc_agents - Número de agentes asociados |
| COORD_NODE_NUM | SMALLINT | coord_node - Nodo de coordinación |

Tabla 167. Información devuelta por la vista administrativa SNAPAPPL_INFO y la función de tabla SNAP_GET_APPL_INFO_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|---|
| AUTHORITY_LVL | VARCHAR(512) | <p>authority_bitmap - Nivel de autorización del usuario.</p> <p>Esta interfaz devuelve un identificador de texto basado en las autorizaciones de base de datos definidas en sql.h y su fuente y tiene el siguiente formato: autorización(fuente, ...) + autorización(fuente , ...) + ... La fuente de una autorización puede ser múltiple: de USER, de GROUP, o de USER y GROUP.</p> <p>Los posibles valores para "autorización" son:</p> <ul style="list-style-type: none"> • ACCESSCTRL • BINDADD • CONNECT • CREATE_EXT_RT • CREATE_NOT_FENC • CREATETAB • DATAACCESS • DBADM • EXPLAIN • IMPLICIT_SCHEMA • LOAD • LIBADM • QUIESCE_CONN • SECADM • SQLADM • SYSADM • SYSCTRL • SYSMANT • SYSMON • SYSQUIESCE • WLMADM <p>Los posibles valores para "fuente" son:</p> <ul style="list-style-type: none"> • USER – autorización que se otorga al usuario o a un rol otorgado al usuario. • GROUP – autorización que se otorga a un grupo al que pertenece el usuario o a un rol que se ha otorgado al grupo al que pertenece el usuario. |

Tabla 167. Información devuelta por la vista administrativa SNAPAPPL_INFO y la función de tabla SNAP_GET_APPL_INFO_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| CLIENT_PID | BIGINT | client_pid - ID de proceso de cliente |
| COORD_AGENT_PID | BIGINT | coord_agent_pid - Agente coordinador |
| STATUS_CHANGE_TIME | TIMESTAMP | status_change_time - Hora de cambio de estado de aplicación |
| CLIENT_PLATFORM | VARCHAR(12) | <p>client_platform - Plataforma operativa de cliente. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h,</p> <ul style="list-style-type: none"> • AIX • AIX64 • AS400_DRDA • DOS • DYNIX • HP • HP64 • HPIA • HPIA64 • LINUX • LINUX390 • LINUXIA64 • LINUXPPC • LINUXPPC64 • LINUXX8664 • LINUXZ64 • MAC • MVS_DRDA • NT • NT64 • OS2 • OS390 • SCO • SGI • SNI • SUN • SUN64 • UNKNOWN • UNKNOWN_DRDA • VM_DRDA • VSE_DRDA • WINDOWS |

Tabla 167. Información devuelta por la vista administrativa SNAPAPPL_INFO y la función de tabla SNAP_GET_APPL_INFO_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------|---------------|---|
| CLIENT_PROTOCOL | VARCHAR(10) | client_protocol - Protocolo de comunicaciones de cliente. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h, <ul style="list-style-type: none"> • CPIC • LOCAL • NETBIOS • NPIPE • TCPIP (para DB2 UDB) • TCPIP4 • TCPIP6 |
| TERRITORY_CODE | SMALLINT | territory_code - Código de territorio de base de datos |
| APPL_NAME | VARCHAR(256) | appl_name - Nombre de aplicación |
| APPL_ID | VARCHAR(128) | appl_id - ID de aplicación |
| SEQUENCE_NO | VARCHAR(4) | sequence_no - Número de secuencia |
| PRIMARY_AUTH_ID | VARCHAR(128) | auth_id - ID de autorización |
| SESSION_AUTH_ID | VARCHAR(128) | session_auth_id - ID de autorización de sesión |
| CLIENT_NNAME | VARCHAR(128) | El elemento del supervisor client_nname ha quedado obsoleto. El valor devuelto no es un valor válido. |
| CLIENT_PRDID | VARCHAR(128) | client_prdid - ID de producto/versión de cliente |
| INPUT_DB_ALIAS | VARCHAR(128) | input_db_alias - Alias de base de datos de entrada |
| CLIENT_DB_ALIAS | VARCHAR(128) | client_db_alias - Alias de base de datos utilizado por aplicación |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| DB_PATH | VARCHAR(1024) | db_path - Vía de acceso de base de datos |
| EXECUTION_ID | VARCHAR(128) | execution_id - ID de inicio de sesión de usuario |
| CORR_TOKEN | VARCHAR(128) | corr_token - Símbolo de correlación DRDA |
| TPMON_CLIENT_USERID | VARCHAR(256) | tpmon_client_userid - ID de usuario de cliente de supervisor de TP |
| TPMON_CLIENT_WKSTN | VARCHAR(256) | tpmon_client_wkstn - Nombre de estación de trabajo de cliente de supervisor de TP |

Tabla 167. Información devuelta por la vista administrativa SNAPAPPL_INFO y la función de tabla SNAP_GET_APPL_INFO_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|---|
| TPMON_CLIENT_APP | VARCHAR(256) | tpmon_client_app - Nombre de aplicación de cliente de supervisor de TP |
| TPMON_ACC_STR | VARCHAR(200) | tpmon_acc_str - Serie de contabilidad de cliente de supervisor de TP |
| DBPARTITIONNUM | SMALLINT | La partición de base de datos desde la que se han recuperado los datos para la fila. |
| WORKLOAD_ID | INTEGER | ID de carga de trabajo actual. |
| IS_SYSTEM_APPL | SMALLINT | El valor de IS_SYSTEM_APPL indica si la aplicación es o no una aplicación del sistema interno DB2 0 significa que se una aplicación de usuario 1 significa que es una aplicación del sistema. Un ejemplo de una aplicación del sistema DB2 es un supervisor de sucesos de DB2. En general, los nombres de aplicaciones del sistema DB2 comienzan por "db2". Por ejemplo: db2stmm, db2taskd. |

Vista administrativa SNAPAPPL y función de tabla SNAP_GET_APPL_V95 – Recuperar información de instantáneas del grupo de datos lógicos appl

La "SNAPAPPL, vista administrativa" en la página 530 y la "SNAP_GET_APPL_V95, función de tabla" en la página 531 devuelven información sobre aplicaciones desde una instantánea de aplicaciones, en concreto, el grupo de datos lógicos appl.

SNAPAPPL, vista administrativa

Esta vista administrativa permite recuperar la información de instantáneas del grupo de datos lógicos appl para la base de datos conectada actualmente.

La vista administrativa SNAPAPPL, utilizada con las vistas administrativas SNAPAGENT, SNAPAGENT_MEMORY_POOL, SNAPAPPL_INFO, SNAPSTMT y SNAPSUBSECTION, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR APPLICATIONS ON alias de base de datos, pero recupera datos de todas las particiones de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 138 en la página 533 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPAPPL
- Privilegio CONTROL en la vista administrativa SNAPAPPL
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_APPL_V95
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMAINT
- SYSADM

Ejemplo

Recuperar detalles en filas leídas y grabadas para cada aplicación en la base de datos conectada.

```
SELECT SUBSTR(DB_NAME,1,8) AS DB_NAME, AGENT_ID, ROWS_READ, ROWS_WRITTEN
      FROM SYSIBMADM.SNAPAPPL
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| DB_NAME | AGENT_ID | ROWS_READ | ROWS_WRITTEN |
|---------|----------|-----------|--------------|
| SAMPLE | | 7 | 25 |

1 registro(s) seleccionado(s).

SNAP_GET_APPL_V95, función de tabla

La función de tabla SNAP_GET_APPL_V95 devuelve la misma información que la vista administrativa SNAPAPPL, pero permite recuperar la información para una base de datos concreta en una partición específica de la base de datos, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_APPL_V95, utilizada con las funciones de tabla SNAP_GET_AGENT, SNAP_GET_AGENT_MEMORY_POOL, SNAP_GET_APPL_INFO_V95, SNAP_GET_STMT y SNAP_GET_SUBSECTION, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR ALL APPLICATIONS, pero recupera datos de todas las particiones de base de datos.

Consulte la Tabla 138 en la página 533 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
▶▶ SNAP_GET_APPL_V95 (—nombrebd— [ , númparticiónbd ] )▶▶
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique una serie vacía para tomar la instantánea desde la base de datos conectada actualmente. Especifique un valor NULL para tomar la instantánea desde todas las bases de datos que se encuentran en la misma instancia que la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_APPL_V95 tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_APPL_V95
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar detalles en filas leídas y grabadas para cada aplicación en todas las bases de datos activas.

```
SELECT SUBSTR(DB_NAME,1,8) AS DB_NAME, AGENT_ID, ROWS_READ, ROWS_WRITTEN
FROM TABLE (SNAP_GET_APPL_V95(CAST(NULL AS VARCHAR(128)),-1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| DB_NAME | AGENT_ID | ROWS_READ | ROWS_WRITTEN |
|---------|----------|-----------|--------------|
| WSDB | 679 | 0 | 0 |
| WSDB | 461 | 3 | 0 |
| WSDB | 460 | 4 | 0 |
| TEST | 680 | 4 | 0 |
| TEST | 455 | 6 | 0 |
| TEST | 454 | 0 | 0 |
| TEST | 453 | 50 | 0 |

Información devuelta

Tabla 168. Información devuelta por la vista administrativa SNAPAPPL y la función de tabla SNAP_GET_APPL_V95

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| AGENT_ID | BIGINT | agent_id - Descriptor de contexto de aplicación (ID de agente) |
| UOW_LOG_SPACE_USED | BIGINT | uow_log_space_used - Unidad de espacio de registro de trabajo utilizado |
| ROWS_READ | BIGINT | rows_read - Filas leídas |
| ROWS_WRITTEN | BIGINT | rows_written - Filas grabadas |
| INACT_STMTHIST_SZ | BIGINT | stmt_history_list_size - Tamaño de lista de historial de sentencias |
| POOL_DATA_L_READS | BIGINT | pool_data_l_reads - Lecturas lógicas de datos de la agrupación de almacenamientos intermedios |
| POOL_DATA_P_READS | BIGINT | pool_data_p_reads - Lecturas físicas de datos de la agrupación de almacenamientos intermedios |
| POOL_DATA_WRITES | BIGINT | pool_data_writes - Grabaciones de datos de la agrupación de almacenamientos intermedios |
| POOL_INDEX_L_READS | BIGINT | pool_index_l_reads - Lecturas lógicas de índices de la agrupación de almacenamientos intermedios |
| POOL_INDEX_P_READS | BIGINT | pool_index_p_reads - Lecturas físicas de índices de la agrupación de almacenamientos intermedios |

Tabla 168. Información devuelta por la vista administrativa SNAPAPPL y la función de tabla SNAP_GET_APPL_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|--|
| POOL_INDEX_WRITES | BIGINT | pool_index_writes - Grabaciones de índices de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_L_READS | BIGINT | pool_temp_data_l_reads - Lecturas lógicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_P_READS | BIGINT | pool_temp_data_p_reads - Lecturas físicas de datos temporales de agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_L_READS | BIGINT | pool_temp_index_l_reads - Lecturas lógicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_P_READS | BIGINT | pool_temp_index_p_reads - Lecturas físicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_L_READS | BIGINT | pool_temp_xda_l_reads - Lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_P_READS | BIGINT | pool_temp_xda_p_reads - Elemento del supervisor de las lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |
| POOL_XDA_L_READS | BIGINT | pool_xda_l_reads - Lecturas lógicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_XDA_P_READS | BIGINT | pool_xda_p_reads - Lecturas físicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_XDA_WRITES | BIGINT | pool_xda_writes - Grabaciones de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_READ_TIME | BIGINT | pool_read_time - Tiempo total de lectura física de la agrupación de almacenamientos intermedios |
| POOL_WRITE_TIME | BIGINT | pool_write_time - Tiempo total de grabación física de la agrupación de almacenamientos intermedios |
| DIRECT_READS | BIGINT | direct_reads - Lecturas directas de la base de datos |
| DIRECT_WRITES | BIGINT | direct_writes - Grabaciones directas en la base de datos |

Tabla 168. Información devuelta por la vista administrativa SNAPAPPL y la función de tabla SNAP_GET_APPL_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------|---------------|--|
| DIRECT_READ_REQS | BIGINT | direct_read_reqs - Peticiones de lectura directa |
| DIRECT_WRITE_REQS | BIGINT | direct_write_reqs - Peticiones de grabación directa |
| DIRECT_READ_TIME | BIGINT | direct_read_time - Tiempo de lectura directa |
| DIRECT_WRITE_TIME | BIGINT | direct_write_time - Tiempo de grabación directa |
| UNREAD_PREFETCH_PAGES | BIGINT | unread_prefetch_pages - Páginas de captación previa no leídas |
| LOCKS_HELD | BIGINT | locks_held - Bloqueos retenidos |
| LOCK_WAITS | BIGINT | lock_waits - Esperas de bloqueo |
| LOCK_WAIT_TIME | BIGINT | lock_wait_time - Tiempo esperado en bloqueos |
| LOCK_ESCALS | BIGINT | lock_escals - Número de escalamientos de bloqueo |
| X_LOCK_ESCALS | BIGINT | x_lock_escals - Escalas de bloqueo exclusivas |
| DEADLOCKS | BIGINT | deadlocks - Puntos muertos detectados |
| TOTAL_SORTS | BIGINT | total_sorts - Clasificaciones totales |
| TOTAL_SORT_TIME | BIGINT | total_sort_time - Tiempo de clasificación total |
| SORT_OVERFLOWS | BIGINT | sort_overflows - Desbordamientos de clasificación |
| COMMIT_SQL_STMTS | BIGINT | commit_sql_stmts - Sentencias de confirmación intentadas |
| ROLLBACK_SQL_STMTS | BIGINT | rollback_sql_stmts - Sentencias de retrotracción intentadas |
| DYNAMIC_SQL_STMTS | BIGINT | dynamic_sql_stmts - Sentencias de SQL dinámico intentadas |
| STATIC_SQL_STMTS | BIGINT | static_sql_stmts - Sentencias de SQL estático intentadas |
| FAILED_SQL_STMTS | BIGINT | failed_sql_stmts - Operaciones de sentencia anómalas |
| SELECT_SQL_STMTS | BIGINT | select_sql_stmts - Sentencias Select de SQL ejecutadas |
| DDL_SQL_STMTS | BIGINT | ddl_sql_stmts - Sentencias de lenguaje de definición de datos (DDL) de SQL |
| UID_SQL_STMTS | BIGINT | uid_sql_stmts - Sentencias UPDATE/INSERT/DELETE de SQL ejecutadas |
| INT_AUTO_REBINDS | BIGINT | int_auto_rebinds - Revinculaciones automáticas internas |

Tabla 168. Información devuelta por la vista administrativa SNAPAPPL y la función de tabla SNAP_GET_APPL_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------|---------------|--|
| INT_ROWS_DELETED | BIGINT | int_rows_deleted - Filas internas suprimidas |
| INT_ROWS_UPDATED | BIGINT | int_rows_updated - Filas internas actualizadas |
| INT_COMMITS | BIGINT | int_commits - Confirmaciones internas |
| INT_ROLLBACKS | BIGINT | int_rollback - Retrotracciones internas |
| INT_DEADLOCK_ROLLBACKS | BIGINT | int_deadlock_rollback - Retrotracciones internas debido a punto muerto |
| ROWS_DELETED | BIGINT | rows_deleted - Filas suprimidas |
| ROWS_INSERTED | BIGINT | rows_inserted - Filas insertadas |
| ROWS_UPDATED | BIGINT | rows_updated - Filas actualizadas |
| ROWS_SELECTED | BIGINT | rows_selected - Filas seleccionadas |
| BINDS_PRECOMPILES | BIGINT | binds_precompiles - Vinculaciones/precompilaciones intentadas |
| OPEN_REM_CURS | BIGINT | open_rem_curs - Cursores remotos abiertos |
| OPEN_REM_CURS_BLK | BIGINT | open_rem_curs_blk - Cursores remotos abiertos con bloqueo |
| REJ_CURS_BLK | BIGINT | rej_curs_blk - Peticiones de cursor de bloqueo rechazadas |
| ACC_CURS_BLK | BIGINT | acc_curs_blk - Peticiones de cursor de bloqueo aceptadas |
| SQL_REQS_SINCE_COMMIT | BIGINT | sql_reqs_since_commit - Peticiones SQL desde la última confirmación |
| LOCK_TIMEOUTS | BIGINT | lock_timeouts - Número de tiempos de espera de bloqueo |
| INT_ROWS_INSERTED | BIGINT | int_rows_inserted - Filas internas insertadas |
| OPEN_LOC_CURS | BIGINT | open_loc_curs - Cursores locales abiertos |
| OPEN_LOC_CURS_BLK | BIGINT | open_loc_curs_blk - Cursores locales abiertos con bloqueo |
| PKG_CACHE_LOOKUPS | BIGINT | pkg_cache_lookups - Búsquedas de antememoria de paquete |
| PKG_CACHE_INSERTS | BIGINT | pkg_cache_inserts - Inserciones de antememoria de paquete |
| CAT_CACHE_LOOKUPS | BIGINT | cat_cache_lookups - Búsquedas de antememoria de catálogo |
| CAT_CACHE_INSERTS | BIGINT | cat_cache_inserts - Inserciones de antememoria de catálogo |

Tabla 168. Información devuelta por la vista administrativa SNAPAPPL y la función de tabla SNAP_GET_APPL_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------------|---------------|---|
| CAT_CACHE_OVERFLOWS | BIGINT | cat_cache_overflows - Desbordamientos de antememoria de catálogo |
| NUM_AGENTS | BIGINT | num_agents - Número de agentes trabajando en una sentencia |
| AGENTS_STOLEN | BIGINT | agents_stolen - Agentes robados |
| ASSOCIATED_AGENTS_TOP | BIGINT | associated_agents_top - Número máximo de agentes asociados |
| APPL_PRIORITY | BIGINT | appl_priority - Prioridad de agente de aplicación |
| APPL_PRIORITY_TYPE | VARCHAR(16) | appl_priority_type - Tipo de prioridad de la aplicación. Esta interfaz devuelve un identificador de texto, basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • DYNAMIC_PRIORITY • FIXED_PRIORITY |
| PREFETCH_WAIT_TIME | BIGINT | prefetch_wait_time - Tiempo esperado para captación previa |
| APPL_SECTION_LOOKUPS | BIGINT | appl_section_lookups - Búsquedas de sección |
| APPL_SECTION_INSERTS | BIGINT | appl_section_inserts - Inserciones de sección |
| LOCKS_WAITING | BIGINT | locks_waiting - Agentes actuales en espera de bloqueos |
| TOTAL_HASH_JOINS | BIGINT | total_hash_joins - Total de uniones hash |
| TOTAL_HASH_LOOPS | BIGINT | total_hash_loops - Total de bucles hash |
| HASH_JOIN_OVERFLOWS | BIGINT | hash_join_overflows - Desbordamientos de uniones hash |
| HASH_JOIN_SMALL_OVERFLOWS | BIGINT | hash_join_small_overflows - Desbordamientos pequeños de uniones hash |
| APPL_IDLE_TIME | BIGINT | appl_idle_time - Tiempo de inactividad de aplicación |
| UOW_LOCK_WAIT_TIME | BIGINT | uow_lock_wait_time - Tiempo total que la unidad de trabajo ha esperado en bloqueos |

Tabla 168. Información devuelta por la vista administrativa SNAPAPPL y la función de tabla SNAP_GET_APPL_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------|---------------|---|
| UOW_COMP_STATUS | VARCHAR(14) | uow_comp_status - Estado de finalización de la unidad de trabajo. Esta interfaz devuelve un identificador de texto, basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • APPL_END • UOWABEND • UOWCOMMIT • UOWDEADLOCK • UOWLOCKTIMEOUT • UOWROLLBACK • UOWUNKNOWN |
| AGENT_USR_CPU_TIME_S | BIGINT | agent_usr_cpu_time - Tiempo de CPU de usuario utilizado por el agente |
| AGENT_USR_CPU_TIME_MS | BIGINT | agent_usr_cpu_time - Tiempo de CPU de usuario utilizado por el agente |
| AGENT_SYS_CPU_TIME_S | BIGINT | agent_sys_cpu_time - Tiempo de CPU de sistema utilizado por el agente |
| AGENT_SYS_CPU_TIME_MS | BIGINT | agent_sys_cpu_time - Tiempo de CPU de sistema utilizado por el agente |
| APPL_CON_TIME | TIMESTAMP | appl_con_time - Indicación de fecha y hora de inicio de petición de conexión |
| CONN_COMPLETE_TIME | TIMESTAMP | conn_complete_time - Indicación de fecha y hora de finalización de petición de conexión |
| LAST_RESET | TIMESTAMP | last_reset - Indicación de fecha y hora de último restablecimiento |
| UOW_START_TIME | TIMESTAMP | uow_start_time - Indicación de fecha y hora de inicio de unidad de trabajo |
| UOW_STOP_TIME | TIMESTAMP | uow_stop_time - Indicación de fecha y hora de detención de unidad de trabajo |
| PREV_UOW_STOP_TIME | TIMESTAMP | prev_uow_stop_time - Indicación de fecha y hora de finalización de unidad de trabajo anterior |
| UOW_ELAPSED_TIME_S | BIGINT | uow_elapsed_time - Tiempo transcurrido de unidad de trabajo más reciente |
| UOW_ELAPSED_TIME_MS | BIGINT | uow_elapsed_time - Tiempo transcurrido de unidad de trabajo más reciente |

Tabla 168. Información devuelta por la vista administrativa SNAPAPPL y la función de tabla SNAP_GET_APPL_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------------------|---------------|--|
| ELAPSED_EXEC_TIME_S | BIGINT | elapsed_exec_time - Tiempo transcurrido de ejecución de sentencia |
| ELAPSED_EXEC_TIME_MS | BIGINT | elapsed_exec_time - Tiempo transcurrido de ejecución de sentencia |
| INBOUND_COMM_ADDRESS | VARCHAR(32) | inbound_comm_address - Dirección de comunicaciones de entrada |
| LOCK_TIMEOUT_VAL | BIGINT | lock_timeout_val - Tiempo de espera excedido de bloqueo (segundos) |
| PRIV_WORKSPACE_NUM_OVERFLOWS | BIGINT | priv_workspace_num_overflows - Desbordamientos de espacio de trabajo privado |
| PRIV_WORKSPACE_SECTION_INSERTS | BIGINT | priv_workspace_section_inserts - Inserciones de sección de espacio de trabajo privado |
| PRIV_WORKSPACE_SECTION_LOOKUPS | BIGINT | priv_workspace_section_lookups - Búsquedas de sección de espacio de trabajo privado |
| PRIV_WORKSPACE_SIZE_TOP | BIGINT | priv_workspace_size_top - Tamaño máximo de espacio de trabajo privado |
| SHR_WORKSPACE_NUM_OVERFLOWS | BIGINT | shr_workspace_num_overflows - Desbordamientos de espacio de trabajo compartido |
| SHR_WORKSPACE_SECTION_INSERTS | BIGINT | shr_workspace_section_inserts - Inserciones de sección de espacio de trabajo compartido |
| SHR_WORKSPACE_SECTION_LOOKUPS | BIGINT | shr_workspace_section_lookups - Búsquedas de sección de espacio de trabajo compartido |
| SHR_WORKSPACE_SIZE_TOP | BIGINT | shr_workspace_size_top - Tamaño máximo de espacio de trabajo compartido |
| DBPARTITIONNUM | SMALLINT | La partición de base de datos desde la que se han recuperado los datos para la fila. |
| CAT_CACHE_SIZE_TOP | BIGINT | cat_cache_size_top - Marca de límite superior de antememoria de catálogo |
| TOTAL_OLAP_FUNCS | BIGINT | Número total de funciones OLAP ejecutadas. |
| OLAP_FUNC_OVERFLOWS | BIGINT | Número de veces que los datos de función OLAP han superado el espacio de pila de clasificación disponible. |

Vista administrativa SNAPBP y función de tabla SNAP_GET_BP_V95 – Recuperar información de instantáneas del grupo lógico bufferpool

La vista administrativa SNAPBP y la función de tabla SNAP_GET_BP_V95 devuelven información sobre agrupaciones de almacenamientos intermedios de una instantánea de bufferpool, en particular el grupo de datos lógicos de bufferpool.

SNAPBP, vista administrativa

Esta vista administrativa permite recuperar la información de instantáneas del grupo de datos lógicos bufferpool para la base de datos conectada actualmente.

La vista administrativa SNAPBP, utilizada con la vista administrativa SNAPBP_PART, ofrece los datos equivalentes al mandato CLP GET SNAPSHOT FOR BUFFERPOOLS ON alias de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 139 en la página 542 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPBP
- Privilegio CONTROL en la vista administrativa SNAPBP
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_BP_V95
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCtrl
- SYSMAINT
- SYSADM

Ejemplo

Recuperar grabaciones en el índice y los datos de todas las agrupaciones de almacenamientos intermedios de la base de datos conectada actualmente.

```
SELECT SUBSTR(DB_NAME,1,8) AS DB_NAME,SUBSTR(BP_NAME,1,15)
      AS BP_NAME,POOL_DATA_WRITES,POOL_INDEX_WRITES
FROM SYSIBMADM.SNAPBP
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| DB_NAME | BP_NAME | POOL_DATA_WRITES | POOL_INDEX_WRITES |
|---------|----------------|------------------|-------------------|
| TEST | IBMDEFAULTBP | 0 | 0 |
| TEST | IBMSYSTEMBP4K | 0 | 0 |
| TEST | IBMSYSTEMBP8K | 0 | 0 |
| TEST | IBMSYSTEMBP16K | 0 | 0 |
| TEST | IBMSYSTEMBP32K | 0 | 0 |

5 registro(s) seleccionado(s)

SNAP_GET_BP_V95, función de tabla

La función de tabla SNAP_GET_BP_V95 devuelve la misma información que la vista administrativa SNAPBP, pero permite recuperar la información de una base de datos concreta en una partición específica de la base de datos, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_BP_V95, utilizada con la función de tabla SNAP_GET_BP_PART, ofrece los datos equivalentes al mandato CLP GET SNAPSHOT FOR ALL BUFFERPOOLS.

Consulte la Tabla 139 en la página 542 para ver una lista completa de la información que se puede devolver.

Sintaxis

```

▶▶ SNAP_GET_BP_V95 ( ( nombrebd [ , numparticiónbd ] ) )

```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique una serie vacía para tomar la instantánea desde la base de datos conectada actualmente. Especifique un valor NULL para tomar la instantánea desde todas las bases de datos que se encuentran en la misma instancia que la base de datos conectada actualmente.

numparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *numparticiónbd* se establece en NULL, -1 se establece implícitamente para *numparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *numparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es

posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_BP_V95 tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_BP_V95
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar lecturas lógicas y físicas para todas las agrupaciones de almacenamientos intermedios de todas las bases de datos activas en la partición de base de datos conectada actualmente.

```
SELECT SUBSTR(T.DB_NAME,1,10) AS DB_NAME,
       SUBSTR(T.BP_NAME,1,20) AS BP_NAME,
       (T.POOL_DATA_L_READS+T.POOL_INDEX_L_READS) AS TOTAL_LOGICAL_READS,
       (T.POOL_DATA_P_READS+T.POOL_INDEX_P_READS) AS TOTAL_PHYSICAL_READS,
       T.DBPARTITIONNUM
FROM TABLE(SNAP_GET_BP_V95(CAST(NULL AS VARCHAR(128)), -1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
DB_NAME      BP_NAME      TOTAL_LOGICAL_READS  ...
-----
SAMPLE      IBMDEFAULTBP      0 ...
TOOLSDB     IBMDEFAULTBP      0 ...
TOOLSDB     BP32K0000         0 ...
```

3 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```
... TOTAL_PHYSICAL_READS DBPARTITIONNUM
... -----
...                0                0
...                0                0
...                0                0
```

Información devuelta

Tabla 169. Información devuelta por la vista administrativa SNAPBP y la función de tabla SNAP_GET_BP_V95

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |

Tabla 169. Información devuelta por la vista administrativa SNAPBP y la función de tabla SNAP_GET_BP_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------|---------------|---|
| BP_NAME | VARCHAR(128) | bp_name - Nombre de la agrupación de almacenamientos intermedios. |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| DB_PATH | VARCHAR(1024) | db_path - Vía de acceso de base de datos |
| INPUT_DB_ALIAS | VARCHAR(128) | input_db_alias - Alias de base de datos de entrada |
| POOL_DATA_L_READS | BIGINT | pool_data_l_reads - Lecturas lógicas de datos de la agrupación de almacenamientos intermedios |
| POOL_DATA_P_READS | BIGINT | pool_data_p_reads - Lecturas físicas de datos de la agrupación de almacenamientos intermedios |
| POOL_DATA_WRITES | BIGINT | pool_data_writes - Grabaciones de datos de la agrupación de almacenamientos intermedios |
| POOL_INDEX_L_READS | BIGINT | pool_index_l_reads - Lecturas lógicas de índices de la agrupación de almacenamientos intermedios |
| POOL_INDEX_P_READS | BIGINT | pool_index_p_reads - Lecturas físicas de índices de la agrupación de almacenamientos intermedios |
| POOL_INDEX_WRITES | BIGINT | pool_index_writes - Grabaciones de índices de la agrupación de almacenamientos intermedios |
| POOL_XDA_L_READS | BIGINT | pool_xda_l_reads - Lecturas lógicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_XDA_P_READS | BIGINT | pool_xda_p_reads - Lecturas físicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_XDA_WRITES | BIGINT | pool_xda_writes - Grabaciones de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_READ_TIME | BIGINT | pool_read_time - Tiempo total de lectura física de la agrupación de almacenamientos intermedios |
| POOL_WRITE_TIME | BIGINT | pool_write_time - Tiempo total de grabación física de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_READS | BIGINT | pool_async_data_reads - Lecturas de datos asíncronas de agrupación de almacenamientos intermedios |

Tabla 169. Información devuelta por la vista administrativa SNAPBP y la función de tabla SNAP_GET_BP_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|----------------------------|---------------|---|
| POOL_ASYNC_DATA_WRITES | BIGINT | pool_async_data_writes - Grabaciones de datos asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_READS | BIGINT | pool_async_index_reads - Lecturas de índice asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_WRITES | BIGINT | pool_async_index_writes - Grabaciones de índice asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_XDA_READS | BIGINT | pool_async_xda_reads - Lecturas de datos XDA asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_XDA_WRITES | BIGINT | pool_async_xda_writes - Grabaciones de datos XDA asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_READ_TIME | BIGINT | pool_async_read_time - Tiempo de lectura asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_WRITE_TIME | BIGINT | pool_async_write_time - Tiempo de grabación asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_READ_REQS | BIGINT | pool_async_data_read_reqs - Peticiones de lectura asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_READ_REQS | BIGINT | pool_async_index_read_reqs - Peticiones de lectura de índice asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_XDA_READ_REQS | BIGINT | pool_async_xda_read_reqs - Peticiones de lectura XDA asíncrona de agrupación de almacenamientos intermedios |
| DIRECT_READS | BIGINT | direct_reads - Lecturas directas de la base de datos |
| DIRECT_WRITES | BIGINT | direct_writes - Grabaciones directas en la base de datos |
| DIRECT_READ_REQS | BIGINT | direct_read_reqs - Peticiones de lectura directa |
| DIRECT_WRITE_REQS | BIGINT | direct_write_reqs - Peticiones de grabación directa |
| DIRECT_READ_TIME | BIGINT | direct_read_time - Tiempo de lectura directa |

Tabla 169. Información devuelta por la vista administrativa SNAPBP y la función de tabla SNAP_GET_BP_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|--|
| DIRECT_WRITE_TIME | BIGINT | direct_write_time - Tiempo de grabación directa |
| UNREAD_PREFETCH_PAGES | BIGINT | unread_prefetch_pages - Páginas de captación previa no leídas |
| FILES_CLOSED | BIGINT | files_closed - Archivos de base de datos cerrados |
| POOL_TEMP_DATA_L_READS | BIGINT | pool_temp_data_l_reads - Lecturas lógicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_P_READS | BIGINT | pool_temp_data_p_reads - Lecturas físicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_L_READS | BIGINT | pool_temp_index_l_reads - Lecturas lógicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_P_READS | BIGINT | pool_temp_index_p_reads - Lecturas físicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_L_READS | BIGINT | pool_temp_xda_l_reads - Lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_P_READS | BIGINT | pool_temp_xda_p_reads - Elemento del supervisor de las lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |
| POOL_NO_VICTIM_BUFFER | BIGINT | pool_no_victim_buffer - Almacenamiento intermedio sin víctimas de la agrupación de almacenamientos intermedios |
| PAGES_FROM_BLOCK_IOS | BIGINT | pages_from_block_ios - Número total de páginas leídas por E/S de bloque |
| PAGES_FROM_VECTORED_IOS | BIGINT | pages_from_vectored_ios - Número total de páginas leídas por E/S de vector |
| VECTORED_IOS | BIGINT | vectored_ios - Número de peticiones de E/S de vector |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa SNAPBP_PART y función de tabla SNAP_GET_BP_PART – Recuperar información de instantáneas del grupo de datos lógicos bufferpool_nodeinfo

La vista administrativa SNAPBP_PART y la función de tabla SNAP_GET_BP_PART devuelven información acerca de las agrupaciones de almacenamientos intermedios a partir de una instantánea determinada, el grupo de datos lógico bufferpool_nodeinfo.

SNAPBP_PART, vista administrativa

Esta vista administrativa permite recuperar la información de instantáneas del grupo de datos lógicos bufferpool_nodeinfo para la base de datos conectada actualmente.

La vista administrativa SNAPBP_PART, utilizada con la vista administrativa SNAPBP, ofrece los datos equivalentes al mandato CLP GET SNAPSHOT FOR BUFFERPOOLS ON alias de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 140 en la página 549 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPBP_PART
- Privilegio CONTROL en la vista administrativa SNAPBP_PART
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_BP_PART
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar datos de todas las agrupaciones de almacenamientos intermedios cuando esté conectado con la base de datos SAMPLE.

```
SELECT SUBSTR(DB_NAME,1,8) AS DB_NAME, SUBSTR(BP_NAME,1,15) AS BP_NAME,  
       BP_CUR_BUFFSZ, BP_NEW_BUFFSZ, BP_PAGES_LEFT_TO_REMOVE, BP_TBSP_USE_COUNT  
FROM SYSIBMADM.SNAPBP_PART
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| DB_NAME | BP_NAME | BP_CUR_BUFFSZ | BP_NEW_BUFFSZ | ... |
|---------|----------------|---------------|---------------|-----|
| SAMPLE | IBMDEFAULTBP | 1000 | 1000 | ... |
| SAMPLE | IBMSYSTEMBP4K | 16 | 16 | ... |
| SAMPLE | IBMSYSTEMBP8K | 16 | 16 | ... |
| SAMPLE | IBMSYSTEMBP16K | 16 | 16 | ... |

4 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

| ... | BP_PAGES_LEFT_TO_REMOVE | BP_TBSP_USE_COUNT |
|-----|-------------------------|-------------------|
| ... | 0 | 3 |
| ... | 0 | 0 |
| ... | 0 | 0 |
| ... | 0 | 0 |
| ... | | |

SNAP_GET_BP_PART, función de tabla

La función de tabla SNAP_GET_BP_PART devuelve la misma información que la vista administrativa SNAPBP_PART, pero permite recuperar la información para una base de datos concreta en una partición específica de la base de datos, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

Junto con la función de tabla SNAP_GET_BP_V95, la función de tabla SNAP_GET_BP_PART proporciona los datos equivalentes al mandato CLP GET SNAPSHOT FOR ALL BUFFERPOOLS.

Consulte la Tabla 140 en la página 549 para ver una lista completa de la información que se puede devolver.

Sintaxis

```

▶▶ SNAP_GET_BP_PART ( ( nombrebd [ , numparticiónbd ] ) )

```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique una serie vacía para tomar la instantánea desde la base de datos conectada actualmente. Especifique un valor NULL para tomar la instantánea de todas las agrupaciones de almacenamientos intermedios de todas las bases de datos de la misma instancia que la base de datos conectada actualmente.

numparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de

datos activas. Si *nombredb* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombredb*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombredb* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_BP_PART toma una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_BP_PART
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar datos de todas las agrupaciones de almacenamientos intermedios de todas las bases de datos activas cuando estén conectadas con la base de datos SAMPLE.

```
SELECT SUBSTR(DB_NAME,1,8) AS DB_NAME, SUBSTR(BP_NAME,1,15) AS BP_NAME,
       BP_CUR_BUFFSZ, BP_NEW_BUFFSZ, BP_PAGES_LEFT_TO_REMOVE, BP_TBSP_USE_COUNT
FROM TABLE(SNAP_GET_BP_PART(CAST(NULL AS VARCHAR(128)),-1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| DB_NAME | BP_NAME | BP_CUR_BUFFSZ | BP_NEW_BUFFSZ | ... |
|---------|----------------|---------------|---------------|-----|
| SAMPLE | IBMDEFAULTBP | 250 | 250 | ... |
| SAMPLE | IBMSYSTEMBP4K | 16 | 16 | ... |
| SAMPLE | IBMSYSTEMBP8K | 16 | 16 | ... |
| SAMPLE | IBMSYSTEMBP16K | 16 | 16 | ... |
| SAMPLE | IBMSYSTEMBP32K | 16 | 16 | ... |
| TESTDB | IBMDEFAULTBP | 250 | 250 | ... |
| TESTDB | IBMSYSTEMBP4K | 16 | 16 | ... |
| TESTDB | IBMSYSTEMBP8K | 16 | 16 | ... |
| TESTDB | IBMSYSTEMBP16K | 16 | 16 | ... |
| TESTDB | IBMSYSTEMBP32K | 16 | 16 | ... |

...

Datos de salida de esta consulta (continuación).

| ... | BP_PAGES_LEFT_TO_REMOVE | BP_TBSP_USE_COUNT |
|-----|-------------------------|-------------------|
| ... | 0 | 3 |


```

...           0           0
...           0           0
...           0           0
...           0           0
...           0           3
...           0           0
...           0           0
...           0           0
...           0           0
...

```

Información devuelta

Tabla 170. Información devuelta por la vista administrativa SNAPBP_PART y la función de tabla SNAP_GET_BP_PART

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| BP_NAME | VARCHAR(128) | bp_name - Nombre de la agrupación de almacenamientos intermedios. |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| BP_CUR_BUFFSZ | BIGINT | bp_cur_buffsz - Tamaño actual de agrupación de almacenamientos intermedios |
| BP_NEW_BUFFSZ | BIGINT | bp_new_buffsz - Tamaño de agrupación de almacenamientos intermedios nuevo |
| BP_PAGES_LEFT_TO_REMOVE | BIGINT | bp_pages_left_to_remove - Número de páginas restantes para eliminar |
| BP_TBSP_USE_COUNT | BIGINT | bp_tbsp_use_count - Número de espacios de tablas correlacionados con la agrupación de almacenamientos intermedios |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa SNAPCONTAINER y función de tabla SNAP_GET_CONTAINER_V91 – Recuperar información acerca de la instantánea de grupo de datos lógicos tablespace_container

La vista administrativa SNAPCONTAINER y la función de tabla SNAP_GET_CONTAINER_V91 devuelve información de la instantánea del espacio de tablas del grupo de datos lógicos tablespace_container.

SNAPCONTAINER, vista administrativa

Esta vista administrativa le permite recuperar información de instantánea de grupo de datos lógicos tablespace_container para la base de datos conectada actualmente.

La vista administrativa SNAPCONTAINER, utilizada con las vistas administrativas SNAPTBSP, SNAPTBSP_PART, SNAPTBSP QUIESCER y SNAPTBSP_RANGE, devuelve datos que son equivalentes a los datos que devuelve el mandato CLP GET SNAPSHOT FOR TABLESPACES ON alias de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 141 en la página 553 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPCONTAINER
- Privilegio CONTROL en la vista administrativa SNAPCONTAINER
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_CONTAINER_V91
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar información detallada acerca de los contenedores de espacio de tablas para todas las particiones de base de datos de la base de datos conectada actualmente.

```
SELECT SNAPSHOT_TIMESTAMP, SUBSTR(TBSP_NAME, 1, 15) AS TBSP_NAME,
       TBSP_ID, SUBSTR(CONTAINER_NAME, 1, 20) AS CONTAINER_NAME,
       CONTAINER_ID, CONTAINER_TYPE, ACCESSIBLE, DBPARTITIONNUM
FROM SYSIBMADM.SNAPCONTAINER ORDER BY DBPARTITIONNUM
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
SNAPSHOT_TIMESTAMP      TBSP_NAME      TBSP_ID      ...
-----
2006-01-08-16.49.24.639945 SYSCATSPACE      0 ...
2006-01-08-16.49.24.639945 TEMPSPACE1      1 ...
2006-01-08-16.49.24.639945 USERSPACE1      2 ...
2006-01-08-16.49.24.639945 SYSTOOLSPACE    3 ...
2006-01-08-16.49.24.640747 TEMPSPACE1      1 ...
2006-01-08-16.49.24.640747 USERSPACE1      2 ...
2006-01-08-16.49.24.639981 TEMPSPACE1      1 ...
2006-01-08-16.49.24.639981 USERSPACE1      2 ...
...
8 registro(s) seleccionado(s).
```

Datos de salida de esta consulta (continuación).

```
... CONTAINER_NAME      CONTAINER_ID      CONTAINER_TYPE      ...
-----
... /home/swalkty/swalkt      0 FILE_EXTENT_TAG      ...
```

```

... /home/swalkty/swalkt      0 PATH ...
... /home/swalkty/swalkt      0 FILE_EXTENT_TAG ...
... /home/swalkty/swalkt      0 FILE_EXTENT_TAG ...
... /home/swalkty/swalkt      0 PATH ...
... /home/swalkty/swalkt      0 FILE_EXTENT_TAG ...
... /home/swalkty/swalkt      0 PATH ...
... /home/swalkty/swalkt      0 FILE_EXTENT_TAG ...

```

Datos de salida de esta consulta (continuación).

```

... ACCESSIBLE DBPARTITIONNUM
... -----
...          1          0
...          1          0
...          1          0
...          1          0
...          1          1
...          1          1
...          1          2
...          1          2

```

SNAP_GET_CONTAINER_V91, función de tabla

La función de tabla SNAP_GET_CONTAINER_V91 devuelve la misma información que la vista administrativa SNAPCONTAINER, pero le permite recuperar la información para una base de datos específica de una partición de base de datos específica, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_CONTAINER_V91, utilizada con las funciones de tabla SNAP_GET_TBSP_V91, SNAP_GET_TBSP_PART_V91, SNAP_GET_TBSP QUIESCER y SNAP_GET_TBSP_RANGE, devuelve datos que son equivalentes a los que devuelve el mandato CLP GET SNAPSHOT FOR TABLESPACES ON alias de base de datos.

Consulte la Tabla 141 en la página 553 para ver una lista completa de la información que se puede devolver.

Sintaxis

```

▶▶ SNAP_GET_CONTAINER_V91 (—nombrebd—, númparticiónbd)

```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique NULL o una serie vacía para tomar la instantánea de la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de

datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_CONTAINER_V91 tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_CONTAINER_V91
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar información detallada acerca de los contenedores de espacio de tablas de la base de datos conectada actualmente de la partición de base de datos conectada actualmente.

```
SELECT SNAPSHOT_TIMESTAMP, TBSP_NAME, TBSP_ID, CONTAINER_NAME,
       CONTAINER_ID, CONTAINER_TYPE, ACCESSIBLE
FROM TABLE(SNAP_GET_CONTAINER_V91(' ', -1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| SNAPSHOT_TIMESTAMP | TBSP_NAME | TBSP_ID | ... |
|----------------------------|-------------------|---------|-----|
| 2005-04-25-14.42.10.899253 | SYSCATSPACE | 0 | ... |
| 2005-04-25-14.42.10.899253 | TEMPSPACE1 | 1 | ... |
| 2005-04-25-14.42.10.899253 | USERSPACE1 | 2 | ... |
| 2005-04-25-14.42.10.899253 | SYSTOOLSPACE | 3 | ... |
| 2005-04-25-14.42.10.899253 | MYTEMP | 4 | ... |
| 2005-04-25-14.42.10.899253 | WHATSNEWTEMPSPACE | 5 | ... |

Datos de salida de esta consulta (continuación).

| ... | CONTAINER_NAME | CONTAINER_ID | ... |
|-----|---------------------------------------|--------------|-----|
| ... | D:\DB2\NODE0000\SQL00002\SQLT0000.0 | 0 | ... |
| ... | D:\DB2\NODE0000\SQL00002\SQLT0001.0 | 0 | ... |
| ... | D:\DB2\NODE0000\SQL00002\SQLT0002.0 | 0 | ... |
| ... | D:\DB2\NODE0000\SQL00002\SYSTOOLSPACE | 0 | ... |
| ... | D:\DB2\NODE0000\SQL003 | 0 | ... |
| ... | d:\DGTTsWhatsNewContainer | 0 | ... |

Datos de salida de esta consulta (continuación).

```

... CONTAINER_TYPE ACCESSIBLE
... -----
... CONT_PATH          1
... CONT_PATH          1
... CONT_PATH          1
... CONT_PATH          1
... CONT_PATH          1
... CONT_PATH          1

```

Información devuelta

NOTA: Para que se devuelva la información del sistema de archivos, debe activarse el conmutador de supervisor del gestor de bases de datos BUFFERPOOL.

Tabla 171. Información devuelta por la vista administrativa SNAPCONTAINER y la función de tabla SNAP_GET_CONTAINER_V91

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| TBSP_NAME | VARCHAR(128) | tablespace_name - Nombre de espacio de tablas |
| TBSP_ID | BIGINT | tablespace_id - Identificación de espacio de tablas |
| CONTAINER_NAME | VARCHAR(256) | container_name - Nombre de contenedor |
| CONTAINER_ID | BIGINT | container_id - Identificación de contenedor |
| CONTAINER_TYPE | VARCHAR(16) | container_type - Tipo de contenedor. Es un identificador de texto basado en las sentencias define de sqlutil.h y es uno de los siguientes: <ul style="list-style-type: none"> • DISK_EXTENT_TAG • DISK_PAGE_TAG • FILE_EXTENT_TAG • FILE_PAGE_TAG • PATH |
| TOTAL_PAGES | BIGINT | container_total_pages - Total de páginas en contenedor |
| USABLE_PAGES | BIGINT | container_usable_pages - Páginas utilizables en contenedor |
| ACCESSIBLE | SMALLINT | container_accessible - Accesibilidad de contenedor |
| STRIPE_SET | BIGINT | container_stripe_set - Conjunto de bandas |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |
| FS_ID | VARCHAR(22) | fs_id - Número de identificación del sistema de archivos exclusivo |

Tabla 171. Información devuelta por la vista administrativa SNAPCONTAINER y la función de tabla SNAP_GET_CONTAINER_V91 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|---|
| FS_TOTAL_SIZE | BIGINT | fs_total_size - Tamaño total de un sistema de archivos |
| FS_USED_SIZE | BIGINT | fs_used_size - Cantidad de espacio utilizada por un sistema de archivos |

Vista administrativa SNAPDB y función de tabla SNAP_GET_DB_V95 – Recuperar información de instantánea desde el grupo lógico dbase

La “SNAPDB, vista administrativa” en la página 554 y la “SNAP_GET_DB_V95, función de tabla” en la página 555 devuelven información de instantánea desde el grupo lógico de base de datos (dbase).

SNAPDB, vista administrativa

Esta vista administrativa le permite recuperar información de instantánea desde el grupo lógico dbase para la base de datos conectada actualmente.

La vista administrativa SNAPDB, que se utiliza junto con las vistas administrativas SNAPDB_MEMORY_POOL, SNAPDETAILOG, SNAPHADR y SNAPSTORAGE_PATHS, proporciona información que es equivalente a la información que proporciona el mandato CLP GET SNAPSHOT FOR DATABASE ON alias de base de datos.

El esquema es SYSIBMADM.

En la Tabla 142 en la página 558 encontrará una lista completa de la información que se devuelve.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPDB
- Privilegio CONTROL en la vista administrativa SNAPDB
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_DB_V95
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplos

Recuperar el estado, la plataforma, la ubicación y el tiempo de conexión para todas las particiones de base de datos de la base de datos conectada actualmente.

```
SELECT SUBSTR(DB_NAME, 1, 20) AS DB_NAME, DB_STATUS, SERVER_PLATFORM,
       DB_LOCATION, DB_CONN_TIME, DBPARTITIONNUM
FROM SYSIBMADM.SNAPDB ORDER BY DBPARTITIONNUM
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| DB_NAME | DB_STATUS | SERVER_PLATFORM | DB_LOCATION | ... |
|---------|-----------|-----------------|-------------|-----|
| TEST | ACTIVE | AIX64 | LOCAL | ... |
| TEST | ACTIVE | AIX64 | LOCAL | ... |
| TEST | ACTIVE | AIX64 | LOCAL | ... |

3 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

| DB_CONN_TIME | DBPARTITIONNUM |
|----------------------------|----------------|
| 2006-01-08-16.48.30.665477 | 0 |
| 2006-01-08-16.48.34.005328 | 1 |
| 2006-01-08-16.48.34.007937 | 2 |

Esta rutina se puede utilizar realizando la siguiente llamada en la línea de mandatos:

```
SELECT TOTAL_OLAP_FUNCS, OLAP_FUNC_OVERFLOWS, ACTIVE_OLAP_FUNCS
FROM SYSIBMADM.SNAPDB
```

| TOTAL_OLAP_FUNCS | OLAP_FUNC_OVERFLOWS | ACTIVE_OLAP_FUNCS |
|------------------|---------------------|-------------------|
| 7 | 2 | 1 |

1 registro(s) seleccionado(s).

Después de ejecutar una carga de trabajo, el usuario puede utilizar la siguiente consulta:

```
SELECT STATS_CACHE_SIZE, STATS_FABRICATIONS, SYNC_RUNSTATS,
       ASYNC_RUNSTATS, STATS_FABRICATE_TIME, SYNC_RUNSTATS_TIME
FROM SYSIBMADM.SNAPDB
```

| STATS_CACHE_SIZE | STATS_FABRICATIONS | SYNC_RUNSTATS | ASYNC_RUNSTATS | ... |
|------------------|--------------------|---------------|----------------|-----|
| 128 | 2 | 1 | 0 | ... |

| STATS_FABRICATE_TIME | SYNC_RUNSTATS_TIME |
|----------------------|--------------------|
| 10 | 100 |

1 registro(s) seleccionado(s).

SNAP_GET_DB_V95, función de tabla

La función de tabla SNAP_GET_DB_V95 devuelve la misma información que la vista administrativa SNAPDB.

La función de tabla SNAP_GET_DB_V91, utilizada con las funciones de tabla SNAP_GET_DB_MEMORY_POOL, SNAP_GET_DETAILLOG_V95,

SNAP_GET_HADR y SNAP_GET_STORAGE_PATHS, proporciona información que es equivalente a la que proporciona el mandato CLP GET SNAPSHOT FOR ALL DATABASES.

En la Tabla 142 en la página 558 encontrará una lista completa de la información que se devuelve.

Sintaxis

```
▶▶ SNAP_GET_DB_V95 ( ( nombrebd [ , númparticiónbd ] ) ) ▶▶
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique una serie vacía para tomar la instantánea desde la base de datos conectada actualmente. Especifique un valor NULL para tomar la instantánea desde todas las bases de datos que se encuentran en la misma instancia que la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_DB_V95 tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_DB_V95
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL

- SYSMANT
- SYSADM

Ejemplos

Ejemplo 1: Recuperar el estado, la plataforma, la ubicación y el tiempo de conexión como vista de agregación de todas las particiones de base de datos de la base de datos conectada actualmente.

```
SELECT SUBSTR(DB_NAME, 1, 20) AS DB_NAME, DB_STATUS, SERVER_PLATFORM,
       DB_LOCATION, DB_CONN_TIME FROM TABLE(SNAP_GET_DB_V95(' ', -2)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
DB_NAME      DB_STATUS    SERVER_PLATFORM ...
-----
SAMPLE      ACTIVE      AIX64          ...
```

1 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```
... DB_LOCATION DB_CONN_TIME
... -----
... LOCAL      2005-07-24-22.09.22.013196
```

Ejemplo 2: Recuperar el estado, la plataforma, la ubicación y el tiempo de conexión como vista de agregación de todas las particiones de base de datos para todas las bases de datos activas que se encuentran en la misma instancia que contiene la base de datos conectada actualmente.

```
SELECT SUBSTR(DB_NAME, 1, 20) AS DB_NAME, DB_STATUS, SERVER_PLATFORM,
       DB_LOCATION, DB_CONN_TIME
       FROM TABLE(SNAP_GET_DB_V95(CAST (NULL AS VARCHAR(128)), -2)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
DB_NAME      DB_STATUS    SERVER_PLATFORM ...
-----
TOOLSDB     ACTIVE      AIX64          ...
SAMPLE     ACTIVE      AIX64          ...
```

Datos de salida de esta consulta (continuación).

```
... DB_LOCATION DB_CONN_TIME
... -----
... LOCAL      2005-07-24-22.26.54.396335
... LOCAL      2005-07-24-22.09.22.013196
```

Ejemplo 3: Esta rutina se puede utilizar realizando la siguiente llamada en la línea de mandatos:

Cuando está conectado a una base de datos:

```
SELECT TOTAL_OLAP_FUNCS, OLAP_FUNC_OVERFLOWS, ACTIVE_OLAP_FUNCS
       FROM TABLE (SNAP_GET_DB_V95(' ', 0)) AS T
```

La salida se parecerá a la siguiente:

```
TOTAL_OLAP_FUNCS  OLAP_FUNC_OVERFLOWS  ACTIVE_OLAP_FUNCS
-----
                7                2                1
```

1 registro(s) seleccionado(s).

Ejemplo 4: Después de ejecutar una carga de trabajo, el usuario puede utilizar la siguiente consulta con la función de tabla.

```
SELECT STATS_CACHE_SIZE, STATS_FABRICATIONS, SYNC_RUNSTATS,
       ASYNC_RUNSTATS, STATS_FABRICATE_TIME, SYNC_RUNSTATS_TIME
FROM TABLE (SNAP_GET_DB_V95('mytestdb', -1)) AS SNAPDB
```

```
STATS_CACHE_SIZE  STATS_FABRICATIONS  SYNC_RUNSTATS  ASYNC_RUNSTATS  ...
-----
                200                    1                2                0  ...
```

Continuación

```
...STATS_FABRICATE_TIME  SYNC_RUNSTATS_TIME
-----
...          2              32
```

1 registro(s) seleccionado(s).

Metadatos de la vista administrativa SNAPDB y la función de tabla SNAP_GET_DB_V95

Tabla 172. Información devuelta por la vista administrativa SNAPDB y la función de tabla SNAP_GET_DB_V95

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| DB_PATH | VARCHAR(1024) | db_path - Vía de acceso de base de datos |
| INPUT_DB_ALIAS | VARCHAR(128) | input_db_alias - Alias de base de datos de entrada |
| DB_STATUS | VARCHAR(12) | db_status - Estado de base de datos. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • ACTIVE • QUIESCE_PEND • QUIESCED • ROLLFWD |
| CATALOG_PARTITION | SMALLINT | catalog_node - Número de nodo de catálogo |
| CATALOG_PARTITION_NAME | VARCHAR(128) | catalog_node_name - Nombre de red de nodo de catálogo |

Tabla 172. Información devuelta por la vista administrativa SNAPDB y la función de tabla SNAP_GET_DB_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|---|
| SERVER_PLATFORM | VARCHAR(12) | <p>server_platform - Sistema operativo del servidor. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes:</p> <ul style="list-style-type: none"> • AIX • AIX64 • AS400_DRDA • DOS • DYNIX • HP • HP64 • HPIA • HPIA64 • LINUX • LINUX390 • LINUXIA64 • LINUXPPC • LINUXPPC64 • LINUXX8664 • LINUXZ64 • MAC • MVS_DRDA • NT • NT64 • OS2 • OS390 • SCO • SGI • SNI • SUN • SUN64 • UNKNOWN • UNKNOWN_DRDA • VM_DRDA • VSE_DRDA • WINDOWS |
| DB_LOCATION | VARCHAR(12) | <p>db_location - Ubicación de la base de datos. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h, y es uno de los siguientes:</p> <ul style="list-style-type: none"> • LOCAL • REMOTE |
| DB_CONN_TIME | TIMESTAMP | <p>db_conn_time - Indicación de fecha y hora de activación de base de datos</p> |

Tabla 172. Información devuelta por la vista administrativa SNAPDB y la función de tabla SNAP_GET_DB_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|--|
| LAST_RESET | TIMESTAMP | last_reset - Indicación de fecha y hora de último restablecimiento |
| LAST_BACKUP | TIMESTAMP | last_backup - Indicación de fecha y hora de última copia de seguridad |
| CONNECTIONS_TOP | BIGINT | connections_top - Número máximo de conexiones simultáneas |
| TOTAL_CONS | BIGINT | total_cons - Conexiones desde la activación de base de datos |
| TOTAL_SEC_CONS | BIGINT | total_sec_cons - Conexiones secundarias |
| APPLS_CUR_CONS | BIGINT | appls_cur_cons - Aplicaciones conectadas actualmente |
| APPLS_IN_DB2 | BIGINT | appls_in_db2 - Aplicaciones que se ejecutan en la base de datos actualmente |
| NUM_ASSOC_AGENTS | BIGINT | num_assoc_agents - Número de agentes asociados |
| AGENTS_TOP | BIGINT | agents_top - Número de agentes creados |
| COORD_AGENTS_TOP | BIGINT | coord_agents_top - Número máximo de agentes de coordinación |
| LOCKS_HELD | BIGINT | locks_held - Bloqueos retenidos |
| LOCK_WAITS | BIGINT | lock_waits - Esperas de bloqueo |
| LOCK_WAIT_TIME | BIGINT | lock_wait_time - Tiempo esperado en bloqueos |
| LOCK_LIST_IN_USE | BIGINT | lock_list_in_use - Memoria total de lista de bloqueos en uso |
| DEADLOCKS | BIGINT | deadlocks - Puntos muertos detectados |
| LOCK_ESCALS | BIGINT | lock_escals - Número de escalamientos de bloqueo |
| X_LOCK_ESCALS | BIGINT | x_lock_escals - Escalas de bloqueo exclusivas |
| LOCKS_WAITING | BIGINT | locks_waiting - Agentes actuales en espera de bloqueos |
| LOCK_TIMEOUTS | BIGINT | lock_timeouts - Número de tiempos de espera de bloqueo |
| NUM_INDOUBT_TRANS | BIGINT | num_indoubt_trans - Número de transacciones dudosas |
| SORT_HEAP_ALLOCATED | BIGINT | sort_heap_allocated - Total de pila de clasificación asignada |
| SORT_SHRHEAP_ALLOCATED | BIGINT | sort_shrheap_allocated - Pila de compartimiento de clasificación asignada actualmente |
| SORT_SHRHEAP_TOP | BIGINT | sort_shrheap_top - Marca de límite superior de pila de compartimiento de clasificación |
| POST_SHRTHRESHOLD_SORTS | BIGINT | post_shrthreshold_sorts - Clasificaciones de umbral compartidas posteriores |
| TOTAL_SORTS | BIGINT | total_sorts - Clasificaciones totales |

Tabla 172. Información devuelta por la vista administrativa SNAPDB y la función de tabla SNAP_GET_DB_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|--|
| TOTAL_SORT_TIME | BIGINT | total_sort_time - Tiempo de clasificación total |
| SORT_OVERFLOWS | BIGINT | sort_overflows - Desbordamientos de clasificación |
| ACTIVE_SORTS | BIGINT | active_sorts - Clasificaciones activas |
| POOL_DATA_L_READS | BIGINT | pool_data_l_reads - Lecturas lógicas de datos de la agrupación de almacenamientos intermedios |
| POOL_DATA_P_READS | BIGINT | pool_data_p_reads - Lecturas físicas de datos de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_L_READS | BIGINT | pool_temp_data_l_reads - Lecturas lógicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_P_READS | BIGINT | pool_temp_data_p_reads - Lecturas físicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_READS | BIGINT | pool_async_data_reads - Lecturas de datos asíncronas de agrupación de almacenamientos intermedios |
| POOL_DATA_WRITES | BIGINT | pool_data_writes - Grabaciones de datos de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_WRITES | BIGINT | pool_async_data_writes - Grabaciones de datos asíncronas de agrupación de almacenamientos intermedios |
| POOL_INDEX_L_READS | BIGINT | pool_index_l_reads - Lecturas lógicas de índices de la agrupación de almacenamientos intermedios |
| POOL_INDEX_P_READS | BIGINT | pool_index_p_reads - Lecturas físicas de índices de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_L_READS | BIGINT | pool_temp_index_l_reads - Lecturas lógicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_P_READS | BIGINT | pool_temp_index_p_reads - Lecturas físicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_READS | BIGINT | pool_async_index_reads - Lecturas de índice asíncronas de agrupación de almacenamientos intermedios |
| POOL_INDEX_WRITES | BIGINT | pool_index_writes - Grabaciones de índices de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_WRITES | BIGINT | pool_async_index_writes - Grabaciones de índice asíncronas de agrupación de almacenamientos intermedios |

Tabla 172. Información devuelta por la vista administrativa SNAPDB y la función de tabla SNAP_GET_DB_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|----------------------------|---------------|--|
| POOL_XDA_P_READS | BIGINT | pool_xda_p_reads - Lecturas físicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_XDA_L_READS | BIGINT | pool_xda_l_reads - Lecturas lógicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_XDA_WRITES | BIGINT | pool_xda_writes - Grabaciones de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_XDA_READS | BIGINT | pool_async_xda_reads - Lecturas de datos XDA asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_XDA_WRITES | BIGINT | pool_async_xda_writes - Grabaciones de datos XDA asíncronas de agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_P_READS | BIGINT | pool_temp_xda_p_reads - Elemento del supervisor de las lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_L_READS | BIGINT | pool_temp_xda_l_reads - Lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |
| POOL_READ_TIME | BIGINT | pool_read_time - Tiempo total de lectura física de la agrupación de almacenamientos intermedios |
| POOL_WRITE_TIME | BIGINT | pool_write_time - Tiempo total de grabación física de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_READ_TIME | BIGINT | pool_async_read_time - Tiempo de lectura asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_WRITE_TIME | BIGINT | pool_async_write_time - Tiempo de grabación asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_READ_REQS | BIGINT | pool_async_data_read_reqs - Peticiones de lectura asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_READ_REQS | BIGINT | pool_async_index_read_reqs - Peticiones de lectura de índice asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_XDA_READ_REQS | BIGINT | pool_async_xda_read_reqs - Peticiones de lectura XDA asíncrona de agrupación de almacenamientos intermedios |
| POOL_NO_VICTIM_BUFFER | BIGINT | pool_no_victim_buffer - Almacenamiento intermedio sin víctimas de la agrupación de almacenamientos intermedios |

Tabla 172. Información devuelta por la vista administrativa SNAPDB y la función de tabla SNAP_GET_DB_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|--|
| POOL_LSN_GAP_CLNS | BIGINT | pool_lsn_gap_clns - Limpiadores de espacio de anotación cronológica de agrupación de almacenamientos intermedios activados |
| POOL_DRTY_PG_STEAL_CLNS | BIGINT | pool_drt_y_pg_steal_clns - Limpiadores de páginas víctima de agrupación de agrupación de almacenamientos intermedios activados |
| POOL_DRTY_PG_THRSH_CLNS | BIGINT | pool_drt_y_pg_thrsh_clns - Limpiadores de umbral de agrupación de almacenamientos intermedios activados |
| PREFETCH_WAIT_TIME | BIGINT | prefetch_wait_time - Tiempo esperado para captación previa |
| UNREAD_PREFETCH_PAGES | BIGINT | unread_prefetch_pages - Páginas de captación previa no leídas |
| DIRECT_READS | BIGINT | direct_reads - Lecturas directas de la base de datos |
| DIRECT_WRITES | BIGINT | direct_writes - Grabaciones directas en la base de datos |
| DIRECT_READ_REQS | BIGINT | direct_read_reqs - Peticiones de lectura directa |
| DIRECT_WRITE_REQS | BIGINT | direct_write_reqs - Peticiones de grabación directa |
| DIRECT_READ_TIME | BIGINT | direct_read_time - Tiempo de lectura directa |
| DIRECT_WRITE_TIME | BIGINT | direct_write_time - Tiempo de grabación directa |
| FILES_CLOSED | BIGINT | files_closed - Archivos de base de datos cerrados |
| ELAPSED_EXEC_TIME_S | BIGINT | elapsed_exec_time - Tiempo transcurrido de ejecución de sentencia |
| ELAPSED_EXEC_TIME_MS | BIGINT | elapsed_exec_time - Tiempo transcurrido de ejecución de sentencia |
| COMMIT_SQL_STMTS | BIGINT | commit_sql_stmts - Sentencias de confirmación intentadas |
| ROLLBACK_SQL_STMTS | BIGINT | rollback_sql_stmts - Sentencias de retrotracción intentadas |
| DYNAMIC_SQL_STMTS | BIGINT | dynamic_sql_stmts - Sentencias de SQL dinámico intentadas |
| STATIC_SQL_STMTS | BIGINT | static_sql_stmts - Sentencias de SQL estático intentadas |
| FAILED_SQL_STMTS | BIGINT | failed_sql_stmts - Operaciones de sentencia anómalas |
| SELECT_SQL_STMTS | BIGINT | select_sql_stmts - Sentencias Select de SQL ejecutadas |
| UID_SQL_STMTS | BIGINT | uid_sql_stmts - Sentencias UPDATE/INSERT/DELETE de SQL ejecutadas |

Tabla 172. Información devuelta por la vista administrativa SNAPDB y la función de tabla SNAP_GET_DB_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------|---------------|---|
| DDL_SQL_STMTS | BIGINT | ddl_sql_stmts - Sentencias de lenguaje de definición de datos (DDL) de SQL |
| INT_AUTO_REBINDS | BIGINT | int_auto_rebinds - Revinculaciones automáticas internas |
| INT_ROWS_DELETED | BIGINT | int_rows_deleted - Filas internas suprimidas |
| INT_ROWS_INSERTED | BIGINT | int_rows_inserted - Filas internas insertadas |
| INT_ROWS_UPDATED | BIGINT | int_rows_updated - Filas internas actualizadas |
| INT_COMMITS | BIGINT | int_commits - Confirmaciones internas |
| INT_ROLLBACKS | BIGINT | int_rollback - Retrotracciones internas |
| INT_DEADLOCK_ROLLBACKS | BIGINT | int_deadlock_rollback - Retrotracciones internas debido a punto muerto |
| ROWS_DELETED | BIGINT | rows_deleted - Filas suprimidas |
| ROWS_INSERTED | BIGINT | rows_inserted - Filas insertadas |
| ROWS_UPDATED | BIGINT | rows_updated - Filas actualizadas |
| ROWS_SELECTED | BIGINT | rows_selected - Filas seleccionadas |
| ROWS_READ | BIGINT | rows_read - Filas leídas |
| BINDS_PRECOMPILES | BIGINT | binds_precompiles - Vinculaciones/precompilaciones intentadas |
| TOTAL_LOG_AVAILABLE | BIGINT | total_log_available - Anotaciones cronológicas totales disponibles |
| TOTAL_LOG_USED | BIGINT | total_log_used - Espacio total de anotaciones cronológicas utilizado |
| SEC_LOG_USED_TOP | BIGINT | sec_log_used_top - Espacio máximo de anotaciones cronológicas secundarias utilizado |
| TOT_LOG_USED_TOP | BIGINT | tot_log_used_top - Espacio máximo de anotaciones cronológicas totales utilizado |
| SEC_LOGS_ALLOCATED | BIGINT | sec_logs_allocated - Anotaciones cronológicas secundarias asignadas actualmente |
| LOG_READS | BIGINT | log_reads - Número de páginas de anotaciones cronológicas leídas |
| LOG_READ_TIME_S | BIGINT | log_read_time - Tiempo de lectura de anotaciones cronológicas |
| LOG_READ_TIME_NS | BIGINT | log_read_time - Tiempo de lectura de anotaciones cronológicas |
| LOG_WRITES | BIGINT | log_writes - Número de páginas de anotaciones cronológicas escritas |
| LOG_WRITE_TIME_S | BIGINT | log_write_time - Tiempo de grabación de anotaciones cronológicas |
| LOG_WRITE_TIME_NS | BIGINT | log_write_time - Tiempo de grabación de anotaciones cronológicas |
| NUM_LOG_WRITE_IO | BIGINT | num_log_write_io - Número de grabaciones de anotaciones cronológicas |

Tabla 172. Información devuelta por la vista administrativa SNAPDB y la función de tabla SNAP_GET_DB_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------------|---------------|--|
| NUM_LOG_READ_IO | BIGINT | num_log_read_io - Número de lecturas de anotaciones cronológicas |
| NUM_LOG_PART_PAGE_IO | BIGINT | num_log_part_page_io - Número de grabaciones de páginas de anotaciones cronológicas parciales |
| NUM_LOG_BUFFER_FULL | BIGINT | num_log_buffer_full - Número de almacenamientos intermedios de anotaciones cronológicas llenos |
| NUM_LOG_DATA_FOUND_IN_BUFFER | BIGINT | num_log_data_found_in_buffer - Número de datos de anotaciones cronológicas encontrados en el almacenamiento intermedio |
| APPL_ID_OLDEST_XACT | BIGINT | appl_id_oldest_xact - Aplicación con la transacción más antigua |
| LOG_TO_REDO_FOR_RECOVERY | BIGINT | log_to_redo_for_recovery - Cantidad de anotaciones cronológicas a rehacer para la recuperación |
| LOG_HELD_BY_DIRTY_PAGES | BIGINT | log_held_by_dirty_pages - Cantidad de espacio de anotaciones cronológicas contabilizado por páginas sucias |
| PKG_CACHE_LOOKUPS | BIGINT | pkg_cache_lookups - Búsquedas de antememoria de paquete |
| PKG_CACHE_INSERTS | BIGINT | pkg_cache_inserts - Inserciones de antememoria de paquete |
| PKG_CACHE_NUM_OVERFLOWS | BIGINT | pkg_cache_num_overflows - Desbordamientos de antememoria de paquete |
| PKG_CACHE_SIZE_TOP | BIGINT | pkg_cache_size_top - Marca de límite superior de antememoria de paquete |
| APPL_SECTION_LOOKUPS | BIGINT | appl_section_lookups - Búsquedas de sección |
| APPL_SECTION_INSERTS | BIGINT | appl_section_inserts - Inserciones de sección |
| CAT_CACHE_LOOKUPS | BIGINT | cat_cache_lookups - Búsquedas de antememoria de catálogo |
| CAT_CACHE_INSERTS | BIGINT | cat_cache_inserts - Inserciones de antememoria de catálogo |
| CAT_CACHE_OVERFLOWS | BIGINT | cat_cache_overflows - Desbordamientos de antememoria de catálogo |
| CAT_CACHE_SIZE_TOP | BIGINT | cat_cache_size_top - Marca de límite superior de antememoria de catálogo |
| PRIV_WORKSPACE_SIZE_TOP | BIGINT | priv_workspace_size_top - Tamaño máximo de espacio de trabajo privado |
| PRIV_WORKSPACE_NUM_OVERFLOWS | BIGINT | priv_workspace_num_overflows - Desbordamientos de espacio de trabajo privado |

Tabla 172. Información devuelta por la vista administrativa SNAPDB y la función de tabla SNAP_GET_DB_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------------------|---------------|--|
| PRIV_WORKSPACE_SECTION_INSERTS | BIGINT | priv_workspace_section_inserts - Inserciones de sección de espacio de trabajo privado |
| PRIV_WORKSPACE_SECTION_LOOKUPS | BIGINT | priv_workspace_section_lookups - Búsquedas de sección de espacio de trabajo privado |
| SHR_WORKSPACE_SIZE_TOP | BIGINT | shr_workspace_size_top - Tamaño máximo de espacio de trabajo compartido |
| SHR_WORKSPACE_NUM_OVERFLOWS | BIGINT | shr_workspace_num_overflows - Desbordamientos de espacio de trabajo compartido |
| SHR_WORKSPACE_SECTION_INSERTS | BIGINT | shr_workspace_section_inserts - Inserciones de sección de espacio de trabajo compartido |
| SHR_WORKSPACE_SECTION_LOOKUPS | BIGINT | shr_workspace_section_lookups - Búsquedas de sección de espacio de trabajo compartido |
| TOTAL_HASH_JOINS | BIGINT | total_hash_joins - Total de uniones hash |
| TOTAL_HASH_LOOPS | BIGINT | total_hash_loops - Total de bucles hash |
| HASH_JOIN_OVERFLOWS | BIGINT | hash_join_overflows - Desbordamientos de uniones hash |
| HASH_JOIN_SMALL_OVERFLOWS | BIGINT | hash_join_small_overflows - Desbordamientos pequeños de uniones hash |
| POST_SHRTHRESHOLD_HASH_JOINS | BIGINT | post_shrthreshold_hash_joins - Uniones hash de umbral posteriores |
| ACTIVE_HASH_JOINS | BIGINT | active_hash_joins - Uniones hash activas |
| NUM_DB_STORAGE_PATHS | BIGINT | num_db_storage_paths - Número de vías de acceso de almacenamiento automático |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |
| SMALLEST_LOG_AVAIL_NODE | INTEGER | smallest_log_avail_node - Nodo con menos espacio de anotaciones cronológicas disponible |
| TOTAL_OLAP_FUNCS | BIGINT | Número total de funciones OLAP ejecutadas. |
| OLAP_FUNC_OVERFLOWS | BIGINT | Número de veces que los datos de función OLAP han superado el espacio de pila de clasificación disponible. |
| ACTIVE_OLAP_FUNCS | BIGINT | Número total de funciones OLAP que se están ejecutando actualmente y que están consumiendo memoria de pila de clasificación. |
| STATS_CACHE_SIZE | BIGINT | Tamaño de antememoria de estadísticas en bytes |

Tabla 172. Información devuelta por la vista administrativa SNAPDB y la función de tabla SNAP_GET_DB_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------------|---------------|--|
| STATS_FABRICATIONS | BIGINT | Número total de actividades de recopilación de estadísticas para que el sistema cree estadísticas sin exploración de tabla ni de índice. |
| SYNC_RUNSTATS | BIGINT | Número total de actividades de recopilación de estadísticas síncronas durante compilación de consultas. |
| ASYNC_RUNSTATS | BIGINT | Cambiaremos la salida de esta columna por el número total de actividades asíncronas satisfactorias de recopilación de estadísticas. |
| STATS_FABRICATE_TIME | BIGINT | Tiempo total que el sistema ha empleado en crear estadísticas sin exploración de tabla o índice durante la compilación de consultas en milisegundos. |
| SYNC_RUNSTATS_TIME | BIGINT | Tiempo total empleado en actividades síncronas de recopilación de estadísticas en milisegundos. |
| NUM_THRESHOLD_VIOLATIONS | BIGINT | Número de violaciones de umbral sucedidas en la base de datos. |

Vista administrativa SNAPDB_MEMORY_POOL y función de tabla SNAP_GET_DB_MEMORY_POOL – Recuperar información sobre el uso de la memoria en el nivel de la base de datos

La vista administrativa SNAPDB_MEMORY_POOL y la función de tabla SNAP_GET_DB_MEMORY_POOL devuelven información acerca del uso de la memoria en el nivel de la base de datos únicamente para plataformas de UNIX.

Vista administrativa SNAPDB_MEMORY_POOL

Esta vista administrativa permite recuperar la información sobre el uso de la memoria en el nivel de la base de datos para la base de datos conectada actualmente.

La vista administrativa SNAPDB_MEMORY_POOL, utilizada con las vistas administrativas SNAPDB, SNAPDETAILLOG, SNAPHADR y SNAPSTORAGE_PATHS, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR DATABASE ON alias de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 143 en la página 570 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPDB_MEMORY_POOL

- Privilegio CONTROL en la vista administrativa SNAPDB_MEMORY_POOL
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_DB_MEMORY_POOL
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar una lista de agrupaciones de memoria y su tamaño actual para la base de datos conectada actualmente. SAMPLE.

```
SELECT POOL_ID,
POOL_CUR_SIZE FROM SYSIBMADM.SNAPDB_MEMORY_POOL
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| POOL_ID | POOL_CUR_SIZE |
|---------------|---------------|
| UTILITY | 32768 |
| PACKAGE_CACHE | 475136 |
| CAT_CACHE | 65536 |
| BP | 2097152 |
| BP | 1081344 |
| BP | 540672 |
| BP | 278528 |
| BP | 147456 |
| BP | 81920 |
| LOCK_MGR | 294912 |
| DATABASE | 3833856 |
| OTHER | 0 |

12 registro(s) seleccionado(s).

SNAP_GET_DB_MEMORY_POOL, función de tabla

La función de tabla SNAP_GET_DB_MEMORY_POOL devuelve la misma información que la vista administrativa SNAPDB_MEMORY_POOL, pero permite recuperar la información para una base de datos concreta en una partición específica de la base de datos, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_DB_MEMORY_POOL, utilizada con las funciones de tabla SNAP_GET_DB_V95, SNAP_GET_DETAILLOG_V91, SNAP_GET_HADR y SNAP_GET_STORAGE_PATHS, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR ALL DATABASES.

Consulte la Tabla 143 en la página 570 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
▶▶ SNAP_GET_DB_MEMORY_POOL ( ( nombrebd [ , númparticiónbd ] ) ) ▶▶
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique una serie vacía para tomar la instantánea desde la base de datos conectada actualmente. Especifique un valor NULL para tomar la instantánea desde todas las bases de datos que se encuentran en la misma instancia que la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_DB_MEMORY_POOL toma una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_DB_MEMORY_POOL
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar una lista de agrupaciones de memoria y su tamaño actual para todas las bases de datos.

```
SELECT SUBSTR(DB_NAME,1,8) AS DB_NAME, POOL_ID, POOL_CUR_SIZE  
FROM TABLE(SNAPSHOT_GET_DB_MEMORY_POOL  
(CAST(NULL AS VARCHAR(128)), -1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| DB_NAME | POOL_ID | POOL_CUR_SIZE |
|---------|---------------|---------------|
| TESTDB | UTILITY | 65536 |
| TESTDB | PACKAGE_CACHE | 851968 |
| TESTDB | CAT_CACHE | 65536 |
| TESTDB | BP | 35913728 |
| TESTDB | BP | 589824 |
| TESTDB | BP | 327680 |
| TESTDB | BP | 196608 |
| TESTDB | BP | 131072 |
| TESTDB | SHARED_SORT | 65536 |
| TESTDB | LOCK_MGR | 10092544 |
| TESTDB | DATABASE | 4980736 |
| TESTDB | OTHER | 196608 |
| SAMPLE | UTILITY | 65536 |
| SAMPLE | PACKAGE_CACHE | 655360 |
| SAMPLE | CAT_CACHE | 131072 |
| SAMPLE | BP | 4325376 |
| SAMPLE | BP | 589824 |
| SAMPLE | BP | 327680 |
| SAMPLE | BP | 196608 |
| SAMPLE | BP | 131072 |
| SAMPLE | SHARED_SORT | 0 |
| SAMPLE | LOCK_MGR | 655360 |
| SAMPLE | DATABASE | 4653056 |
| SAMPLE | OTHER | 196608 |

24 registro(s) seleccionado(s).

Información devuelta

Tabla 173. Información devuelta por la vista administrativa `SNAPDB_MEMORY_POOL` y la función de tabla `SNAP_GET_DB_MEMORY_POOL`

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |

Tabla 173. Información devuelta por la vista administrativa SNAPDB_MEMORY_POOL y la función de tabla SNAP_GET_DB_MEMORY_POOL (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|--|
| POOL_ID | VARCHAR(14) | pool_id - Identificador de agrupación de memoria. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • APP_GROUP • APPL_CONTROL • APPLICATION • BP • CAT_CACHE • DATABASE • DFM • FCMBP • IMPORT_POOL • LOCK_MGR • MONITOR • OTHER • PACKAGE_CACHE • QUERY • SHARED_SORT • SORT • STATEMENT • STATISTICS • UTILITY |
| POOL_SECONDARY_ID | VARCHAR(32) | pool_secondary_id - Identificador secundario de agrupación de memoria |
| POOL_CUR_SIZE | BIGINT | pool_cur_size - Tamaño actual de la agrupación de memoria |
| POOL_WATERMARK | BIGINT | pool_watermark - Marca límite de agrupación de memoria |
| POOL_CONFIG_SIZE | BIGINT | pool_config_size - Tamaño configurado de la agrupación de memoria |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa SNAPDBM y función de tabla SNAP_GET_DBM_V95 – Recuperar la información de instantáneas del grupo de datos lógicos dbm

La vista administrativa SNAPDBM y la función de tabla SNAP_GET_DBM_V95 devuelven información acerca de los grupos lógicos del gestor de bases de datos DB2 (dbm) y del supervisor de instantáneas.

Vista administrativa SNAPDBM

La vista administrativa SNAPDBM, utilizada con las vistas administrativas SNAPDBM_MEMORY_POOL, SNAPFCM, SNAPFCM_PART y SNAPSWITCHES, ofrece los datos equivalentes al mandato GET SNAPSHOT FOR DBM.

El esquema es SYSIBMADM.

Consulte la Tabla 144 en la página 574 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPDBM
- Privilegio CONTROL en la vista administrativa SNAPDBM
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_DBM_V95
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar estado del gestor de bases de datos e información sobre la conexión para todas las particiones de base de datos.

```
SELECT DB2_STATUS, DB2START_TIME, LAST_RESET, LOCAL_CONS, REM_CONS_IN,  
       (AGENTS_CREATED_EMPTY_POOL/AGENTS_FROM_POOL) AS AGENT_USAGE,  
       DBPARTITIONNUM FROM SYSIBMADM.SNAPDBM ORDER BY DBPARTITIONNUM
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| DB2_STATUS | DB2START_TIME | LAST_RESET | ... |
|------------|----------------------------|------------|-------|
| ACTIVE | 2006-01-06-14.59.59.059879 | | - ... |
| ACTIVE | 2006-01-06-14.59.59.097605 | | - ... |
| ACTIVE | 2006-01-06-14.59.59.062798 | | - ... |

3 registro(s) seleccionado(s). ...

Datos de salida de esta consulta (continuación).

| ... | LOCAL_CONS | REM_CONS_IN | AGENT_USAGE | DBPARTITIONNUM |
|-----|------------|-------------|-------------|----------------|
| ... | 1 | 1 | 0 | 0 |
| ... | 0 | 0 | 0 | 1 |
| ... | 0 | 0 | 0 | 2 |

SNAP_GET_DBM_V95, función de tabla

La función de tabla SNAP_GET_DBM_V95 devuelve la misma información que la vista administrativa SNAPDBM, pero le permite recuperar la información para una partición de base de datos específica, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_DBM_V95, utilizada con las funciones de tabla SNAP_GET_DBM_MEMORY_POOL, SNAP_GET_FCM, SNAP_GET_FCM_PART y SNAP_GET_SWITCHES, ofrece los datos equivalentes al mandato GET SNAPSHOT FOR DBM.

Consulte la Tabla 144 en la página 574 para ver una lista completa de la información que se puede devolver.

Sintaxis

```

▶▶ SNAP_GET_DBM_V95 ( [ númparticiónbd ] )

```

El esquema es SYSPROC.

Parámetro de la función de tabla

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si no se utiliza esta opción de entrada, se devolverán datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *núm_partición_bd* se establece en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_DBM_V95 toma la instantánea de la memoria.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_DBM_V95
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar la hora de inicio y el estado actual del número 2 de partición de base de datos.

```
SELECT DB2START_TIME, DB2_STATUS FROM TABLE(SNAP_GET_DBM_V95(2)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
DB2START_TIME          DB2_STATUS
-----
2006-01-06-14.59.59.062798 ACTIVE
```

Información devuelta

Tabla 174. Información que devuelve la vista administrativa SNAPDBM y la función de tabla SNAP_GET_DBM_V95

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| SORT_HEAP_ALLOCATED | BIGINT | sort_heap_allocated - Total de pila de clasificación asignada |
| POST_THRESHOLD_SORTS | BIGINT | post_threshold_sorts - Clasificaciones de umbral posteriores |
| PIPED_SORTS_REQUESTED | BIGINT | piped_sorts_requested - Clasificaciones mediante conexiones solicitadas |
| PIPED_SORTS_ACCEPTED | BIGINT | piped_sorts_accepted - Clasificaciones mediante conexiones aceptadas |
| REM_CONS_IN | BIGINT | rem_cons_in - Conexiones remotas con gestor de bases de datos |
| REM_CONS_IN_EXEC | BIGINT | rem_cons_in_exec - Conexiones remotas que se ejecutan en el elemento de supervisor de gestor de bases de datos |
| LOCAL_CONS | BIGINT | local_cons - Conexiones locales |
| LOCAL_CONS_IN_EXEC | BIGINT | local_cons_in_exec - Conexiones locales que se ejecutan en el elemento de supervisor de gestor de bases de datos |
| CON_LOCAL_DBASES | BIGINT | con_local_dbases - Bases de datos locales con conexiones actuales |
| AGENTS_REGISTERED | BIGINT | agents_registered - Agentes registrados |
| AGENTS_WAITING_ON_TOKEN | BIGINT | agents_waiting_on_token - Agentes en espera de una señal |

Tabla 174. Información que devuelve la vista administrativa SNAPDBM y la función de tabla SNAP_GET_DBM_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------------|---------------|--|
| DB2_STATUS | VARCHAR(12) | db2_status - Estado de instancia de DB2 Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • ACTIVE • QUIESCE_PEND • QUIESCED |
| AGENTS_REGISTERED_TOP | BIGINT | agents_registered_top - Número máximo de agentes registrados |
| AGENTS_WAITING_TOP | BIGINT | agents_waiting_top - Número máximo de agentes en espera |
| COMM_PRIVATE_MEM | BIGINT | comm_private_mem - Memoria privada confirmada |
| IDLE_AGENTS | BIGINT | idle_agents - Número de agentes desocupados |
| AGENTS_FROM_POOL | BIGINT | agents_from_pool - Agentes asignados de la agrupación |
| AGENTS_CREATED_EMPTY_POOL | BIGINT | agents_created_empty_pool - Agentes creados porque la agrupación de agentes está vacía |
| COORD_AGENTS_TOP | BIGINT | coord_agents_top - Número máximo de agentes de coordinación |
| MAX_AGENT_OVERFLOW | BIGINT | max_agent_overflows - Número máximo de desbordamientos de agente |
| AGENTS_STOLEN | BIGINT | agents_stolen - Agentes robados |
| GW_TOTAL_CONS | BIGINT | gw_total_cons - Número total de conexiones intentadas para DB2 Connect |
| GW_CUR_CONS | BIGINT | gw_cur_cons - Número actual de conexiones para DB2 Connect |
| GW_CONS_WAIT_HOST | BIGINT | gw_cons_wait_host - Número de conexiones en espera para que conteste el sistema principal |
| GW_CONS_WAIT_CLIENT | BIGINT | gw_cons_wait_client - Número de conexiones en espera para que el cliente envíe petición |
| POST_THRESHOLD_HASH_JOINS | BIGINT | post_threshold_hash_joins - Umbral de uniones hash |
| NUM_GW_CONN_SWITCHES | BIGINT | num_gw_conn_switches - Conmutadores de conexión |
| DB2START_TIME | TIMESTAMP | db2start_time - Indicación de fecha y hora de inicio de gestor de bases de datos |
| LAST_RESET | TIMESTAMP | last_reset - Indicación de fecha y hora de último restablecimiento |
| NUM_NODES_IN_DB2_INSTANCE | INTEGER | num_nodes_in_db2_instance - Número de nodos en la partición de base de datos |

Tabla 174. Información que devuelve la vista administrativa SNAPDBM y la función de tabla SNAP_GET_DBM_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------------|---------------|--|
| PRODUCT_NAME | VARCHAR(32) | product_name - Nombre del producto |
| SERVICE_LEVEL | VARCHAR(18) | service_level - Nivel de servicio |
| SORT_HEAP_TOP | BIGINT | sort_heap_top - Marca de límite superior de pila privada de clasificación |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |
| POST_THRESHOLD_OLAP_FUNCS | BIGINT | <p>Número de funciones OLAP que han solicitado una pila de clasificación una vez superado el umbral de la pila de clasificación.</p> <p>Las clasificaciones, las uniones de generación aleatoria y las funciones OLAP son ejemplos de operaciones que utilizan una pila de clasificación. Bajo condiciones normales, el gestor de bases de dato asignará pila de clasificación utilizando el valor especificado por el parámetro de configuración sortheap. Si la cantidad de memoria asignada a las pilas de clasificación supera el umbral de pila de clasificación (parámetro de configuración sheapthres), el gestor de bases de datos asignará las siguientes pilas de clasificación utilizando un valor menor que el especificado por el parámetro de configuración sortheap.</p> <p>Es posible que las funciones OLAP que empiezan después de que se alcance el umbral de pila de clasificación no reciban la cantidad óptima de memoria para su ejecución.</p> |

Vista administrativa SNAPDBM_MEMORY_POOL y función de tabla SNAP_GET_DBM_MEMORY_POOL – Recuperar información sobre el uso de la memoria en el nivel del gestor de bases de datos

La vista administrativa SNAPDBM_MEMORY_POOL y la tabla de función SNAP_GET_DBM_MEMORY_POOL devuelven información acerca del uso de la memoria en el gestor de bases de datos.

Vista administrativa SNAPDBM_MEMORY_POOL

La vista administrativa SNAPDBM_MEMORY_POOL, utilizada con las vistas administrativas SNAPDBM, SNAPFCM, SNAPFCM_PART y SNAPSWITCHES, ofrece los datos equivalentes al mandato GET SNAPSHOT FOR DBM.

El esquema es SYSIBMADM.

Consulte la Tabla 145 en la página 579 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPDBM_MEMORY_POOL
- Privilegio CONTROL en la vista administrativa SNAPDBM_MEMORY_POOL
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE en la función de tabla SNAP_GET_DBM_MEMORY_POOL
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMAINT
- SYSADM

Ejemplo

Recuperar una lista de agrupaciones de memoria y su tamaño actual para el gestor de bases de datos de la base de datos conectada.

```
SELECT POOL_ID, POOL_CUR_SIZE FROM SNAPDBM_MEMORY_POOL
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
POOL_ID          POOL_CUR_SIZE
-----
MONITOR                    65536
OTHER                    29622272
FCMBP                    57606144
...
```

SNAP_GET_DBM_MEMORY_POOL, función de tabla

La función de tabla SNAP_GET_DBM_MEMORY_POOL devuelve la misma información que la vista administrativa SNAPDBM_MEMORY_POOL, pero permite recuperar la información de una partición específica de la base de datos, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_DBM_MEMORY_POOL, utilizada con las funciones de tabla SNAP_GET_DBM_V95, SNAP_GET_FCM, SNAP_GET_FCM_PART y SNAP_GET_SWITCHES, ofrece los datos equivalentes al mandato GET SNAPSHOT FOR DBM.

Consulte la Tabla 145 en la página 579 para ver una lista completa de la información que se puede devolver.

Sintaxis

► SNAP_GET_DBM_MEMORY_POOL (núm-partición-bd) ◄

El esquema es SYSPROC.

Parámetro de la función de tabla

núm-partición-bd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si no se utiliza esta opción de entrada, se devolverán datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *núm-partición-bd* se establece en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_DBM_MEMORY_POOL toma una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE en la función de tabla SNAP_GET_DBM_MEMORY_POOL
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar una lista de agrupaciones de memoria y su tamaño actual para todas las particiones de base de datos del gestor de bases de datos, de la base de datos conectada.

```
SELECT POOL_ID, POOL_CUR_SIZE, DBPARTITIONNUM  
FROM TABLE(SYSPROC.SNAP_GET_DBM_MEMORY_POOL())  
AS T ORDER BY DBPARTITIONNUM
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| POOL_ID | POOL_CUR_SIZE | DBPARTITIONNUM |
|---------|---------------|----------------|
| MONITOR | 65536 | 0 |
| OTHER | 29622272 | 0 |
| FCMBP | 57606144 | 0 |
| MONITOR | 65536 | 1 |

| | | |
|---------|----------|---|
| OTHER | 29425664 | 1 |
| FCMBP | 57606144 | 1 |
| MONITOR | 65536 | 2 |
| OTHER | 29425664 | 2 |
| FCMBP | 57606144 | 2 |

Información devuelta

Tabla 175. Información que devuelve la vista administrativa SNAPDBM_MEMORY_POOL y la función de tabla SNAP_GET_DBM_MEMORY_POOL

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| POOL_ID | VARCHAR(14) | pool_id - Identificador de agrupación de memoria. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • APP_GROUP • APPL_CONTROL • APPLICATION • BP • CAT_CACHE • DATABASE • DFM • FCMBP • IMPORT_POOL • LOCK_MGR • MONITOR • OTHER • PACKAGE_CACHE • QUERY • SHARED_SORT • SORT • STATEMENT • STATISTICS • UTILITY |
| POOL_CUR_SIZE | BIGINT | pool_cur_size - Tamaño actual de la agrupación de memoria |
| POOL_WATERMARK | BIGINT | pool_watermark - Marca límite de agrupación de memoria |
| POOL_CONFIG_SIZE | BIGINT | pool_config_size - Tamaño configurado de la agrupación de memoria |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa SNAPDETAILLOG y función de tabla SNAP_GET_DETAILLOG_V91 - Recuperar información de instantánea del grupo de datos lógicos detail_log

La vista administrativa SNAPDETAILLOG y la función de tabla SNAP_GET_DETAILLOG_V91 devuelven información de la instantánea del grupo de datos lógicos detail_log.

SNAPDETAILLOG, vista administrativa

Esta vista administrativa le permite recuperar información de instantánea desde el grupo de datos lógicos detail_log para la base de datos conectada actualmente.

La vista administrativa SNAPDETAILLOG, que se utiliza junto con las vistas administrativas SNAPDB, SNAPDB_MEMORY_POOL, SNAPHDR y SNAPSTORAGE_PATHS, proporciona información equivalente a la que proporciona el mandato CLP GET SNAPSHOT FOR DATABASE ON alias de base de datos.

El esquema es SYSIBMADM.

En la Tabla 146 en la página 582 encontrará una lista completa de la información que se devuelve.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPDETAILLOG
- Privilegio CONTROL en la vista administrativa SNAPDETAILLOG
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_DETAILLOG_V91
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar información acerca de las anotaciones cronológicas para todas las particiones de base de datos para la base de datos conectada actualmente.

```
SELECT SUBSTR(DB_NAME, 1, 8) AS DB_NAME, FIRST_ACTIVE_LOG,  
       LAST_ACTIVE_LOG, CURRENT_ACTIVE_LOG, CURRENT_ARCHIVE_LOG,  
       DBPARTITIONNUM  
FROM SYSIBMADM.SNAPDETAILLOG ORDER BY DBPARTITIONNUM
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| DB_NAME | FIRST_ACTIVE_LOG | LAST_ACTIVE_LOG | ... |
|---------|------------------|-----------------|-----|
| TEST | 0 | 8 | ... |
| TEST | 0 | 8 | ... |
| TEST | 0 | 8 | ... |

3 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

| ... | CURRENT_ACTIVE_LOG | CURRENT_ARCHIVE_LOG | DBPARTITIONNUM |
|-----|--------------------|---------------------|----------------|
| ... | 0 | - | 0 |
| ... | 0 | - | 1 |
| ... | 0 | - | 2 |

Función de tabla SNAP_GET_DETAILLOG_V91

La función de tabla SNAP_GET_DETAILLOG_V91 devuelve la misma información que la vista administrativa SNAPDETAILLOG.

La función de tabla SNAP_GET_DETAILLOG, que se utiliza junto con las funciones de tabla SNAP_GET_DB_V95, SNAP_GET_DB_MEMORY_POOL, SNAP_GET_HADR y SNAP_GET_STORAGE_PATHS, proporciona información que es equivalente a la que proporciona el mandato CLP GET SNAPSHOT FOR ALL DATABASES.

En la Tabla 146 en la página 582 encontrará una lista completa de la información que se devuelve.

Sintaxis

```

▶▶ SNAP_GET_DETAILLOG_V91 ( ( nombrebd , numparticiónbd ) )

```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique una serie vacía para tomar la instantánea desde la base de datos conectada actualmente. Especifique un valor NULL para tomar la instantánea desde todas las bases de datos que se encuentran en la misma instancia que la base de datos conectada actualmente.

numparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *numparticiónbd* se establece en NULL, -1 se establece implícitamente para *numparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_DETAILLOG_V91 tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_DETAILLOG_V91
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar información acerca de las anotaciones cronológicas para la partición de base de datos 1 para la base de datos conectada actualmente.

```
SELECT SUBSTR(DB_NAME, 1, 8) AS DB_NAME, FIRST_ACTIVE_LOG,
       LAST_ACTIVE_LOG, CURRENT_ACTIVE_LOG, CURRENT_ARCHIVE_LOG
FROM TABLE(SNAP_GET_DETAILLOG_V91(' ', 1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
DB_NAME FIRST_ACTIVE_LOG LAST_ACTIVE_LOG ...
-----
TEST          0          8 ...
```

1 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```
... CURRENT_ACTIVE_LOG CURRENT_ARCHIVE_LOG
... -----
...          0          -
```

Metadatos de la vista administrativa SNAPDETAILLOG y la función de tabla SNAP_GET_DETAILLOG_V91

Tabla 176. Información devuelta por la vista administrativa SNAPDETAILLOG y la función de tabla SNAP_GET_DETAILLOG_V91

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |

Tabla 176. Información devuelta por la vista administrativa SNAPDETAILLOG y la función de tabla SNAP_GET_DETAILLOG_V91 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------|---------------|---|
| FIRST_ACTIVE_LOG | BIGINT | first_active_log - Primer número de archivo de anotaciones cronológicas activo |
| LAST_ACTIVE_LOG | BIGINT | last_active_log - Último número de archivo de anotaciones cronológicas activo |
| CURRENT_ACTIVE_LOG | BIGINT | current_active_log - Número de archivo de anotaciones cronológicas activo actual |
| CURRENT_ARCHIVE_LOG | BIGINT | current_archive_log - Número de archivo de anotaciones cronológicas de archivado actual |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa SNAPDYN_SQL y función de tabla SNAP_GET_DYN_SQL_V95 – Recuperar información de instantáneas del grupo de datos lógicos dynsql

La “SNAPDYN_SQL, vista administrativa” en la página 583 y la “Función de tabla SNAP_GET_DYN_SQL_V95” en la página 584 devuelven información de instantánea desde el grupo de datos lógicos dynsql.

SNAPDYN_SQL, vista administrativa

Esta vista administrativa permite recuperar la información de instantáneas del grupo lógico dynsql para la base de datos conectada actualmente.

Esta vista proporciona información equivalente a la que proporciona el mandato CLP GET SNAPSHOT FOR DYNAMIC SQL ON alias de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 147 en la página 587 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPDYN_SQL
- Privilegio CONTROL en la vista administrativa SNAPDYN_SQL
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_DYN_SQL_V95
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMMAINT
- SYSADM

Ejemplo

Recuperar una lista de sentencias de SQL dinámico ejecutadas en todas las particiones de base de datos, ordenadas por el número de filas leídas.

```
SELECT PREP_TIME_WORST, NUM_COMPILATIONS, SUBSTR(STMT_TEXT, 1, 60)
AS STMT_TEXT, DBPARTITIONNUM
FROM SYSIBMADM.SNAPDYN_SQL ORDER BY ROWS_READ
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| PREP_TIME_WORST | NUM_COMPILATIONS | ... |
|-----------------|------------------|-----|
| 98 | 1 | ... |
| 9 | 1 | ... |
| 0 | 0 | ... |
| 0 | 1 | ... |
| 0 | 1 | ... |
| 0 | 1 | ... |
| 0 | 1 | ... |
| 0 | 1 | ... |
| 0 | 1 | ... |
| 40 | 1 | ... |

9 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```
... STMT_TEXT
... -----
... select prep_time_worst, num_compilations, substr(stmt_text,
... select * from dbuser.employee
... SET CURRENT LOCALE LC_CTYPE = 'en_US'
... select prep_time_worst, num_compilations, substr(stmt_text,
... select prep_time_worst, num_compilations, substr(stmt_text,
... select * from dbuser.employee
... insert into dbuser.employee values(1)
... select * from dbuser.employee
... insert into dbuser.employee values(1)
```

Datos de salida de esta consulta (continuación).

```
... DBPARTITIONNUM
... -----
... 0
... 0
... 0
... 2
... 1
... 2
... 2
... 1
... 0
```

Función de tabla SNAP_GET_DYN_SQL_V95

La función de tabla SNAP_GET_DYN_SQL_V95 devuelve la misma información que la vista administrativa SNAPDYN_SQL, pero le permite recuperar la

información para una base de datos específica de una partición de base de datos específica, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

Esta función de tabla devuelve información equivalente a la que proporciona el mandato CLP GET SNAPSHOT FOR DYNAMIC SQL ON alias de base de datos.

Consulte la Tabla 147 en la página 587 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
▶▶ SNAP_GET_DYN_SQL_V95 (—nombrebd— [ , númparticiónbd ] )▶▶
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique NULL o una serie vacía para tomar la instantánea de la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_DYN_SQL_V95 tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_DYN_SQL_V95
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON

- SYSCTRL
- SYSMANT
- SYSADM

Ejemplo

Recuperar una lista de sentencias de SQL dinámico ejecutadas en la partición de base de datos correspondiente a la base de datos conectada actualmente, ordenadas por el número de filas leídas.

```
SELECT PREP_TIME_WORST, NUM_COMPILATIONS, SUBSTR(STMT_TEXT, 1, 60)
AS STMT_TEXT FROM TABLE(SNAP_GET_DYN_SQL_V95('','-1)) as T
ORDER BY ROWS_READ
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
PREP_TIME_WORST      ...
-----
0 ...
3 ...
...
4 ...
...
4 ...
...
4 ...
...
3 ...
...
4 ...
...
...
```

Datos de salida de esta consulta (continuación).

```
... NUM_COMPILATIONS  STMT_TEXT
... -----
... 0 SET CURRENT LOCALE LC_CTYPE = 'en_US'
... 1 select rows_read, rows_written,
...      substr(stmt_text, 1, 40) as
... 1 select * from table
...      (snap_get_dyn_sqlv9('','-1)) as t
... 1 select * from table
...      (snap_getdetaillog9('','-1)) as t
... 1 select * from table
...      (snap_get_hadr('','-1)) as t
... 1 select prep_time_worst, num_compilations,
...      substr(stmt_text,
... 1 select prep_time_worst, num_compilations,
...      substr(stmt_text,
```

Después de ejecutar una carga de trabajo, el usuario puede utilizar la siguiente consulta con la función de tabla.

```
SELECT STATS_FABRICATE_TIME, SYNC_RUNSTATS_TIME
FROM TABLE (SNAP_GET_DYN_SQL_V95('mytestdb', -1))
AS SNAPDB
```

```
STATS_FABRICATE_TIME SYNC_RUNSTATS_TIME
-----
2 12
1 30
```

Para la vista basada en esta función de tabla:

```
SELECT STATS_FABRICATE_TIME,SYNC_RUNSTATS_TIME
FROM SYSIBMADM.SNAPDYN_SQL
```

```
STATS_FABRICATE_TIME SYNC_RUNSTATS_TIME
-----
                    5                10
                    3                20
```

2 registro(s) seleccionado(s).

Información devuelta

Tabla 177. Información devuelta por la vista administrativa SNAPDYN_SQL y la función de tabla SNAP_GET_DYN_SQL_V95

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| NUM_EXECUTIONS | BIGINT | num_executions - Ejecuciones de sentencia |
| NUM_COMPILATIONS | BIGINT | num_compilations - Compilaciones de sentencia |
| PREP_TIME_WORST | BIGINT | prep_time_worst - Peor tiempo de preparación de sentencia |
| PREP_TIME_BEST | BIGINT | prep_time_best - Mejor tiempo de preparación de sentencia |
| INT_ROWS_DELETED | BIGINT | int_rows_deleted - Filas internas suprimidas |
| INT_ROWS_INSERTED | BIGINT | int_rows_inserted - Filas internas insertadas |
| INT_ROWS_UPDATED | BIGINT | int_rows_updated - Filas internas actualizadas |
| ROWS_READ | BIGINT | rows_read - Filas leídas |
| ROWS_WRITTEN | BIGINT | rows_written - Filas grabadas |
| STMT_SORTS | BIGINT | stmt_sorts - Clasificaciones de sentencias |
| SORT_OVERFLOWS | BIGINT | sort_overflows - Desbordamientos de clasificación |
| TOTAL_SORT_TIME | BIGINT | total_sort_time - Tiempo de clasificación total |
| POOL_DATA_L_READS | BIGINT | pool_data_l_reads - Lecturas lógicas de datos de la agrupación de almacenamientos intermedios |
| POOL_DATA_P_READS | BIGINT | pool_data_p_reads - Lecturas físicas de datos de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_L_READS | BIGINT | pool_temp_data_l_reads - Lecturas lógicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_P_READS | BIGINT | pool_temp_data_p_reads - Lecturas físicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_INDEX_L_READS | BIGINT | pool_index_l_reads - Lecturas lógicas de índices de la agrupación de almacenamientos intermedios |
| POOL_INDEX_P_READS | BIGINT | pool_index_p_reads - Lecturas físicas de índices de la agrupación de almacenamientos intermedios |

Tabla 177. Información devuelta por la vista administrativa SNAPDYN_SQL y la función de tabla SNAP_GET_DYN_SQL_V95 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|--|
| POOL_TEMP_INDEX_L_READS | BIGINT | pool_temp_index_l_reads - Lecturas lógicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_P_READS | BIGINT | pool_temp_index_p_reads - Lecturas físicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_XDA_L_READS | BIGINT | pool_xda_l_reads - Lecturas lógicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_XDA_P_READS | BIGINT | pool_xda_p_reads - Lecturas físicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_L_READS | BIGINT | pool_temp_xda_l_reads - Lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_P_READS | BIGINT | pool_temp_xda_p_reads - Elemento del supervisor de las lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |
| TOTAL_EXEC_TIME | BIGINT | total_exec_time - Tiempo transcurrido de ejecución de sentencia |
| TOTAL_EXEC_TIME_MS | BIGINT | total_exec_time - Tiempo transcurrido de ejecución de sentencia |
| TOTAL_USR_CPU_TIME | BIGINT | total_usr_cpu_time - Total de CPU de usuario para una sentencia |
| TOTAL_USR_CPU_TIME_MS | BIGINT | total_usr_cpu_time - Total de CPU de usuario para una sentencia |
| TOTAL_SYS_CPU_TIME | BIGINT | total_sys_cpu_time - Total de CPU de sistema para una sentencia |
| TOTAL_SYS_CPU_TIME_MS | BIGINT | total_sys_cpu_time - Total de CPU de sistema para una sentencia |
| STMT_TEXT | CLOB(2 M) | stmt_text - Texto de sentencia de SQL |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |
| STATS_FABRICATE_TIME | BIGINT | Tiempo total (en milisegundos) que ha empleado el sistema en crear las estadísticas necesarias sin exploración de tabla o de índice durante la compilación de consultas para una sentencia dinámica. |
| SYNC_RUNSTATS_TIME | BIGINT | Tiempo total (en milisegundos) empleado en actividades síncronas de recopilación de estadísticas durante la compilación de consultas para una sentencia dinámica. |

Vista administrativa SNAPFCM y función de tabla SNAP_GET_FCM – Recuperar la información de instantáneas del grupo de datos lógicos fcm

La vista administrativa SNAPFCM y la función de tabla SNAP_GET_FCM devuelven información acerca del gestor de comunicaciones rápidas (FCM) a partir de la instantánea de una base de datos, en concreto el grupo de datos lógicos fcm.

SNAPFCM, vista administrativa

La vista administrativa SNAPFCM, utilizada con las vistas administrativas SNAPDBM, SNAPDBM_MEMORY_POOL, SNAPFCM_PART y SNAPSWITCHES, ofrece los datos equivalentes al mandato GET SNAPSHOT FOR DBM.

El esquema es SYSIBMADM.

Consulte la Tabla 148 en la página 591 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPFCM
- Privilegio CONTROL en la vista administrativa SNAPFCM
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE para la función de tabla SNAP_GET_FCM
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar información sobre los almacenamientos intermedios de mensajes de Fast Communication Manager en todas las particiones de base de datos.

```
SELECT BUFF_FREE, BUFF_FREE_BOTTOM, DBPARTITIONNUM  
FROM SYSIBMADM.SNAPFCM ORDER BY DBPARTITIONNUM
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| BUFF_FREE | BUFF_FREE_BOTTOM | DBPARTITIONNUM |
|-----------|------------------|----------------|
| 5120 | 5100 | 0 |
| 5120 | 5100 | 1 |
| 5120 | 5100 | 2 |

SNAP_GET_FCM, función de tabla

La función de tabla SNAP_GET_FCM devuelve la misma información que la vista administrativa SNAPFCM, pero le permite recuperar la información para una partición de base de datos específica, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_FCM, utilizada con las funciones de tabla SNAP_GET_DBM_V95, SNAP_GET_DBM_MEMORY_POOL, SNAP_GET_FCM_PART y SNAP_GET_SWITCHES, ofrece los datos equivalentes al mandato GET SNAPSHOT FOR DBM.

Consulte la Tabla 148 en la página 591 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
▶▶ SNAP_GET_FCM ( [ núm-partición-bd ] )
```

El esquema es SYSPROC.

Parámetro de la función de tabla

núm-partición-bd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si no se utiliza esta opción de entrada, se devolverán datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *núm-partición-bd* se establece en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_FCM toma una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE para la función de tabla SNAP_GET_FCM
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar información sobre los almacenamientos intermedios de mensajes de Fast Communication Manager en la partición 1 de la base de datos.

```
SELECT BUFF_FREE, BUFF_FREE_BOTTOM, DBPARTITIONNUM  
FROM TABLE(SYSPROC.SNAP_GET_FCM( 1 )) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
BUFF_FREE          BUFF_FREE_BOTTOM    DBPARTITIONNUM  
-----  
          5120                5100                1
```

Información devuelta

Tabla 178. Información que devuelve la vista administrativa SNAPFCM y la función de tabla SNAP_GET_FCM

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| BUFF_FREE | BIGINT | buff_free - Almacenamientos intermedios FCM libres actualmente |
| BUFF_FREE_BOTTOM | BIGINT | buff_free_bottom - Almacenamientos intermedios FCM mínimos libres |
| CH_FREE | BIGINT | ch_free - Canales libres actualmente |
| CH_FREE_BOTTOM | BIGINT | ch_free_bottom - Número mínimo de canales libres |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa SNAPFCM_PART y función de tabla SNAP_GET_FCM_PART – Recuperar la información de instantáneas del grupo de datos lógicos fcm_node

La vista administrativa SNAPFCM_PART y la función de tabla SNAP_GET_FCM_PART devuelven información acerca del gestor de comunicaciones rápidas (FCM) a partir de la instantánea del gestor de bases de datos, en concreto el grupo de datos lógicos fcm_node.

SNAPFCM_PART, vista administrativa

La vista administrativa SNAPFCM_PART, utilizada con las vistas administrativas SNAPDBM, SNAPDBM_MEMORY_POOL, SNAPFCM y SNAPSWITCHES, ofrece los datos equivalentes al mandato GET SNAPSHOT FOR DBM.

El esquema es SYSIBMADM.

Consulte la Tabla 149 en la página 594 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPFCM_PART
- Privilegio CONTROL en la vista administrativa SNAPFCM_PART
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE en la función de tabla SNAP_GET_FCM_PART
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar información enviada y recibida de almacenamientos intermedios para el Fast Communication Manager.

```
SELECT CONNECTION_STATUS, TOTAL_BUFFERS_SENT, TOTAL_BUFFERS_RECEIVED
FROM SYSIBMADM.SNAPFCM_PART WHERE DBPARTITIONNUM = 0
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| CONNECTION_STATUS | TOTAL_BUFFERS_SENT | TOTAL_BUFFERS_RCVD |
|-------------------|--------------------|--------------------|
| INACTIVE | 2 | 1 |

1 registro(s) seleccionado(s).

SNAP_GET_FCM_PART, función de tabla

La función de tabla SNAP_GET_FCM_PART devuelve la misma información que la vista administrativa SNAPFCM_PART, pero le permite recuperar la información para una partición de base de datos específica, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_FCM_PART, utilizada con las funciones de tabla SNAP_GET_DBM_V95, SNAP_GET_DBM_MEMORY_POOL, SNAP_GET_FCM y SNAP_GET_SWITCHES, ofrece los datos equivalentes al mandato GET SNAPSHOT FOR DBM.

Consulte la Tabla 149 en la página 594 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
▶▶ SNAP_GET_FCM_PART ( [núm-partición-bd] ) ▶▶▶▶
```

El esquema es SYSPROC.

Parámetro de la función de tabla

núm_partición_bd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición actual o -2 para un agregado de todas las particiones de base de datos activas. Si no se utiliza esta opción de entrada, se devolverán datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *núm_partición_bd* se establece en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_FCM_PART toma una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE en la función de tabla SNAP_GET_FCM_PART
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMANT
- SYSADM

Ejemplo

Recuperar información enviada y recibida de almacenamientos intermedios para el Fast Communication Manager en todas las particiones de base de datos.

```
SELECT FCM_DBPARTITIONNUM, TOTAL_BUFFERS_SENT, TOTAL_BUFFERS_RCVD,  
       DBPARTITIONNUM FROM TABLE(SNAP_GET_FCM_PART()) AS T  
ORDER BY DBPARTITIONNUM
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| FCM_DBPARTITIONNUM | TOTAL_BUFFERS_SENT | TOTAL_BUFFERS_RCVD | DBPARTITIONNUM |
|--------------------|--------------------|--------------------|----------------|
| 0 | 305 | 305 | 0 |
| 1 | 5647 | 1664 | 0 |
| 2 | 5661 | 1688 | 0 |
| 0 | 19 | 19 | 1 |
| 1 | 305 | 301 | 1 |
| 2 | 1688 | 5661 | 1 |
| 0 | 1664 | 5647 | 2 |
| 1 | 10 | 10 | 2 |
| 2 | 301 | 305 | 2 |

Información devuelta

Tabla 179. Información que devuelven la vista administrativa SNAPFCM_PART y la función de tabla SNAP_GET_FCM_PART

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| CONNECTION_STATUS | VARCHAR(10) | connection_status - Estado de conexión. Esta interfaz devuelve un identificador de texto basándose en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none">• INACTIVE• ACTIVE• CONGESTED |
| TOTAL_BUFFERS_SENT | BIGINT | total_buffers_sent - Total de almacenamientos intermedios FCM enviados |
| TOTAL_BUFFERS_RCVD | BIGINT | total_buffers_rcvd - Total de almacenamientos intermedios FCM recibidos |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |
| FCM_DBPARTITIONNUM | SMALLINT | El número de la partición de base de datos a la que se han enviado o desde la que se han recibido los datos (según las columnas TOTAL_BUFFERS_SENT y TOTAL_BUFFERS_RCVD). |

Vista administrativa SNAPHADR y función de tabla SNAP_GET_HADR – Recuperar información de instantáneas del grupo de datos lógicos hadr

La vista administrativa SNAPHADR y la función de tabla SNAP_GET_HADR devuelven información sobre la recuperación de catástrofes de alta disponibilidad a partir de la instantánea de una base de datos, en concreto el grupo de datos lógicos hadr.

SNAPHADR, vista administrativa

Esta vista administrativa permite recuperar la información de instantáneas del grupo de datos lógicos hadr para la base de datos conectada actualmente. Esta vista sólo devuelve los datos si la base de datos es principal o de recuperación de catástrofes de alta disponibilidad (HADR) en espera.

La vista administrativa SNAPHADR, utilizada con las vistas administrativas SNAPDB, SNAPDB_MEMORY_POOL, SNAPDETAILLOG y SNAPSTORAGE_PATHS, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR DATABASE ON alias de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 150 en la página 597 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPHADR
- Privilegio CONTROL en la vista administrativa SNAPHADR
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_HADR
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMMAINT
- SYSADM

Ejemplo

Recuperar la información sobre el estado y la configuración de HADR en la base de datos HADR principal.

```
SELECT SUBSTR(DB_NAME, 1, 8) AS DBNAME, HADR_ROLE, HADR_STATE,  
       HADR_SYNCMODE, HADR_CONNECT_STATUS  
FROM SYSIBMADM.SNAPHADR
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| DBNAME | HADR_ROLE | HADR_STATE | HADR_SYNCMODE | HADR_CONNECT_STATUS |
|--------|-----------|------------|---------------|---------------------|
| SAMPLE | PRIMARY | PEER | SYNC | CONNECTED |

1 registro(s) seleccionado(s).

SNAP_GET_HADR, función de tabla

La función de tabla SNAP_GET_HADR devuelve la misma información que la vista administrativa SNAPHADR, pero permite recuperar la información para una base de datos concreta en una partición específica de la base de datos, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_HADR, utilizada con las funciones de tabla SNAP_GET_DB_V95, SNAP_GET_DB_MEMORY_POOL, SNAP_GET_DETAILLOG_V91 y SNAP_GET_STORAGE_PATHS, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR ALL DATABASES.

Consulte la Tabla 150 en la página 597 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
▶▶ SNAP_GET_HADR ( [ nombrebd ] [ , númparticiónbd ] ) ▶▶
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique una serie vacía para tomar la instantánea desde la base de datos conectada actualmente. Especifique un valor NULL para tomar la instantánea desde todas las bases de datos que se encuentran en la misma instancia que la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_HADR toma una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_HADR
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar la información sobre el estado y la configuración de HADR en todas las bases de datos.

```
SELECT SUBSTR(DB_NAME, 1, 8) AS DBNAME, HADR_ROLE, HADR_STATE,
       HADR_SYNCMODE, HADR_CONNECT_STATUS
FROM TABLE (SNAP_GET_HADR (CAST (NULL as VARCHAR(128)), 0)) as T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| DBNAME | HADR_ROLE | HADR_STATE | HADR_SYNCMODE | HADR_CONNECT_STATUS |
|--------|-----------|--------------|---------------|---------------------|
| SAMPLE | PRIMARY | PEER | SYNC | CONNECTED |
| TESTDB | PRIMARY | DISCONNECTED | NEARSYNC | DISCONNECTED |

2 registro(s) seleccionado(s).

Información devuelta

Tabla 180. Información devuelta por la vista administrativa SNAPHADR y la función de tabla SNAP_GET_HADR

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| HADR_ROLE | VARCHAR(10) | hadr_role - Rol HADR. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> PRIMARY STANDARD STANDBY |
| HADR_STATE | VARCHAR(14) | hadr_state - Estado HADR. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> DISCONNECTED LOCAL_CATCHUP PEER REM_CATCH_PEN REM_CATCHUP |
| HADR_SYNCMODE | VARCHAR(10) | hadr_syncmode - Modalidad de sincronización HADR. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> ASYNC NEARSYNC SYNC |

Tabla 180. Información devuelta por la vista administrativa SNAPHADR y la función de tabla SNAP_GET_HADR (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------|---------------|--|
| HADR_CONNECT_STATUS | VARCHAR(12) | hadr_connect_status - Estado de conexión HADR. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • CONGESTED • CONNECTED • DISCONNECTED |
| HADR_CONNECT_TIME | TIMESTAMP | hadr_connect_time - Hora de conexión de HADR |
| HADR_HEARTBEAT | INTEGER | hadr_heartbeat - Latido de HADR |
| HADR_LOCAL_HOST | VARCHAR(255) | hadr_local_host - Sistema principal local de HADR |
| HADR_LOCAL_SERVICE | VARCHAR(40) | hadr_local_service - Servicio local de HADR |
| HADR_REMOTE_HOST | VARCHAR(255) | hadr_remote_host - Sistema principal remoto de HADR |
| HADR_REMOTE_SERVICE | VARCHAR(40) | hadr_remote_service - Servicio remoto de HADR |
| HADR_REMOTE_INSTANCE | VARCHAR(128) | hadr_remote_instance - Instancia remota de HADR |
| HADR_TIMEOUT | BIGINT | hadr_timeout - Tiempo de espera de HADR |
| HADR_PRIMARY_LOG_FILE | VARCHAR(255) | hadr_primary_log_file - Archivo de anotaciones cronológicas primario de HADR |
| HADR_PRIMARY_LOG_PAGE | BIGINT | hadr_primary_log_page - Página de anotaciones cronológicas primarias de HADR |
| HADR_PRIMARY_LOG_LSN | BIGINT | hadr_primary_log_lsn - LSN de anotaciones cronológicas primarias de HADR |
| HADR_STANDBY_LOG_FILE | VARCHAR(255) | hadr_standby_log_file - Archivo de anotaciones cronológica en espera de HADR |
| HADR_STANDBY_LOG_PAGE | BIGINT | hadr_standby_log_page - Página de anotaciones cronológicas en espera de HADR |
| HADR_STANDBY_LOG_LSN | BIGINT | hadr_standby_log_lsn - LSN de anotaciones cronológicas en espera de HADR |
| HADR_LOG_GAP | BIGINT | hadr_log_gap - Distancia de anotación cronológica de HADR |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa SNAPLOCK y función de tabla SNAP_GET_LOCK – Recuperar información de instantáneas del grupo de datos lógicos lock

La vista administrativa SNAPLOCK y la función de tabla SNAP_GET_LOCK devuelven información de instantáneas acerca de los bloqueos, en concreto el grupo de datos lógicos lock.

SNAPLOCK, vista administrativa

Esta vista administrativa permite recuperar la información de instantáneas del grupo de datos lógicos lock para la base de datos conectada actualmente.

La vista administrativa SNAPLOCK, utilizada con la vista administrativa SNAPLOCKWAIT, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR LOCKS ON alias de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 151 en la página 601 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPLOCK
- Privilegio CONTROL en la vista administrativa SNAPLOCK
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_LOCK
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar información de bloqueo de la partición 0 de la base de datos conectada actualmente.

```
SELECT AGENT_ID, LOCK_OBJECT_TYPE, LOCK_MODE, LOCK_STATUS
FROM SYSIBMADM.SNAPLOCK WHERE DBPARTITIONNUM = 0
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| AGENT_ID | LOCK_OBJECT_TYPE | LOCK_MODE | LOCK_STATUS |
|----------|------------------|-----------|-------------|
| 7 | TABLE | IX | GRNT |

1 registro(s) seleccionado(s).

SNAP_GET_LOCK, función de tabla

La función de tabla SNAP_GET_LOCK devuelve la misma información que la vista administrativa SNAPLOCK, pero permite recuperar la información para una base de datos concreta en una partición específica de la base de datos, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_LOCK, utilizada con la función de tabla SNAP_GET_LOCKWAIT, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR LOCKS ON alias de base de datos.

Consulte la Tabla 151 en la página 601 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
▶▶ SNAP_GET_LOCK ( ( nombrebd [ , númparticiónbd ] ) ) ▶▶
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique un valor nulo o una serie vacía para tomar la instantánea desde la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_LOCK toma una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_LOCK
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMANT
- SYSADM

Ejemplo

Recuperar información de bloqueo de la partición actual de la base de datos conectada actualmente.

```
SELECT AGENT_ID, LOCK_OBJECT_TYPE, LOCK_MODE, LOCK_STATUS
FROM TABLE(SNAP_GET_LOCK('1',-1)) as T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
AGENT_ID      LOCK_OBJECT_TYPE  LOCK_MODE  LOCK_STATUS
-----
          680 INTERNALV_LOCK      S          GRNT
          680 INTERNALP_LOCK      S          GRNT
```

2 registro(s) seleccionado(s).

Información devuelta

Tabla 181. Información que devuelve la vista administrativa SNAPLOCK y la función de tabla SNAP_GET_LOCK

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| AGENT_ID | BIGINT | agent_id - Descriptor de contexto de aplicación (ID de agente) |
| TAB_FILE_ID | BIGINT | table_file_id - Identificación de archivo de tabla |

Tabla 181. Información que devuelve la vista administrativa SNAPLOCK y la función de tabla SNAP_GET_LOCK (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|--|
| LOCK_OBJECT_TYPE | VARCHAR(18) | <p>lock_object_type - Tipo de objeto de bloqueo esperado. Esta interfaz devuelve un identificador de texto basándose en las sentencias define de sqlmon.h y es uno de los siguientes:</p> <ul style="list-style-type: none"> • AUTORESIZE_LOCK • AUTOSTORAGE_LOCK • BLOCK_LOCK • EOT_LOCK • INPLACE_REORG_LOCK • INTERNAL_LOCK • INTERNALB_LOCK • INTERNALC_LOCK • INTERNALJ_LOCK • INTERNALL_LOCK • INTERNALO_LOCK • INTERNALQ_LOCK • INTERNALP_LOCK • INTERNALS_LOCK • INTERNALT_LOCK • INTERNALV_LOCK • KEYVALUE_LOCK • ROW_LOCK • SYSBOOT_LOCK • TABLE_LOCK • TABLE_PART_LOCK • TABLESPACE_LOCK • XML_PATH_LOCK |
| LOCK_MODE | VARCHAR(10) | <p>lock_mode - Modalidad de bloqueo. Esta interfaz devuelve un identificador de texto basándose en las sentencias define de sqlmon.h y es uno de los siguientes:</p> <ul style="list-style-type: none"> • IN • IS • IX • NON (si no existe bloqueo) • NS • NW • S • SIX • U • X • Z |

Tabla 181. Información que devuelve la vista administrativa SNAPLOCK y la función de tabla SNAP_GET_LOCK (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|---|
| LOCK_STATUS | VARCHAR(10) | lock_status - Estado de bloqueo. Esta interfaz devuelve un identificador de texto basándose en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • CONV • GRNT |
| LOCK_ESCALATION | SMALLINT | lock_escalation - Escalamiento de bloqueos |
| TABNAME | VARCHAR(128) | table_name - Nombre de tabla |
| TABSCHEMA | VARCHAR(128) | table_schema - Nombre de esquema de tabla |
| TBSP_NAME | VARCHAR(128) | tablespace_name - Nombre de espacio de tablas |
| LOCK_ATTRIBUTES | VARCHAR(128) | lock_attributes - Atributos de bloqueo. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h. Si no hay bloqueos, el identificador de texto es NONE; de lo contrario, es cualquier combinación de los elementos siguientes separados por un signo '+': <ul style="list-style-type: none"> • ALLOW_NEW • DELETE_IN_BLOCK • ESCALATED • INSERT • NEW_REQUEST • RR • RR_IN_BLOCK • UPDATE_DELETE • WAIT_FOR_AVAIL |
| LOCK_COUNT | BIGINT | lock_count - Número de bloqueos |

Tabla 181. Información que devuelve la vista administrativa SNAPLOCK y la función de tabla SNAP_GET_LOCK (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| LOCK_CURRENT_MODE | VARCHAR(10) | lock_current_mode - Modalidad de bloqueo original antes de la conversión. Esta interfaz devuelve un identificador de texto basándose en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • IN • IS • IX • NON (si no existe bloqueo) • NS • NW • S • SIX • U • X • Z |
| LOCK_HOLD_COUNT | BIGINT | lock_hold_count - Número de bloqueos retenidos |
| LOCK_NAME | VARCHAR(32) | lock_name - Nombre de bloqueo |
| LOCK_RELEASE_FLAGS | BIGINT | lock_release_flags - Distintivos de liberación de bloqueo |
| DATA_PARTITION_ID | INTEGER | data_partition_id - Identificador de la partición de datos. Para una tabla no particionada, este elemento será NULL. |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa SNAPLOCKWAIT y función de tabla SNAP_GET_LOCKWAIT – Recuperar información de instantáneas del grupo de datos lógicos lockwait

La vista administrativa SNAPLOCKWAIT y la función de tabla SNAP_GET_LOCKWAIT devuelven información de instantáneas acerca de las esperas de bloqueo, en concreto el grupo de datos lógicos lockwait.

SNAPLOCKWAIT, vista administrativa

Esta vista administrativa permite recuperar la información de instantáneas del grupo de datos lógicos lockwait para la base de datos conectada actualmente.

La vista administrativa SNAPLOCKWAIT, utilizada con la vista administrativa SNAPLOCK, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR LOCKS ON alias de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 152 en la página 607 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPLOCKWAIT
- Privilegio CONTROL en la vista administrativa SNAPLOCKWAIT
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_LOCKWAIT
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMAINT
- SYSADM

Ejemplo

Recuperar información acerca de las esperas de bloqueo en la partición de base de datos 0 para la base de datos conectada actualmente.

```
SELECT AGENT_ID, LOCK_MODE, LOCK_OBJECT_TYPE, AGENT_ID_HOLDING_LK,
       LOCK_MODE_REQUESTED FROM SYSIBMADM.SNAPLOCKWAIT
WHERE DBPARTITIONNUM = 0
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
AGENT_ID    LOCK_MODE LOCK_OBJECT_TYPE ...
-----
          7 IX          TABLE          ...
```

1 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```
... AGENT_ID_HOLDING_LK LOCK_MODE_REQUESTED
... -----
...                   12 IS
```

SNAP_GET_LOCKWAIT, función de tabla

La función de tabla SNAP_GET_LOCKWAIT devuelve la misma información que la vista administrativa SNAPLOCKWAIT, pero permite recuperar la información para una base de datos concreta en una partición específica de la base de datos, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_LOCKWAIT, utilizada con la función de tabla SNAP_GET_LOCK, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR LOCKS ON alias de base de datos.

Consulte la Tabla 152 en la página 607 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
▶▶ SNAP_GET_LOCKWAIT ( nombrebd [ , númparticiónbd ] ) ▶▶
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique un valor nulo o una serie vacía para tomar la instantánea desde la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_LOCKWAIT tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_LOCKWAIT
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT

- SYSADM

Ejemplo

Recuperar información acerca de las esperas de bloqueo en la partición actual de la base de datos conectada actualmente.

```
SELECT AGENT_ID, LOCK_MODE, LOCK_OBJECT_TYPE, AGENT_ID_HOLDING_LK,
       LOCK_MODE_REQUESTED FROM TABLE(SNAP_GET_LOCKWAIT('',-1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
AGENT_ID      LOCK_MODE  LOCK_OBJECT_TYPE  ...
-----
          12 X          ROW_LOCK          ...
```

1 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```
... AGENT_ID_HOLDING_LK  LOCK_MODE_REQUESTED
... -----
...                      7 X
```

Nota sobre uso

Para ver información de esperas de bloqueo, primero deberá activar el conmutador de supervisor LOCK por omisión en la configuración del gestor de bases de datos. Para que el cambio tenga efecto inmediatamente, conéctese explícitamente a la instancia utilizando CLP y luego vuelva a emitir el mandato CLP:

```
UPDATE DATABASE MANAGER CONFIGURATION CLP USING DFT_MON_LOCK ON
```

Los valores por omisión también se puede activar mediante el procedimiento almacenado ADMIN_CMD. Por ejemplo:

```
CALL SYSPROC.ADMIN_CMD('update dbm cfg using DFT_MON_LOCK ON')
```

Si se utiliza el procedimiento almacenado ADMIN_CMD o si se utiliza el mandato clp sin conectarse previamente a una instancia, la instancia deberá reciclarse antes de que el cambio tiene efecto.

Información devuelta

Tabla 182. Información que devuelve la vista administrativa SNAPLOCKWAIT y la función de tabla SNAP_GET_LOCKWAIT

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| AGENT_ID | BIGINT | agent_id - Descriptor de contexto de aplicación (ID de agente) |
| SUBSECTION_NUMBER | BIGINT | ss_number - Número de subsección |

Tabla 182. Información que devuelve la vista administrativa SNAPLOCKWAIT y la función de tabla SNAP_GET_LOCKWAIT (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|---|
| LOCK_MODE | VARCHAR(10) | lock_mode - Modalidad de bloqueo. Esta interfaz devuelve un identificador de texto basándose en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • IN • IS • IX • NON (si no existe bloqueo) • NS • NW • S • SIX • U • X • Z |
| LOCK_OBJECT_TYPE | VARCHAR(18) | lock_object_type - Tipo de objeto de bloqueo esperado. Esta interfaz devuelve un identificador de texto basándose en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • AUTORESIZE_LOCK • AUTOSTORAGE_LOCK • BLOCK_LOCK • EOT_LOCK • INPLACE_REORG_LOCK • INTERNAL_LOCK • INTERNALB_LOCK • INTERNALC_LOCK • INTERNALJ_LOCK • INTERNALL_LOCK • INTERNALO_LOCK • INTERNALQ_LOCK • INTERNALP_LOCK • INTERNALS_LOCK • INTERNALT_LOCK • INTERNALV_LOCK • KEYVALUE_LOCK • ROW_LOCK • SYSBOOT_LOCK • TABLE_LOCK • TABLE_PART_LOCK • TABLESPACE_LOCK • XML_PATH_LOCK |

Tabla 182. Información que devuelve la vista administrativa SNAPLOCKWAIT y la función de tabla SNAP_GET_LOCKWAIT (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|----------------------|---------------|--|
| AGENT_ID_HOLDING_LK | BIGINT | agent_id_holding_lock - ID de agente que mantiene el bloqueo |
| LOCK_WAIT_START_TIME | TIMESTAMP | lock_wait_start_time - Indicación de fecha y hora de inicio de espera de bloqueo |
| LOCK_MODE_REQUESTED | VARCHAR(10) | lock_mode_requested - Modalidad de bloqueo solicitada. Esta interfaz devuelve un identificador de texto basándose en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • IN • IS • IX • NON (si no existe bloqueo) • NS • NW • S • SIX • U • X • Z |
| LOCK_ESCALATION | SMALLINT | lock_escalation - Escalamiento de bloqueos |
| TABNAME | VARCHAR(128) | table_name - Nombre de tabla |
| TABSHEMA | VARCHAR(128) | table_schema - Nombre de esquema de tabla |
| TBSP_NAME | VARCHAR(128) | tablespace_name - Nombre de espacio de tablas |
| APPL_ID_HOLDING_LK | VARCHAR(128) | appl_id_holding_lk - ID de aplicación que mantiene el bloqueo |

Tabla 182. Información que devuelve la vista administrativa SNAPLOCKWAIT y la función de tabla SNAP_GET_LOCKWAIT (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| LOCK_ATTRIBUTES | VARCHAR(128) | lock_attributes - Atributos de bloqueo. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h. Si no hay bloqueos, el identificador de texto es NONE; de lo contrario, es cualquier combinación de los elementos siguientes separados por un signo '+': <ul style="list-style-type: none"> • ALLOW_NEW • DELETE_IN_BLOCK • ESCALATED • INSERT • NEW_REQUEST • RR • RR_IN_BLOCK • UPDATE_DELETE • WAIT_FOR_AVAIL |
| LOCK_CURRENT_MODE | VARCHAR(10) | lock_current_mode - Modalidad de bloqueo original antes de la conversión. Esta interfaz devuelve un identificador de texto basándose en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • IN • IS • IX • NON (si no existe bloqueo) • NS • NW • S • SIX • U • X • Z |
| LOCK_NAME | VARCHAR(32) | lock_name - Nombre de bloqueo |
| LOCK_RELEASE_FLAGS | BIGINT | lock_release_flags - Distintivos de liberación de bloqueo. |
| DATA_PARTITION_ID | INTEGER | data_partition_id - Identificador de la partición de datos. Para una tabla no particionada, este elemento será NULL. |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa SNAPSTMT y función de tabla SNAP_GET_STMT – Recuperar información de instantáneas de sentencias

La vista administrativa SNAPSTMT y la función de tabla SNAP_GET_STMT devuelven información sobre sentencias SQL o XQuery desde una instantánea de la aplicación.

SNAPSTMT, vista administrativa

Esta vista administrativa le permite recuperar información acerca de la instantánea de sentencia para la base de datos conectada actualmente.

La vista administrativa SNAPSTMT, utilizada con las vistas administrativas SNAPAGENT, SNAPAGENT_MEMORY_POOL, SNAPAPPL, SNAPAPPL_INFO y SNAPSUBSECTION, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR APPLICATIONS ON alias de base de datos, pero recupera datos de todas las particiones de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 153 en la página 613 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPSTMT
- Privilegio CONTROL en la vista administrativa SNAPSTMT
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_STMT
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMAINT
- SYSADM

Ejemplo

Recuperar filas leídas, grabadas y aquellas en que se han efectuado operaciones, para las sentencias ejecutadas en la base de datos de partición única conectada actualmente.

```
SELECT SUBSTR(STMT_TEXT,1,30) AS STMT_TEXT, ROWS_READ, ROWS_WRITTEN,  
       STMT_OPERATION FROM SYSIBMADM.SNAPSTMT
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| STMT_TEXT | ROWS_READ | ROWS_WRITTEN | STMT_OPERATION |
|-----------|-----------|--------------|-----------------|
| - | | 0 | 0 FETCH |
| - | | 0 | 0 STATIC_COMMIT |

2 registro(s) seleccionado(s).

SNAP_GET_STMT, función de tabla

La función de tabla SNAP_GET_STMT devuelve la misma información que la vista administrativa SNAP_STMT, pero permite recuperar la información para una base de datos concreta en una partición específica de la base de datos, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_STMT, utilizada con las funciones de tablas SNAP_GET_AGENT, SNAP_GET_AGENT_MEMORY_POOL, SNAP_GET_APPL_V95, SNAP_GET_APPL_INFO_V95 y SNAP_GET_SUBSECTION, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR ALL APPLICATIONS, pero recupera datos de todas las particiones de base de datos.

Consulte la Tabla 153 en la página 613 para ver una lista completa de la información que se puede devolver.

Sintaxis

```

▶▶ SNAP_GET_STMT ( ( nombrebd [ , númparticiónbd ] ) )

```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique una serie vacía para tomar la instantánea desde la base de datos conectada actualmente. Especifique un valor NULL para tomar la instantánea desde todas las bases de datos que se encuentran en la misma instancia que la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es

posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_STMT toma una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_STMT
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar filas leídas, grabadas y aquellas en que se han efectuado operaciones, para las sentencias ejecutadas en la partición actual de la base de datos conectada actualmente.

```
SELECT SUBSTR(STMT_TEXT,1,30) AS STMT_TEXT, ROWS_READ,
       ROWS_WRITTEN, STMT_OPERATION FROM TABLE(SNAP_GET_STMT(' ', -1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
STMT_TEXT                ROWS_READ    ...
-----
update t set a=3         0 ...
SELECT SUBSTR(STMT_TEXT,1,30) 0 ...
-                        0 ...
-                        0 ...
update t set a=2        9 ...
...
5 registro(s) seleccionado(s).    ...
```

Datos de salida de esta consulta (continuación).

```
... ROWS_WRITTEN    STMT_OPERATION
... -----
...                0 EXECUTE_IMMEDIATE
...                0 FETCH
...                0 NONE
...                0 NONE
...                1 EXECUTE_IMMEDIATE
...
...
```

Información devuelta

Tabla 183. Información que devuelve la vista administrativa SNAPSTMT y la función de tabla SNAP_GET_STMT

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |

Tabla 183. Información que devuelve la vista administrativa SNAPSTMT y la función de tabla SNAP_GET_STMT (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|--|
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| AGENT_ID | BIGINT | agent_id - Descriptor de contexto de aplicación (ID de agente) |
| ROWS_READ | BIGINT | rows_read - Filas leídas |
| ROWS_WRITTEN | BIGINT | rows_written - Filas grabadas |
| NUM_AGENTS | BIGINT | num_agents - Número de agentes trabajando en una sentencia |
| AGENTS_TOP | BIGINT | agents_top - Número de agentes creados |
| STMT_TYPE | VARCHAR(20) | stmt_type - Tipo de sentencia. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • DYNAMIC • NON_STMT • STATIC • STMT_TYPE_UNKNOWN |

Tabla 183. Información que devuelve la vista administrativa SNAPSTMT y la función de tabla SNAP_GET_STMT (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------|---------------|---|
| STMT_OPERATION | VARCHAR(20) | stmt_operation/operation - Operación de sentencia. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • CALL • CLOSE • COMPILE • DESCRIBE • EXECUTE • EXECUTE_IMMEDIATE • FETCH • FREE_LOCATOR • GETAA • GETNEXTCHUNK • GETTA • NONE • OPEN • PREP_COMMIT • PREP_EXEC • PREP_OPEN • PREPARE • REBIND • REDIST • REORG • RUNSTATS • SELECT • SET • STATIC_COMMIT • STATIC_ROLLBACK |
| SECTION_NUMBER | BIGINT | section_number - Número de sección |
| QUERY_COST_ESTIMATE | BIGINT | query_cost_estimate - Estimación de coste de consulta |
| QUERY_CARD_ESTIMATE | BIGINT | query_card_estimate - Estimación de consulta de número de filas |
| DEGREE_PARALLELISM | BIGINT | degree_parallelism - Grado de paralelismo |
| STMT_SORTS | BIGINT | stmt_sorts - Clasificaciones de sentencias |
| TOTAL_SORT_TIME | BIGINT | total_sort_time - Tiempo de clasificación total |
| SORT_OVERFLOWS | BIGINT | sort_overflows - Desbordamientos de clasificación |

Tabla 183. Información que devuelve la vista administrativa SNAPSTMT y la función de tabla SNAP_GET_STMT (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|----------------------|---------------|---|
| INT_ROWS_DELETED | BIGINT | int_rows_deleted - Filas internas suprimidas |
| INT_ROWS_UPDATED | BIGINT | int_rows_updated - Filas internas actualizadas |
| INT_ROWS_INSERTED | BIGINT | int_rows_inserted - Filas internas insertadas |
| FETCH_COUNT | BIGINT | fetch_count - Número de captaciones satisfactorias |
| STMT_START | TIMESTAMP | stmt_start - Indicación de fecha y hora de inicio de operación de sentencia |
| STMT_STOP | TIMESTAMP | stmt_stop - Indicación de fecha y hora de detención de operación de sentencia |
| STMT_USR_CPU_TIME_S | BIGINT | stmt_usr_cpu_time - Tiempo de CPU de usuario utilizado por sentencia |
| STMT_USR_CPU_TIME_MS | BIGINT | stmt_usr_cpu_time - Tiempo de CPU de usuario utilizado por sentencia |
| STMT_SYS_CPU_TIME_S | BIGINT | stmt_sys_cpu_time - Tiempo de CPU de sistema utilizado por sentencia |
| STMT_SYS_CPU_TIME_MS | BIGINT | stmt_sys_cpu_time - Tiempo de CPU de sistema utilizado por sentencia |
| STMT_ELAPSED_TIME_S | BIGINT | stmt_elapsed_time - Tiempo transcurrido de sentencia más reciente |
| STMT_ELAPSED_TIME_MS | BIGINT | stmt_elapsed_time - Tiempo transcurrido de sentencia más reciente |
| BLOCKING_CURSOR | SMALLINT | blocking_cursor - Cursor de bloqueo |
| STMT_NODE_NUMBER | SMALLINT | stmt_node_number - Nodo de sentencia |
| CURSOR_NAME | VARCHAR(128) | cursor_name - Nombre de cursor |
| CREATOR | VARCHAR(128) | creator - Creador de aplicación |
| PACKAGE_NAME | VARCHAR(128) | package_name - Nombre de paquete |
| STMT_TEXT | CLOB(16 M) | stmt_text - Texto de sentencia de SQL |
| CONSISTENCY_TOKEN | VARCHAR(128) | consistency_token - Señal de coherencia del paquete |
| PACKAGE_VERSION_ID | VARCHAR(128) | package_version_id - Versión del paquete |

Tabla 183. Información que devuelve la vista administrativa SNAPSTMT y la función de tabla SNAP_GET_STMT (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|--|
| POOL_DATA_L_READS | BIGINT | pool_data_l_reads - Lecturas lógicas de datos de la agrupación de almacenamientos intermedios |
| POOL_DATA_P_READS | BIGINT | pool_data_p_reads - Lecturas físicas de datos de la agrupación de almacenamientos intermedios |
| POOL_INDEX_L_READS | BIGINT | pool_index_l_reads - Lecturas lógicas de índices de la agrupación de almacenamientos intermedios |
| POOL_INDEX_P_READS | BIGINT | pool_index_p_reads - Lecturas físicas de índices de la agrupación de almacenamientos intermedios |
| POOL_XDA_L_READS | BIGINT | pool_xda_l_reads - Elemento del supervisor de las lecturas lógicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_XDA_P_READS | BIGINT | pool_xda_p_reads - Elemento del supervisor de las lecturas físicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_L_READS | BIGINT | pool_temp_data_l_reads - Lecturas lógicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_P_READS | BIGINT | pool_temp_data_p_reads - Lecturas físicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_L_READS | BIGINT | pool_temp_index_l_reads - Lecturas lógicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_P_READS | BIGINT | pool_temp_index_p_reads - Lecturas físicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_L_READS | BIGINT | pool_temp_xda_l_reads - Lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_P_READS | BIGINT | pool_temp_xda_p_reads - Elemento del supervisor de las lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa **SNAPSTORAGE_PATHS** y función de tabla **SNAP_GET_STORAGE_PATHS** – Recuperar información de vía de acceso del almacenamiento automático

Nota: Esta función de tabla ha quedado en desuso y se ha sustituido por la “Vista administrativa **SNAPSTORAGE_PATHS** y función de tabla **SNAP_GET_STORAGE_PATHS_V97** - Recuperar información de vía de acceso de almacenamiento automático” en la página 618.

La vista administrativa **SNAPSTORAGE_PATHS** y la función de tabla **SNAP_GET_STORAGE_PATHS** devuelven una lista de vías de acceso del almacenamiento automático para la base de datos, incluida la información del sistema de archivos correspondiente a cada vía de acceso del almacenamiento, concretamente del grupo de datos lógicos `db_storage_group`.

SNAPSTORAGE_PATHS, vista administrativa

Esta vista administrativa permite recuperar la información de vía de acceso del almacenamiento automático para la base de datos conectada actualmente.

La vista administrativa **SNAPSTORAGE_PATHS**, utilizada con las vistas administrativas **SNAPDB**, **SNAPDETAILLOG**, **SNAPHADR** y **SNAPDB_MEMORY_POOL**, ofrece información equivalente al mandato `CLP GET SNAPSHOT FOR DATABASE ON` alias de base de datos.

El esquema es `SYSIBMADM`.

Consulte la Tabla 184 en la página 772 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio `SELECT` en la vista administrativa **SNAPSTORAGE_PATHS**
- Privilegio `CONTROL` en la vista administrativa **SNAPSTORAGE_PATHS**
- Autorización `DATAACCESS`

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio `EXECUTE` para la función de tabla **SNAP_GET_STORAGE_PATHS**
- Autorización `DATAACCESS`

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- `SYSMON`
- `SYSCTRL`
- `SYSMAINT`
- `SYSADM`

Ejemplo

Recuperar la vía de acceso de almacenamiento para la base de datos de una sola partición conectada actualmente.

```
SELECT SUBSTR(DB_NAME,1,8) AS DB_NAME, SUBSTR(DB_STORAGE_PATH,1,8)
      AS DB_STORAGE_PATH, SUBSTR(HOSTNAME,1,10) AS HOSTNAME
FROM SYSIBMADM.SNAPSTORAGE_PATHS
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
DB_NAME  DB_STORAGE_PATH  HOSTNAME
-----  -
STOPATH  d:                      JESSICAE
```

1 registro(s) seleccionado(s).

SNAP_GET_STORAGE_PATHS, función de tabla

La función de tabla SNAP_GET_STORAGE_PATHS devuelve la misma información que la vista administrativa SNAPSTORAGE_PATHS, pero le permite recuperar la información para una base de datos específica de una partición de base de datos específica, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_STORAGE_PATHS, utilizada con las funciones de tabla SNAP_GET_DB_V95, SNAP_GET_DETAILLOG_V91, SNAP_GET_HADR y SNAP_GET_DB_MEMORY_POOL, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR ALL DATABASES.

Consulte la Tabla 184 en la página 772 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
→ SNAP_GET_STORAGE_PATHS ( ( nombre_bd ) ) →
                        |
                        |, número_partición_bd
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombre_bd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique una serie vacía para tomar la instantánea desde la base de datos conectada actualmente. Especifique un valor NULL para tomar la instantánea desde todas las bases de datos que se encuentran en la misma instancia que la base de datos conectada actualmente.

número_partición_bd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombre_bd* no se establece en NULL y *número_partición_bd* se establece en NULL, -1 se establece implícitamente para *número_partición_bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre_bd*, se devuelven

datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_STORAGE_PATHS tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE para la función de tabla SNAP_GET_STORAGE_PATHS
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplos

Recuperar la información de vía de acceso de almacenamiento para todas las bases de datos activas.

```
SELECT SUBSTR(DB_NAME,1,8) AS DB_NAME, DB_STORAGE_PATH
FROM TABLE(SNAP_GET_STORAGE_PATHS(CAST (NULL AS VARCHAR(128)), -1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
DB_NAME  DB_STORAGE_PATH
-----  -
STOPATH  /home/jessicae/sdb
MYDB     /home/jessicae/mdb
```

2 registro(s) seleccionado(s)

Información devuelta

Para que se devuelva la información del sistema de archivos, debe activarse el conmutador de supervisor BUFFERPOOL.

Tabla 184. Información devuelta por la vista administrativa SNAPSTORAGE_PATHS y la función de tabla SNAP_GET_STORAGE_PATHS

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |

Tabla 184. Información devuelta por la vista administrativa SNAPSTORAGE_PATHS y la función de tabla SNAP_GET_STORAGE_PATHS (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| DB_STORAGE_PATH | VARCHAR(256) | db_storage_path - Vía de acceso de almacenamiento automático |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |
| FS_ID | VARCHAR(22) | fs_id - Número de identificación del sistema de archivos exclusivo |
| FS_TOTAL_SIZE | BIGINT | fs_total_size - Tamaño total de un sistema de archivos |
| FS_USED_SIZE | BIGINT | fs_used_size - Cantidad de espacio utilizada por un sistema de archivos |
| STO_PATH_FREE_SIZE | BIGINT | sto_path_free_sz - Espacio libre para la vía de acceso de almacenamiento automático |

Vista administrativa SNAPSUBSECTION y función de tabla SNAP_GET_SUBSECTION – Recuperar información de instantáneas del grupo de supervisor lógico de la subsección

La vista administrativa SNAPSUBSECTION y la función de tabla SNAP_GET_SUBSECTION devuelven información sobre las subsecciones de aplicaciones, concretamente la agrupación del supervisor lógico de la subsección.

SNAPSUBSECTION, vista administrativa

Esta vista administrativa permite recuperar la información de instantáneas del grupo de supervisor lógico de la subsección para la base de datos conectada actualmente.

La vista administrativa SNAPSUBSECTION, utilizada con las vistas administrativas SNAPAGENT, SNAPAGENT_MEMORY_POOL, SNAPAPPL, SNAPAPPL_INFO y SNAPSTMT, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR APPLICATIONS ON alias de base de datos, pero recupera datos de todas las particiones de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 155 en la página 623 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPSUBSECTION
- Privilegio CONTROL en la vista administrativa SNAPSUBSECTION
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_SUBSECTION
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Obtener el estado de las subsecciones que se ejecutan en todas las particiones de base de datos.

```
SELECT DB_NAME, STMT_TEXT, SS_STATUS, DBPARTITIONNUM
FROM SYSIBMADM.SNAPSUBSECTION
ORDER BY DB_NAME, SS_STATUS, DBPARTITIONNUM
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| DB_NAME | STMT_TEXT | SS_STATUS | DBPARTITIONNUM |
|---------|------------------------|-----------|----------------|
| SAMPLE | select * from EMPLOYEE | EXEC | 0 |
| SAMPLE | select * from EMPLOYEE | EXEC | 1 |

SNAP_GET_SUBSECTION, función de tabla

La función de tabla SNAP_GET_SUBSECTION devuelve la misma información que la vista administrativa SNAPSUBSECTION, pero permite recuperar la información de una base de datos concreta en una partición específica de la base de datos, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

Consulte la Tabla 155 en la página 623 para ver una lista completa de la información que se puede devolver.

La función de tabla SNAP_GET_SUBSECTION, utilizada con las funciones de tablas SNAP_GET_AGENT, SNAP_GET_AGENT_MEMORY_POOL, SNAP_GET_APPL_V95, SNAP_GET_APPL_INFO_V95 y SNAP_GET_STMT, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR ALL APPLICATIONS, pero recupera datos de todas las particiones de base de datos.

Sintaxis

```
▶▶ SNAP_GET_SUBSECTION ( ( nombrebd [ , númparticiónbd ] ) ) ▶▶
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada

actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique una serie vacía para tomar la instantánea desde la base de datos conectada actualmente. Especifique un valor NULL para tomar la instantánea desde todas las bases de datos que se encuentran en la misma instancia que la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_SUBSECTION tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_SUBSECTION
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCtrl
- SYSMAINT
- SYSADM

Ejemplo

Obtener el estado de las subsecciones que se ejecutan en todas las particiones de base de datos.

```
SELECT DB_NAME, STMT_TEXT, SS_STATUS, DBPARTITIONNUM
FROM TABLE(SYSPROC.SNAP_GET_SUBSECTION( '', 0 )) as T
ORDER BY DB_NAME, SS_STATUS, DBPARTITIONNUM
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| DB_NAME | STMT_TEXT | SS_STATUS | DBPARTITIONNUM |
|---------|------------------------|-----------|----------------|
| SAMPLE | select * from EMPLOYEE | EXEC | 0 |
| SAMPLE | select * from EMPLOYEE | EXEC | 1 |

Información devuelta

Tabla 185. Información que devuelve la vista administrativa SNAPSUBSECTION y la función de tabla SNAP_GET_SUBSECTION

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| STMT_TEXT | CLOB(16 M) | stmt_text - Texto de sentencia de SQL |
| SS_EXEC_TIME | BIGINT | ss_exec_time - Tiempo transcurrido de ejecución de subsección |
| TQ_TOT_SEND_SPILLS | BIGINT | tq_tot_send_spills - Número total de almacenamientos intermedios de colas de tabla desbordados |
| TQ_CUR_SEND_SPILLS | BIGINT | tq_cur_send_spills - Número actual de almacenamientos intermedios de colas de tabla desbordados |
| TQ_MAX_SEND_SPILLS | BIGINT | tq_max_send_spills - Número máximo de desbordamientos de almacenamientos intermedios de colas de tabla |
| TQ_ROWS_READ | BIGINT | tq_rows_read - Número de filas leídas de las colas de tabla |
| TQ_ROWS_WRITTEN | BIGINT | tq_rows_written - Número de filas grabadas en las colas de tabla |
| ROWS_READ | BIGINT | rows_read - Filas leídas |
| ROWS_WRITTEN | BIGINT | rows_written - Filas grabadas |
| SS_USR_CPU_TIME_S | BIGINT | ss_usr_cpu_time - Tiempo de CPU de usuario utilizado por subsección |
| SS_USR_CPU_TIME_MS | BIGINT | ss_usr_cpu_time - Tiempo de CPU de usuario utilizado por subsección |
| SS_SYS_CPU_TIME_S | BIGINT | ss_sys_cpu_time - Tiempo de CPU de sistema utilizado por subsección |
| SS_SYS_CPU_TIME_MS | BIGINT | ss_sys_cpu_time - Tiempo de CPU de sistema utilizado por subsección |
| SS_NUMBER | INTEGER | ss_number - Número de subsección |
| SS_STATUS | VARCHAR(20) | ss_status - Estado de subsección. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • EXEC • TQ_WAIT_TO_RCV • TQ_WAIT_TO_SEND • COMPLETED |

Tabla 185. Información que devuelve la vista administrativa SNAPSUBSECTION y la función de tabla SNAP_GET_SUBSECTION (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SS_NODE_NUMBER | SMALLINT | ss_node_number - Número de nodo de subsección |
| TQ_NODE_WAITED_FOR | SMALLINT | tq_node_waited_for - Nodo esperado en una cola de tabla |
| TQ_WAIT_FOR_ANY | INTEGER | tq_wait_for_any - En espera de cualquier nodo a enviar a una cola de tabla |
| TQ_ID_WAITING_ON | INTEGER | tq_id_waiting_on - Nodo en que se ha esperado en una cola de tabla |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa SNAPSWITCHES y función de tabla SNAP_GET_SWITCHES - Recuperar información de estado de conmutador de instantáneas de base de datos

La vista administrativa SNAPSWITCHES y la función de tabla SNAP_GET_SWITCHES devuelven información sobre el estado del conmutador de instantánea de base de datos.

SNAPSWITCHES, vista administrativa

Esta vista proporciona datos que son equivalentes a los que proporciona el mandato CLP GET DBM MONITOR SWITCHES.

El esquema es SYSIBMADM.

Consulte la Tabla 156 en la página 627 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPSWITCHES
- Privilegio CONTROL en la vista administrativa SNAPSWITCHES
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE en la función de tabla SNAP_GET_SWITCHES
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL

Si `núm_partición_bd` se establece en NULL, se intenta leer los datos del archivo creado por el procedimiento `SNAP_WRITE_FILE`. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla `SNAP_GET_SWITCHES` tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE en la función de tabla `SNAP_GET_SWITCHES`
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplos

Recuperar información acerca del estado de los conmutadores del supervisor DBM para la partición de base de datos actual.

```
SELECT UOW_SW_STATE, STATEMENT_SW_STATE, TABLE_SW_STATE,
       BUFFPOOL_SW_STATE, LOCK_SW_STATE, SORT_SW_STATE, TIMESTAMP_SW_STATE
FROM TABLE(SNAP_GET_SWITCHES(-1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
UOW_SW_STATE STATEMENT_SW_STATE TABLE_SW_STATE...
-----
1              1              1...
...
```

1 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```
... BUFFPOOL_SW_STATE LOCK_SW_STATE SORT_SW_STATE TIMESTAMP_SW_STATE
... -----
...              1              1              0              1
```

Información devuelta

Tabla 186. Información que devuelve la vista administrativa `SNAPSWITCHES` y la función de tabla `SNAP_GET_SWITCHES`

| Nombre de columna | Tipo de datos | Descripción |
|--------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| UOW_SW_STATE | SMALLINT | Estado del conmutador de registro del supervisor de unidad de trabajo (0 ó 1). |

Tabla 186. Información que devuelve la vista administrativa SNAPSWITCHES y la función de tabla SNAP_GET_SWITCHES (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|--------------------|---------------|---|
| UOW_SW_TIME | TIMESTAMP | Si el conmutador de registro del supervisor de unidad de trabajo está activado, la fecha y la hora a la que se activó este conmutador. |
| STATEMENT_SW_STATE | SMALLINT | Estado del conmutador de registro del supervisor de sentencia de SQL (0 ó 1). |
| STATEMENT_SW_TIME | TIMESTAMP | Si el conmutador de registro del supervisor de sentencia de SQL está activado, la fecha y la hora a la que se activó este conmutador. |
| TABLE_SW_STATE | SMALLINT | Estado del conmutador de registro del supervisor de actividad de tabla (0 ó 1). |
| TABLE_SW_TIME | TIMESTAMP | Si el conmutador de registro del supervisor de actividad de tabla está activado, la fecha y la hora a la que se activó este conmutador. |
| BUFFPOOL_SW_STATE | SMALLINT | Estado del conmutador de registro del supervisor de actividad de agrupación de almacenamientos intermedios (0 ó 1). |
| BUFFPOOL_SW_TIME | TIMESTAMP | Si el conmutador de registro del supervisor de actividad de agrupación de almacenamientos intermedios está activado, la fecha y la hora a la que se activó este conmutador. |
| LOCK_SW_STATE | SMALLINT | Estado del conmutador de registro del supervisor de bloqueo (0 ó 1). |
| LOCK_SW_TIME | TIMESTAMP | Si el conmutador de registro del supervisor de bloqueo está activado, la fecha y la hora a la que se activó este conmutador. |
| SORT_SW_STATE | SMALLINT | Estado del conmutador de registro del supervisor de clasificación (0 ó 1). |
| SORT_SW_TIME | TIMESTAMP | Si el conmutador de registro del supervisor de clasificación está activado, la fecha y la hora a la que se activó este conmutador. |
| TIMESTAMP_SW_STATE | SMALLINT | Estado del conmutador de registro del supervisor de indicación de fecha y hora (0 ó 1) |
| TIMESTAMP_SW_TIME | TIMESTAMP | Si el conmutador de registro del supervisor de indicación de fecha y hora está activado, la fecha y la hora a la que se activó este conmutador. |

Tabla 186. Información que devuelve la vista administrativa SNAPSWITCHES y la función de tabla SNAP_GET_SWITCHES (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa SNAPTAB y función de tabla SNAP_GET_TAB_V91 – Recuperar información de instantáneas del grupo de datos lógicos table

La vista administrativa SNAPTAB y la función de tabla SNAP_GET_TAB_V91 devuelven información de la instantánea del grupo de datos lógicos table.

SNAPTAB, vista administrativa

Esta vista administrativa permite recuperar la información de instantáneas del grupo de datos lógicos table para la base de datos conectada actualmente.

La vista administrativa SNAPTAB, utilizada con la vista administrativa SNAPTAB_REORG, devuelve información equivalente a la que proporciona el mandato CLP GET SNAPSHOT FOR TABLES ON alias de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 157 en la página 631 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPTAB
- Privilegio CONTROL en la vista administrativa SNAPTAB
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_TAB_V91
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar el esquema y el nombre de todas las tablas activas.

```
SELECT SUBSTR(TABSHEMA,1,8), SUBSTR(TABNAME,1,15) AS TABNAME, TAB_TYPE,
       DBPARTITIONNUM FROM SYSIBMADM.SNAPTAB
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| TABSCHEMA | TABNAME | TAB_TYPE | DBPARTITIONNUM |
|-----------|---------------|------------|----------------|
| SYSTOOLS | HMON_ATM_INFO | USER_TABLE | 0 |

1 registro(s) seleccionado(s).

SNAP_GET_TAB_V91, función de tabla

La función de tabla SNAP_GET_TAB_V91 devuelve la misma información que la vista administrativa SNAPTAB, pero le permite recuperar la información para una base de datos específica de una partición de base de datos específica, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_TAB_V91, utilizada con la función de tabla SNAP_GET_TAB_REORG, devuelve información equivalente a la que proporciona el mandato CLP GET SNAPSHOT FOR TABLES ON alias de base de datos.

Consulte la Tabla 157 en la página 631 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
▶▶ SNAP_GET_TAB_V91 (—nombrebd— [ , númparticiónbd ] )▶▶
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique NULL o una serie vacía para tomar la instantánea de la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_TAB_V91 tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_TAB_V91
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMANT
- SYSADM

Ejemplo

Recuperar una lista de tablas activas como vista de agregación para la base de datos conectada actualmente.

```
SELECT SUBSTR(TABSCHEMA,1,8) AS TABSCHEMA, SUBSTR(TABNAME,1,15) AS TABNAME,
       TAB_TYPE, DBPARTITIONNUM FROM TABLE(SNAP_GET_TAB('','-2')) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
TABSCHEMA TABNAME          TAB_TYPE          DBPARTITIONNUM
-----
SYSTOOLS  HMON_ATM_INFO      USER_TABLE        -
JESSICAE  EMPLOYEE           USER_TABLE        -
```

Información devuelta

Tabla 187. Información devuelta por la vista administrativa SNAPTAB y la función de tabla SNAP_GET_TAB_V91

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| TABSCHEMA | VARCHAR(128) | table_schema - Nombre de esquema de tabla |
| TABNAME | VARCHAR(128) | table_name - Nombre de tabla |
| TAB_FILE_ID | BIGINT | table_file_id - Identificación de archivo de tabla |
| TAB_TYPE | VARCHAR(14) | table_type - Tipo de tabla. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • USER_TABLE • DROPPED_TABLE • TEMP_TABLE • CATALOG_TABLE • REORG_TABLE |
| DATA_OBJECT_PAGES | BIGINT | data_object_pages - Páginas de objeto de datos |

Tabla 187. Información devuelta por la vista administrativa SNAPTAB y la función de tabla SNAP_GET_TAB_V91 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|--|
| INDEX_OBJECT_PAGES | BIGINT | index_object_pages - Páginas de objeto de índice |
| LOB_OBJECT_PAGES | BIGINT | lob_object_pages - Páginas de objeto LOB |
| LONG_OBJECT_PAGES | BIGINT | long_object_pages - Páginas de objeto largo |
| XDA_OBJECT_PAGES | BIGINT | xda_object_pages - Páginas de objeto XDA |
| ROWS_READ | BIGINT | rows_read - Filas leídas |
| ROWS_WRITTEN | BIGINT | rows_written - Filas grabadas |
| OVERFLOW_ACCESSES | BIGINT | overflow_accesses - Accesos a registros desbordados |
| PAGE_REORGS | BIGINT | page_reorgs - Reorganizaciones de página |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |
| TBSP_ID | BIGINT | tablespace_id - Identificación de espacio de tablas |
| DATA_PARTITION_ID | INTEGER | data_partition_id - Identificador de la partición de datos. Para una tabla no particionada, este elemento será NULL. |

Vista administrativa SNAPTAB_REORG y función de tabla SNAP_GET_TAB_REORG - Recuperar información sobre instantáneas de reorganización de tabla

La vista administrativa SNAPTAB_REORG y la tabla de función SNAP_GET_TAB_REORG devuelven información acerca de la reorganización de tabla. Si no se ha reorganizado ninguna tabla, se devolverán 0 filas.

SNAPTAB_REORG, vista administrativa

Esta vista administrativa le permite recuperar información acerca de la instantánea de reorganización de tabla para la base de datos conectada actualmente.

La vista administrativa SNAPTAB_REORG, utilizada con la vista administrativa SNAPTAB, proporciona datos que son equivalentes a los que proporciona el mandato CLP GET SNAPSHOT FOR TABLES ON alias de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 158 en la página 635 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPTAB_REORG
- Privilegio CONTROL en la vista administrativa SNAPTAB_REORG
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_TAB_REORG
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMAINT
- SYSADM

Ejemplo

Seleccionar información detallada acerca de las operaciones de reorganización para todas las particiones de base de datos de la base de datos conectada actualmente.

```
SELECT SUBSTR(TABNAME, 1, 15) AS TAB_NAME, SUBSTR(TABSHEMA, 1, 15)
      AS TAB_SCHEMA, REORG_PHASE, SUBSTR(REORG_TYPE, 1, 20) AS REORG_TYPE,
      REORG_STATUS, REORG_COMPLETION, DBPARTITIONNUM
FROM SYSIBMADM.SNAPTAB_REORG ORDER BY DBPARTITIONNUM
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| TAB_NAME | TAB_SCHEMA | REORG_PHASE | ... |
|----------|------------|-------------|-----|
| EMPLOYEE | DBUSER | REPLACE | ... |
| EMPLOYEE | DBUSER | REPLACE | ... |
| EMPLOYEE | DBUSER | REPLACE | ... |

3 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

| ... | REORG_TYPE | REORG_STATUS | REORG_COMPLETION | DBPARTITIONNUM |
|-----|----------------------|--------------|------------------|----------------|
| ... | RECLAIM+OFFLINE+ALLO | COMPLETED | SUCCESS | 0 |
| ... | RECLAIM+OFFLINE+ALLO | COMPLETED | SUCCESS | 1 |
| ... | RECLAIM+OFFLINE+ALLO | COMPLETED | SUCCESS | 2 |

Seleccionar toda la información sobre una operación de reorganización para reclamar extensiones de una tabla de clúster multidimensional (MDC) desde la vista administrativa SNAPTAB_REORG.

```
db2 -v "select * from sysibmadm.snaptab_reorg"
```

| TABNAME | REORG_PHASE | REORG_MAX_PHASE | REORG_TYPE |
|---------|-------------|-----------------|-----------------------------|
| T1 | RELEASE | 3 | RECLAIM_EXTENTS+ALLOW_WRITE |

| REORG_STATUS | REORG_COMPLETION | REORG_START | REORG_END |
|--------------|------------------|----------------------------|----------------------------|
| COMPLETED | SUCCESS | 2008-09-24-14.35.30.734741 | 2008-09-24-14.35.31.460674 |

Función de tabla SNAP_GET_TAB_REORG

La función de tabla SNAP_GET_TAB_REORG devuelve la misma información que la vista administrativa SNAPTAB_REORG, pero le permite recuperar la información para una base de datos específica de una partición de base de datos específica, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_TAB_REORG, utilizada con la función de tabla SNAP_GET_TAB, proporciona datos que son equivalentes a los que proporciona el mandato CLP GET SNAPSHOT FOR TABLES ON alias de base de datos.

Consulte la Tabla 158 en la página 635 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
▶▶ SNAP_GET_TAB_REORG ( ( nombrebd [ , numparticiónbd ] ) ) ▶▶
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique NULL o una serie vacía para tomar la instantánea de la base de datos conectada actualmente.

numparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número de partición de base de datos válido. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *numparticiónbd* se establece en NULL, -1 se establece implícitamente para *numparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *numparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_TAB_REORG tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_TAB_REORG

- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMMAINT
- SYSADM

Ejemplo

Seleccionar información detallada acerca de las operaciones de reorganización para la partición de base de datos 1 de la base de datos conectada actualmente.

```
SELECT SUBSTR(TABNAME, 1, 15) AS TAB_NAME, SUBSTR(TABSHEMA, 1, 15)
      AS TAB_SCHEMA, REORG_PHASE, SUBSTR(REORG_TYPE, 1, 20) AS REORG_TYPE,
      REORG_STATUS, REORG_COMPLETION, DBPARTITIONNUM
FROM TABLE( SNAP_GET_TAB_REORG('', 1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
TAB_NAME      TAB_SCHEMA    REORG_PHASE    REORG_TYPE      ...
-----
EMPLOYEE      DBUSER        REPLACE        RECLAIM+OFFLINE+ALLO ...
...
1 registro(s) seleccionado(s).      ...
```

Datos de salida de esta consulta (continuación).

```
... REORG_STATUS REORG_COMPLETION DBPARTITIONNUM
... -----
... COMPLETED   SUCCESS                          1
...
...
```

Seleccionar toda la información sobre una operación de reorganización para reclamar extensiones de una tabla de clúster multidimensional (MDC) utilizando la función de tabla SNAP_GET_TAB_REORG.

```
db2 -v "select * from table(snap_get_tab_reorg(''))"
```

```
TABNAME REORG_PHASE    REORG_MAX_PHASE  REORG_TYPE
-----
T1      RELEASE        3                RECLAIM_EXTENTS+ALLOW_WRITE

REORG_STATUS REORG_COMPLETION REORG_START          REORG_END
-----
COMPLETED   SUCCESS          2008-09-24-14.35.30.734741 2008-09-24-14.35.31.460674
```

Información devuelta

Tabla 188. Información devuelta por la vista administrativa SNAPTAB_REORG y la función de tabla SNAP_GET_TAB_REORG

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| TABNAME | VARCHAR (128) | table_name - Nombre de tabla |
| TABSHEMA | VARCHAR (128) | table_schema - Nombre de esquema de tabla |

Tabla 188. Información devuelta por la vista administrativa SNAPTAB_REORG y la función de tabla SNAP_GET_TAB_REORG (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------|---------------|---|
| PAGE_REORGS | BIGINT | page_reorgs - Reorganizaciones de página |
| REORG_PHASE | VARCHAR(16) | reorg_phase - Fase de reorganización de tabla. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • BUILD • DICT_SAMPLE • INDEX_RECREATE • REPLACE • SORT • SCAN • DRAIN • RELEASE o SORT+DICT_SAMPLE. |
| REORG_MAX_PHASE | INTEGER | reorg_max_phase - Fase máxima de reorganización de tabla |
| REORG_CURRENT_COUNTER | BIGINT | reorg_current_counter - Progreso de reorganización de tabla |
| REORG_MAX_COUNTER | BIGINT | reorg_max_counter - Cantidad total de reorganización de tabla |

Tabla 188. Información devuelta por la vista administrativa SNAPTAB_REORG y la función de tabla SNAP_GET_TAB_REORG (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|--|
| REORG_TYPE | VARCHAR (128) | <p>reorg_type - Atributos de reorganización de tabla. Esta interfaz devuelve un identificador de texto utilizando una combinación de los siguientes identificadores separados mediante '+':</p> <p>Uno de los siguientes:</p> <ul style="list-style-type: none"> • RECLAIM • RECLUSTER • RECLAIM_EXTS <p>y uno de los siguientes:</p> <ul style="list-style-type: none"> • +OFFLINE • +ONLINE <p>Si se especifica la modalidad de acceso, es uno de los siguientes:</p> <ul style="list-style-type: none"> • +ALLOW_NONE • +ALLOW_READ • +ALLOW_WRITE <p>Si está fuera de línea y con la opción RECLUSTER, uno de los siguientes:</p> <ul style="list-style-type: none"> • +INDEXSCAN • +TABLESCAN <p>Si está fuera de línea, uno de los siguientes:</p> <ul style="list-style-type: none"> • +LONGLOB • +DATAONLY <p>Si está fuera de línea, y se especifica una opción, uno de los siguientes:</p> <ul style="list-style-type: none"> • +CHOOSE_TEMP • +KEEPDICTIONARY • +RESETDICTIONARY <p>Si está en línea, y se especifica una opción:</p> <ul style="list-style-type: none"> • +NOTRUNCATE <p>Ejemplo 1: Si se ha ejecutado REORG TABLE TEST.EMPLOYEE, se visualizaría lo siguiente: RECLAIM+OFFLINE+ALLOW_READ+DATAONLY+KEEPDICTIONARY</p> <p>Ejemplo 2: Si se ha ejecutado REORG TABLE TEST.EMPLOYEE INDEX EMPIDX INDEXSCAN, se visualizaría lo siguiente: RECLUSTER+OFFLINE+ALLOW_READ+INDEXSCAN+DATAONLY+KEEPDICTIONARY</p> |

Tabla 188. Información devuelta por la vista administrativa SNAPTAB_REORG y la función de tabla SNAP_GET_TAB_REORG (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|----------------------|---------------|--|
| REORG_STATUS | VARCHAR (10) | reorg_status - Estado de reorganización de tabla. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • COMPLETED • PAUSED • STARTED • STOPPED • TRUNCATE |
| REORG_COMPLETION | VARCHAR (10) | reorg_completion - Distintivo de finalización de reorganización de tabla. Esta interfaz devuelve un identificador de texto, basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • FAIL • SUCCESS |
| REORG_START | TIMESTAMP | reorg_start - Hora de inicio de reorganización de tabla |
| REORG_END | TIMESTAMP | reorg_end - Hora de finalización de reorganización de tabla |
| REORG_PHASE_START | TIMESTAMP | reorg_phase_start - Hora de inicio de fase de reorganización de tabla |
| REORG_INDEX_ID | BIGINT | reorg_index_id - Índice que se utiliza para reorganizar la tabla |
| REORG_TBSPC_ID | BIGINT | reorg_tbspc_id - Espacio de tablas donde se reorganiza la tabla |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |
| DATA_PARTITION_ID | INTEGER | data_partition_id - Identificador de la partición de datos. Para una tabla no particionada, este elemento será NULL. |
| REORG_ROWSCOMPRESSED | BIGINT | reorg_rows_compressed - Filas comprimidas |
| REORG_ROWSREJECTED | BIGINT | reorg_rows_rejected_for_compression - Filas rechazadas para compresión |
| REORG_LONG_TBSPC_ID | BIGINT | reorg_long_tbspc_id - Espacio de tabla donde se reorganizan objetos grandes |

Vista administrativa SNAPTBSP y función de tabla SNAP_GET_TBSP_V91 – Recuperar información de instantáneas del grupo de datos lógicos de espacio de tablas

La vista administrativa SNAPTBSP y la función de tabla SNAP_GET_TBSP_V91 devuelven información de la instantánea del grupo de datos lógicos de espacio de tablas.

SNAPTBSP, vista administrativa

Esta vista administrativa permite recuperar la información de instantáneas del grupo de datos lógicos de espacio de tablas para la base de datos conectada actualmente.

La vista administrativa SNAPTBSP, utilizada con las vistas administrativas SNAPTBSP_PART, SNAPTBSP QUIESCER, SNAPTBSP_RANGE y SNAPCONTAINER, devuelve información equivalente a la que proporciona el mandato CLP GET SNAPSHOT FOR TABLESPACES ON alias de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 159 en la página 641 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPTBSP
- Privilegio CONTROL en la vista administrativa SNAPTBSP
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_TBSP_V91
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMAINT
- SYSADM

Ejemplo

Recuperar una lista de los espacios de tablas de la partición de base de datos de catálogo para la base de datos conectada actualmente.

```
SELECT SUBSTR(TBSP_NAME,1,30) AS TBSP_NAME, TBSP_ID, TBSP_TYPE,  
       TBSP_CONTENT_TYPE FROM SYSIBMADM.SNAPTBSP WHERE DBPARTITIONNUM = 1
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| TBSP_NAME | TBSP_ID | TBSP_TYPE | TBSP_CONTENT_TYPE |
|------------|---------|-----------|-------------------|
| TEMPSPACE1 | 1 | SMS | SYSTEMP |
| USERSPACE1 | 2 | DMS | LONG |

2 registro(s) seleccionado(s).

SNAP_GET_TBSP_V91, función de tabla

La función de tabla SNAP_GET_TBSP_V91 devuelve la misma información que la vista administrativa SNAPTbsp, pero le permite recuperar la información para una base de datos específica de una partición de base de datos específica, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_TBSP_V91, utilizada con las funciones de tabla SNAP_GET_TBSP_PART_V91, SNAP_GET_TBSP QUIESCER, SNAP_GET_TBSP_RANGE y SNAP_GET_CONTAINER_V91, devuelve información que es equivalente a la que proporciona el mandato CLP GET SNAPSHOT FOR TABLESPACES ON alias de base de datos.

Consulte la Tabla 159 en la página 641 para ver una lista completa de la información que se puede devolver.

Sintaxis

```

▶▶ SNAP_GET_TBSP_V91 (—nombrebd— [ , númparticiónbd ] )

```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique NULL o una serie vacía para tomar la instantánea de la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_TBSP_V91 tomará una instantánea del número de partición de base de

datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_TBSP_V91
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMAINT
- SYSADM

Ejemplo

Recuperar una lista de los espacios de tablas de todas las particiones de base de datos para la base de datos conectada actualmente.

```
SELECT SUBSTR(TBSP_NAME,1,10) AS TBSP_NAME, TBSP_ID, TBSP_TYPE,
       TBSP_CONTENT_TYPE, DBPARTITIONNUM FROM TABLE(SNAP_GET_TBSP_V91('')) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| TBSP_NAME | TBSP_ID | TBSP_TYPE | TBSP_CONTENT_TYPE | DBPARTITIONNUM |
|------------|---------|-----------|-------------------|----------------|
| TEMPSPACE1 | 1 | SMS | SYSTEMP | 1 |
| USERSPACE1 | 2 | DMS | LONG | 1 |
| SYSCATSPAC | 0 | DMS | ANY | 0 |
| TEMPSPACE1 | 1 | SMS | SYSTEMP | 0 |
| USERSPACE1 | 2 | DMS | LONG | 0 |
| SYSTOOLSPA | 3 | DMS | LONG | 0 |
| TEMPSPACE1 | 1 | SMS | SYSTEMP | 2 |
| USERSPACE1 | 2 | DMS | LONG | 2 |

8 registro(s) seleccionado(s).

Información devuelta

Tabla 189. Información devuelta por la vista administrativa SNAPTBSP y la función de tabla SNAP_GET_TBSP_V91

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| TBSP_NAME | VARCHAR(128) | tablespace_name - Nombre de espacio de tablas |
| TBSP_ID | BIGINT | tablespace_id - Identificación de espacio de tablas |
| TBSP_TYPE | VARCHAR(10) | tablespace_type - Tipo de espacio de tablas. Esta interfaz devuelve un identificador de texto basado en sentencias define en sqlutil.h y es uno de los siguientes: <ul style="list-style-type: none"> • DMS • SMS |

Tabla 189. Información devuelta por la vista administrativa SNAPTBSP y la función de tabla SNAP_GET_TBSP_V91 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------|---------------|---|
| TBSP_CONTENT_TYPE | VARCHAR(10) | tablespace_content_type - Tipo de contenido de espacio de tablas. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • ANY • LARGE • SYSTEMP • USRTEMP |
| TBSP_PAGE_SIZE | BIGINT | tablespace_page_size - Tamaño de página de espacio de tablas |
| TBSP_EXTENT_SIZE | BIGINT | tablespace_extent_size - Tamaño de extensión de espacio de tablas |
| TBSP_PREFETCH_SIZE | BIGINT | tablespace_prefetch_size - Tamaño de captación previa de espacio de tablas |
| TBSP_CUR_POOL_ID | BIGINT | tablespace_cur_pool_id - Agrupación de almacenamiento intermedio utilizada actualmente |
| TBSP_NEXT_POOL_ID | BIGINT | tablespace_next_pool_id - Agrupación de almacenamiento intermedio que se utilizará en el siguiente arranque |
| FS_CACHING | SMALLINT | fs_caching - Antememoria del sistema de archivos |
| POOL_DATA_L_READS | BIGINT | pool_data_l_reads - Lecturas lógicas de datos de la agrupación de almacenamientos intermedios |
| POOL_DATA_P_READS | BIGINT | pool_data_p_reads - Lecturas físicas de datos de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_L_READS | BIGINT | pool_temp_data_l_reads - Lecturas lógicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_P_READS | BIGINT | pool_temp_data_p_reads - Lecturas físicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_READS | BIGINT | pool_async_data_reads - Lecturas de datos asíncronas de agrupación de almacenamientos intermedios |
| POOL_DATA_WRITES | BIGINT | pool_data_writes - Grabaciones de datos de la agrupación de almacenamientos intermedios |

Tabla 189. Información devuelta por la vista administrativa SNAPTBSP y la función de tabla SNAP_GET_TBSP_V91 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|---|
| POOL_ASYNC_DATA_WRITES | BIGINT | pool_async_data_writes - Grabaciones de datos asíncronas de agrupación de almacenamientos intermedios |
| POOL_INDEX_L_READS | BIGINT | pool_index_l_reads - Lecturas lógicas de índices de la agrupación de almacenamientos intermedios |
| POOL_INDEX_P_READS | BIGINT | pool_index_p_reads - Lecturas físicas de índices de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_L_READS | BIGINT | pool_temp_index_l_reads - Lecturas lógicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_P_READS | BIGINT | pool_temp_index_p_reads - Lecturas físicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_READS | BIGINT | pool_async_index_reads - Lecturas de índice asíncronas de agrupación de almacenamientos intermedios |
| POOL_INDEX_WRITES | BIGINT | pool_index_writes - Grabaciones de índices de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_WRITES | BIGINT | pool_async_index_writes - Grabaciones de índice asíncronas de agrupación de almacenamientos intermedios |
| POOL_XDA_L_READS | BIGINT | pool_xda_l_reads - Lecturas lógicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_XDA_P_READS | BIGINT | pool_xda_p_reads - Lecturas físicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_XDA_WRITES | BIGINT | pool_xda_writes - Grabaciones de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_XDA_READS | BIGINT | pool_async_xda_reads - Lecturas de datos XDA asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_XDA_WRITES | BIGINT | pool_async_xda_writes - Grabaciones de datos XDA asíncronas de agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_L_READS | BIGINT | pool_temp_xda_l_reads - Lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |

Tabla 189. Información devuelta por la vista administrativa SNAPT BSP y la función de tabla SNAP_GET_TBSP_V91 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|----------------------------|---------------|--|
| POOL_TEMP_XDA_P_READS | BIGINT | pool_temp_xda_p_reads - Elemento del supervisor de las lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |
| POOL_READ_TIME | BIGINT | pool_read_time - Tiempo total de lectura física de la agrupación de almacenamientos intermedios |
| POOL_WRITE_TIME | BIGINT | pool_write_time - Tiempo total de grabación física de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_READ_TIME | BIGINT | pool_async_read_time - Tiempo de lectura asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_WRITE_TIME | BIGINT | pool_async_write_time - Tiempo de grabación asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_READ_REQS | BIGINT | pool_async_data_read_reqs - Peticiones de lectura asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_READ_REQS | BIGINT | pool_async_index_read_reqs - Peticiones de lectura de índice asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_XDA_READ_REQS | BIGINT | pool_async_xda_read_reqs - Peticiones de lectura XDA asíncrona de agrupación de almacenamientos intermedios |
| POOL_NO_VICTIM_BUFFER | BIGINT | pool_no_victim_buffer - Almacenamiento intermedio sin víctimas de la agrupación de almacenamientos intermedios |
| DIRECT_READS | BIGINT | direct_reads - Lecturas directas de la base de datos |
| DIRECT_WRITES | BIGINT | direct_writes - Grabaciones directas en la base de datos |
| DIRECT_READ_REQS | BIGINT | direct_read_reqs - Peticiones de lectura directa |
| DIRECT_WRITE_REQS | BIGINT | direct_write_reqs - Peticiones de grabación directa |
| DIRECT_READ_TIME | BIGINT | direct_read_time - Tiempo de lectura directa |
| DIRECT_WRITE_TIME | BIGINT | direct_write_time - Tiempo de grabación directa |
| FILES_CLOSED | BIGINT | files_closed - Archivos de base de datos cerrados |

Tabla 189. Información devuelta por la vista administrativa SNAPTBSP y la función de tabla SNAP_GET_TBSP_V91 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------------|---------------|--|
| UNREAD_PREFETCH_PAGES | BIGINT | unread_prefetch_pages - Páginas de captación previa no leídas |
| TBSP_REBALANCER_MODE | VARCHAR(10) | tablespace_rebalancer_mode - Modalidad de reequilibrador. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h, y es uno de los siguientes: <ul style="list-style-type: none"> • NO_REBAL • FWD_REBAL • REV_REBAL |
| TBSP_USING_AUTO_STORAGE | SMALLINT | tablespace_using_auto_storage - Espacio de tablas habilitado para el almacenamiento automático |
| TBSP_AUTO_RESIZE_ENABLED | SMALLINT | tablespace_auto_resize_enabled - Espacio de tablas habilitado para la modificación de tamaño automática |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa SNAPTBSP_PART y función de tabla SNAP_GET_TBSP_PART_V91 – Recuperar información de instantáneas del grupo de datos lógicos tablespace_nodeinfo

Nota: Esta función de tabla ha quedado en desuso y se ha sustituido por la “Vista administrativa SNAPTBSP_PART y función de tabla SNAP_GET_TBSP_PART_V97 – Recuperar información de instantáneas del grupo de datos lógicos tablespace_nodeinfo” en la página 645.

La vista administrativa SNAPTBSP_PART y la función de tabla SNAP_GET_TBSP_PART_V91 devuelven información de la instantánea del grupo de datos lógicos tablespace_nodeinfo.

Vista administrativa SNAPTBSP_PART

Esta vista administrativa permite recuperar la información de instantáneas del grupo de datos lógicos tablespace_nodeinfo para la base de datos conectada actualmente.

La vista administrativa SNAPTBSP_PART, utilizada con las vistas administrativas SNAPTBSP, SNAPTBSP QUIESCER, SNAPTBSP_RANGE y SNAPCONTAINER, devuelve información equivalente a la que proporciona el mandato CLP GET SNAPSHOT FOR TABLESPACES ON alias de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 190 en la página 800 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPTBSP_PART
- Privilegio CONTROL en la vista administrativa SNAPTBSP_PART
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_TBSP_PART_V91
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMAINT
- SYSADM

Ejemplo

Recuperar una lista de los espacios de tablas, y el estado de éstos, de todas las particiones de base de datos de la base de datos conectada actualmente.

```
SELECT SUBSTR(TBSP_NAME,1,30) AS TBSP_NAME, TBSP_ID,
       SUBSTR(TBSP_STATE,1,30) AS TBSP_STATE, DBPARTITIONNUM
FROM SYSIBMADM.SNAPTbsp_PART
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| TBSP_NAME | TBSP_ID | TBSP_STATE | DBPARTITIONNUM |
|-------------|---------|------------|----------------|
| SYSCATSPACE | 0 | NORMAL | 0 |
| TEMPSPACE1 | 1 | NORMAL | 0 |
| USERSPACE1 | 2 | NORMAL | 0 |
| TEMPSPACE1 | 1 | NORMAL | 1 |
| USERSPACE1 | 2 | NORMAL | 1 |

5 registro(s) seleccionado(s).

SNAP_GET_TBSP_PART_V91, función de tabla

La función de tabla SNAP_GET_TBSP_PART_V91 devuelve la misma información que la vista administrativa SNAPTBSP_PART, pero le permite recuperar la información para una base de datos específica de una partición de base de datos específica, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_TBSP_PART_V91, utilizada con las funciones de tabla SNAP_GET_TBSP_V91, SNAP_GET_TBSP QUIESCER, SNAP_GET_TBSP_RANGE y SNAP_GET_CONTAINER_V91, devuelve información que es equivalente a la que proporciona el mandato CLP GET SNAPSHOT FOR TABLESPACES ON alias de base de datos.

Consulte la Tabla 190 en la página 800 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
▶▶ SNAP_GET_TBSP_PART_V91 (—nombre-bd— [ , númparticiónbd ] )▶▶
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique NULL o una serie vacía para tomar la instantánea de la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_TBSP_PART_V91 tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_TBSP_PART_V91
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar una lista de los espacios de tablas, y el estado de éstos, de la partición de base de datos conectada de la base de datos conectada.

```
SELECT SUBSTR(TBSP_NAME,1,30) AS TBSP_NAME, TBSP_ID,  
       SUBSTR(TBSP_STATE,1,30) AS TBSP_STATE  
FROM TABLE(SNAP_GET_TBSP_PART_V91(CAST(NULL AS VARCHAR(128)),-1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| TBSP_NAME | TBSP_ID | TBSP_STATE |
|------------------|---------|------------|
| SYSCATSPACE | | 0 NORMAL |
| TEMPSPACE1 | | 1 NORMAL |
| USERSPACE1 | | 2 NORMAL |
| SYSTOOLSPACE | | 3 NORMAL |
| SYSTOOLSTMPSPACE | | 4 NORMAL |

5 registro(s) seleccionado(s).

Información devuelta

Tabla 190. Información devuelta por la vista administrativa *SNAPTbsp_PART* y la función de tabla *SNAP_GET_TBSP_PART_V91*

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|------------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| TBSP_NAME | VARCHAR (128) | tablespace_name - Nombre de espacio de tablas |
| TBSP_ID | BIGINT | tablespace_id - Identificación de espacio de tablas |

Tabla 190. Información devuelta por la vista administrativa SNAPTBSP_PART y la función de tabla SNAP_GET_TBSP_PART_V91 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------------|------------------|--|
| TBSP_STATE | VARCHAR (256) | <p>tablespace_state - Estado de espacio de tablas. Esta interfaz devuelve un identificador de texto basado en las sentencias define en sqlutil.h y es una combinación de los siguientes elementos separados por un signo '+':</p> <ul style="list-style-type: none"> • BACKUP_IN_PROGRESS • BACKUP_PENDING • DELETE_PENDING • DISABLE_PENDING • DROP_PENDING • LOAD_IN_PROGRESS • LOAD_PENDING • NORMAL • OFFLINE • PSTAT_CREATION • PSTAT_DELETION • QUIESCED_EXCLUSIVE • QUIESCED_SHARE • QUIESCED_UPDATE • REBAL_IN_PROGRESS • REORG_IN_PROGRESS • RESTORE_IN_PROGRESS • RESTORE_PENDING • ROLLFORWARD_IN_PROGRESS • ROLLFORWARD_PENDING • STORDEF_ALLOWED • STORDEF_CHANGED • STORDEF_FINAL_VERSION • STORDEF_PENDING • SUSPEND_WRITE |
| TBSP_PREFETCH_SIZE | BIGINT | tablespace_prefetch_size - Tamaño de captación previa de espacio de tablas |
| TBSP_NUM QUIESCERS | BIGINT | tablespace_num_quiescers - Número de inmovilizadores |
| TBSP_STATE_CHANGE_OBJECT_ID | BIGINT | tablespace_state_change_object_id - Identificación de objeto de cambio de estado |
| TBSP_STATE_CHANGE_TBSP_ID | BIGINT | tablespace_state_change_ts_id - Identificación de espacio de tablas de cambio de estado |
| TBSP_MIN_RECOVERY_TIME | TIMESTAMP | tablespace_min_recovery_time - Tiempo mínimo de recuperación para la recuperación en avance |
| TBSP_TOTAL_PAGES | BIGINT | tablespace_total_pages - Total de páginas en espacio de tablas |

Tabla 190. Información devuelta por la vista administrativa SNAPTBSP_PART y la función de tabla SNAP_GET_TBSP_PART_V91 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------------|---------------|---|
| TBSP_USABLE_PAGES | BIGINT | tablespace_usable_pages - Páginas utilizables de espacio de tablas |
| TBSP_USED_PAGES | BIGINT | tablespace_used_pages - Páginas utilizadas en espacio de tablas |
| TBSP_FREE_PAGES | BIGINT | tablespace_free_pages - Páginas libres en espacio de tablas |
| TBSP_PENDING_FREE_PAGES | BIGINT | tablespace_pending_free_pages - Páginas libres pendientes en espacio de tablas |
| TBSP_PAGE_TOP | BIGINT | tablespace_page_top - Nivel límite de espacio de tablas |
| REBALANCER_MODE | VARCHAR (10) | tablespace_rebalancer_mode - Modalidad de reequilibrador. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • FWD_REBAL • NO_REBAL • REV_REBAL |
| REBALANCER_EXTENTS_REMAINING | BIGINT | tablespace_rebalancer_extents_remaining - Número total de extensiones que el reequilibrador debe procesar |
| REBALANCER_EXTENTS_PROCESSED | BIGINT | tablespace_rebalancer_extents_processed - Número de extensiones que el reequilibrador ha procesado |
| REBALANCER_PRIORITY | BIGINT | tablespace_rebalancer_priority - Prioridad de reequilibrador actual |
| REBALANCER_START_TIME | TIMESTAMP | tablespace_rebalancer_start_time - Hora de inicio de reequilibrador |
| REBALANCER_RESTART_TIME | TIMESTAMP | tablespace_rebalancer_restart_time - Hora de reinicio de reequilibrador |
| REBALANCER_LAST_EXTENT_MOVED | BIGINT | tablespace_rebalancer_last_extent_moved - Última extensión movida por el reequilibrador |
| TBSP_NUM_RANGES | BIGINT | tablespace_num_ranges - Número de rangos de la correlación de espacio de tablas |
| TBSP_NUM_CONTAINERS | BIGINT | tablespace_num_containers - Número de contenedores de espacio de tablas |
| TBSP_INITIAL_SIZE | BIGINT | tablespace_initial_size - Tamaño inicial de espacio de tablas |
| TBSP_CURRENT_SIZE | BIGINT | tablespace_current_size - Tamaño actual de espacio de tablas |
| TBSP_MAX_SIZE | BIGINT | tablespace_max_size - Tamaño máximo de espacio de tablas |

Tabla 190. Información devuelta por la vista administrativa `SNAPTbsp_Part` y la función de tabla `SNAP_Get_Tbsp_Part_V91` (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---|---------------|---|
| <code>Tbsp_Increase_Size</code> | BIGINT | <code>tablespace_increase_size</code> - Aumento de tamaño en bytes |
| <code>Tbsp_Increase_Size_Percent</code> | SMALLINT | <code>tablespace_increase_size_percent</code> - Aumento de tamaño en porcentaje |
| <code>Tbsp_Last_Resize_Time</code> | TIMESTAMP | <code>tablespace_last_resize_time</code> - Hora del último redimensionamiento satisfactorio |
| <code>Tbsp_Last_Resize_Failed</code> | SMALLINT | <code>tablespace_last_resize_failed</code> - Último intento de redimensionamiento fallido |
| <code>DBPARTITIONNUM</code> | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa `SNAPTbsp_QUIESCER` y función de tabla `SNAP_Get_Tbsp_QUIESCER` – Recuperar información acerca de la instantánea de espacio de tablas de inmovilizador

La vista administrativa `SNAPTbsp_QUIESCER` y la función de tabla `SNAP_Get_Tbsp_QUIESCER` devuelven información sobre inmovilizadores desde una instantánea del espacio de tablas.

`SNAPTbsp_QUIESCER`, vista administrativa

Esta vista administrativa le permite recuperar información acerca de la instantánea de espacio de tablas de inmovilizador para la base de datos conectada actualmente.

La vista administrativa `SNAPTbsp_QUIESCER`, utilizada con las vistas administrativas `SNAPTbsp`, `SNAPTbsp_Part`, `SNAPTbsp_Range`, `SNAPCONTAINER`, proporciona información que es equivalente a la información que proporciona el mandato `CLP GET SNAPSHOT FOR TABLESPACES ON` alias de base de datos.

El esquema es `SYSIBMADM`.

Consulte la Tabla 161 en la página 654 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio `SELECT` en la vista administrativa `SNAPTbsp_QUIESCER`
- Privilegio `CONTROL` en la vista administrativa `SNAPTbsp_QUIESCER`
- Autorización `DATAACCESS`

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio `EXECUTE` para la función de tabla `SNAP_Get_Tbsp_QUIESCER`
- Autorización `DATAACCESS`

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCtrl
- SYSMaint
- SYSADM

Ejemplo

Recuperar información acerca de los espacios de tablas inmovilizados para todas las particiones de base de datos de la base de datos conectada actualmente.

```
SELECT SUBSTR(TBSP_NAME, 1, 10) AS TBSP_NAME, QUIESCER_TS_ID,
       QUIESCER_OBJ_ID, QUIESCER_AUTH_ID, QUIESCER_AGENT_ID,
       QUIESCER_STATE, DBPARTITIONNUM
FROM SYSIBMADM.SNAPTbsp QUIESCER ORDER BY DBPARTITIONNUM
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| TBSP_NAME | QUIESCER_TS_ID | QUIESCER_OBJ_ID | QUIESCER_AUTH_ID | .. |
|------------|----------------|-----------------|------------------|----|
| USERSPACE1 | 2 | 5 | SWALKTY | .. |
| USERSPACE1 | 2 | 5 | SWALKTY | .. |

2 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

| ... QUIESCER_AGENT_ID | QUIESCER_STATE | DBPARTITIONNUM |
|-----------------------|-----------------|----------------|
| ... | 0 EXCLUSIVE | 0 |
| ... | 65983 EXCLUSIVE | 1 |

Ejemplo: determinar los nombres de las tablas particionadas de rangos

Si la tabla está particionada por rangos se mantiene en estado inmovilizado, los diferentes valores del ID de espacio de tablas y del ID de tabla se representan en SYSCAT.TABLES. Estos ID se mostrarán como descripciones breves sin signo. Para encontrar el nombre de la tabla inmovilizada, quizás necesite encontrar primero la descripción breve con signo calculando para ello el ID de espacio de tablas que resta 65536 (el valor máximo) de QEUIESCER_TS_ID y, a continuación, utilizar este ID de espacio de tablas para localizar las tablas inmovilizadas. (El ID de espacio de tablas real se encuentra en SYSCAT.DATAPARTITIONS para cada partición de rango de la tabla).

```
SELECT SUBSTR(TBSP_NAME, 1, 10) AS TBSP_NAME,
       CASE WHEN QUIESCER_TS_ID = 65530
            THEN QUIESCER_TS_ID - 65536
            ELSE QUIESCER_TS_ID END as tbspaceid,
       CASE WHEN QUIESCER_TS_ID = 65530
            THEN QUIESCER_OBJ_ID - 65536
            ELSE QUIESCER_OBJ_ID END as tableid
FROM SYSIBMADM.SNAPTbsp QUIESCER
ORDER BY DBPARTITIONNUM
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| TBSP_NAME | TBSPACEID | TABLEID |
|-----------|-----------|---------|
| TABDATA | -6 | -32768 |


```
DATAMART      -6          -32765
SMALL         5           17
```

3 registro(s) seleccionado(s).

Utilice los valores de TBSPACEID y TABLEID proporcionados por la consulta anterior para encontrar el nombre y el esquema de tabla desde SYSCAT.TABLES.

```
SELECT CHAR(tabschema, 10)tabschema, CHAR(tabname,15)tabname
FROM SYSCAT.TABLES
WHERE tbspaceid = -6 AND tableid in (-32768,-32765)
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
TABSCHEMA    TABNAME
-----
TPCD         ORDERS_RP
TPCD         ORDERS_DMART
```

2 registro(s) seleccionado(s).

```
SELECT CHAR(tabschema, 10)tabschema, CHAR(tabname,15)tabname
FROM SYSCAT.TABLES
WHERE tbspaceid = 5 AND tableid = 17
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
TABSCHEMA    TABNAME
-----
TPCD         NATION
```

1 registro(s) seleccionado(s).

Función de tabla SNAP_GET_TBSP QUIESCER

La función de tabla SNAP_GET_TBSP QUIESCER devuelve la misma información que la vista administrativa SNAPTbsp QUIESCER, pero le permite recuperar la información para una base de datos específica de una partición de una base de datos específica, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_TBSP QUIESCER, utilizada con las funciones de tabla SNAP_GET_TBSP_V91, SNAP_GET_TBSP_PART_V91, SNAP_GET_TBSP_RANGE, SNAP_GET_CONTAINER_V91, proporciona información que es equivalente a la información que proporciona el mandato CLP GET SNAPSHOT FOR TABLESPACES ON alias de base de datos.

Consulte la Tabla 161 en la página 654 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
→ SNAP_GET_TBSP QUIESCER (—nombrebd—, númparticiónbd) →
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada

actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique NULL o una serie vacía para tomar la instantánea de la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_TBSP QUIESCER tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_TBSP QUIESCER
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMANT
- SYSADM

Ejemplo

Recuperar información acerca de los espacios de tablas inmovilizados para la partición de base de datos 1 de la base de datos conectada actualmente.

```
SELECT SUBSTR(TBSP_NAME, 1, 10) AS TBSP_NAME, QUIESCER_TS_ID,
       QUIESCER_OBJ_ID, QUIESCER_AUTH_ID, QUIESCER_AGENT_ID,
       QUIESCER_STATE, DBPARTITIONNUM
FROM TABLE( SYSPROC.SNAP_GET_TBSP QUIESCER( ' ', 1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| TBSP_NAME | QUIESCER_TS_ID | QUIESCER_OBJ_ID | QUIESCER_AUTH_ID | ... |
|------------|----------------|-----------------|------------------|-----|
| USERSPACE1 | 2 | 5 | SWALKTY | ... |

1 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```

... QUIESCER_AGENT_ID    QUIESCER_STATE DBPARTITIONNUM
... -----
...                      65983 EXCLUSIVE                      1

```

Información devuelta

Tabla 191. Información devuelta por la vista administrativa *SNAPTbsp_QUIESCER* y la función de tabla *SNAP_GET_TBSP_QUIESCER*

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| TBSP_NAME | VARCHAR(128) | tablespace_name - Nombre de espacio de tablas |
| QUIESCER_TS_ID | BIGINT | quiescer_ts_id - Identificación del espacio de tablas de inmovilizador |
| QUIESCER_OBJ_ID | BIGINT | quiescer_obj_id - Identificación del objeto de inmovilizador |
| QUIESCER_AUTH_ID | VARCHAR(128) | quiescer_auth_id - Identificación de autorización de usuario de inmovilizador |
| QUIESCER_AGENT_ID | BIGINT | quiescer_agent_id - Identificación de agente de inmovilizador |
| QUIESCER_STATE | VARCHAR(14) | quiescer_state - Estado de inmovilizador. Esta interfaz devuelve un identificador de texto basado en las sentencias define de <code>sqlutil.h</code> y es uno de los siguientes: <ul style="list-style-type: none"> • EXCLUSIVE • UPDATE • SHARE |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa *SNAPTbsp_RANGE* y función de tabla *SNAP_GET_TBSP_RANGE* – Recuperar información acerca de la instantánea de rango

La vista administrativa *SNAPTbsp_RANGE* y la función de tabla *SNAP_GET_TBSP_RANGE* devuelven información desde una instantánea de rango.

SNAPTbsp_RANGE, vista administrativa

Esta vista administrativa le permite recuperar información acerca de la instantánea de rango para la base de datos conectada actualmente.

La vista administrativa *SNAPTbsp_RANGE*, utilizada con las vistas administrativas *SNAPTbsp*, *SNAPTbsp_PART*, *SNAPTbsp_QUIESCER* y

SNAPCONTAINER, proporciona información que es equivalente a la información que proporciona el mandato CLP GET SNAPSHOT FOR TABLESPACES ON alias de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 162 en la página 658 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPTbsp_RANGE
- Privilegio CONTROL en la vista administrativa SNAPTbsp_RANGE
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_TBSP_RANGE
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMMAINT
- SYSADM

Ejemplo

Seleccionar información acerca de los rangos de espacio de tablas para todas las particiones de la base de datos conectada actualmente.

```
SELECT TBSP_ID, SUBSTR(TBSP_NAME, 1, 15) AS TBSP_NAME, RANGE_NUMBER,
       RANGE_STRIPE_SET_NUMBER, RANGE_OFFSET, RANGE_MAX_PAGE,
       RANGE_MAX_EXTENT, RANGE_START_STRIPE, RANGE_END_STRIPE,
       RANGE_ADJUSTMENT, RANGE_NUM_CONTAINER, RANGE_CONTAINER_ID,
       DBPARTITIONNUM FROM SYSIBMADM.SNAPTbsp_RANGE
ORDER BY DBPARTITIONNUM
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| TBSP_ID | TBSP_NAME | RANGE_NUMBER | RANGE_STRIPE_SET_NUMBER | ... |
|---------|--------------|--------------|-------------------------|-----|
| 0 | SYSCATSPACE | 0 | 0 | ... |
| 2 | USERSPACE1 | 0 | 0 | ... |
| 3 | SYSTOOLSPACE | 0 | 0 | ... |
| 2 | USERSPACE1 | 0 | 0 | ... |
| 2 | USERSPACE1 | 0 | 0 | ... |

5 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

| ... | RANGE_OFFSET | RANGE_MAX_PAGE | RANGE_MAX_EXTENT | ... |
|-----|--------------|----------------|------------------|-----|
| ... | 0 | 11515 | 2878 | ... |
| ... | 0 | 479 | 14 | ... |

| | | | |
|-----|---|-----|--------|
| ... | 0 | 251 | 62 ... |
| ... | 0 | 479 | 14 ... |
| ... | 0 | 479 | 14 ... |

Datos de salida de esta consulta (continuación).

| ... | RANGE_START_STRIPE | RANGE_END_STRIPE | RANGE_ADJUSTMENT | ... |
|-----|--------------------|------------------|------------------|-----|
| ... | 0 | 2878 | 0 | ... |
| ... | 0 | 14 | 0 | ... |
| ... | 0 | 62 | 0 | ... |
| ... | 0 | 14 | 0 | ... |
| ... | 0 | 14 | 0 | ... |

Datos de salida de esta consulta (continuación).

| ... | RANGE_NUM_CONTAINER | RANGE_CONTAINER_ID | DBPARTITIONNUM |
|-----|---------------------|--------------------|----------------|
| ... | 1 | 0 | 0 |
| ... | 1 | 0 | 0 |
| ... | 1 | 0 | 0 |
| ... | 1 | 0 | 1 |
| ... | 1 | 0 | 2 |

Función de tabla SNAP_GET_TBSP_RANGE

La función de tabla SNAP_GET_TAB_RANGE devuelve la misma información que la vista administrativa SNAPTAB_RANGE, pero le permite recuperar la información para una base de datos específica de una partición de base de datos específica, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_TBSP_RANGE, utilizada con las funciones de tabla SNAP_GET_TBSP_V91, SNAP_GET_TBSP_PART_V91, SNAP_GET_TBSP QUIESCER y SNAP_GET_CONTAINER_V91, proporciona información que es equivalente a la información que proporciona el mandato CLP GET SNAPSHOT FOR TABLESPACES ON alias de base de datos.

Consulte la Tabla 162 en la página 658 para ver una lista completa de la información que se puede devolver.

Sintaxis

```

>>> SNAP_GET_TBSP_RANGE ( ( nombrebd [ , númparticiónbd ] ) )

```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique NULL o una serie vacía para tomar la instantánea de la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número

válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_TBSP_RANGE tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla SNAP_GET_TBSP_RANGE
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMANT
- SYSADM

Ejemplos

Seleccionar información acerca del rango de espacio de tablas para el espacio de tablas con *tbasp_id* = 2 de la partición de base de datos conectada actualmente.

```
SELECT TBSP_ID, SUBSTR(TBSP_NAME, 1, 15) AS TBSP_NAME, RANGE_NUMBER,
       RANGE_STRIPE_SET_NUMBER, RANGE_OFFSET, RANGE_MAX_PAGE, RANGE_MAX_EXTENT,
       RANGE_START_STRIPE, RANGE_END_STRIPE, RANGE_ADJUSTMENT,
       RANGE_NUM_CONTAINER, RANGE_CONTAINER_ID
FROM TABLE(SNAP_GET_TBSP_RANGE('',-1)) AS T WHERE TBSP_ID = 2
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
TBSP_ID    TBSP_NAME          RANGE_NUMBER    ...
-----
2 USERSPACE1                0 ...
```

1 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```
... RANGE_STRIPE_SET_NUMBER RANGE_OFFSET    RANGE_MAX_PAGE    ...
... -----
...                      0                0                3967 ...
```

Datos de salida de esta consulta (continuación).

```
... RANGE_MAX_EXTENT    RANGE_START_STRIPE    RANGE_END_STRIPE    ...
... -----
...                123                0                123 ...
```

Datos de salida de esta consulta (continuación).

```

... RANGE_ADJUSTMENT    RANGE_NUM_CONTAINER RANGE_CONTAINER_ID
... -----
...                      0                      1                      0

```

Información devuelta

Tabla 192. Información devuelta por la vista administrativa SNAPTBSP_RANGE y la función de tabla SNAP_GET_TBSP_RANGE

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| TBSP_ID | BIGINT | tablespace_id - Identificación de espacio de tablas |
| TBSP_NAME | VARCHAR(128) | tablespace_name - Nombre de espacio de tablas |
| RANGE_NUMBER | BIGINT | range_number - Número de rango |
| RANGE_STRIPE_SET_NUMBER | BIGINT | range_stripe_set_number - Número de conjunto de bandas |
| RANGE_OFFSET | BIGINT | range_offset - Desplazamiento de rango |
| RANGE_MAX_PAGE | BIGINT | range_max_page_number - Página máxima en rango |
| RANGE_MAX_EXTENT | BIGINT | range_max_extent - Extensión máxima en rango |
| RANGE_START_STRIPE | BIGINT | range_start_stripe - Banda inicial |
| RANGE_END_STRIPE | BIGINT | range_end_stripe - Banda final |
| RANGE_ADJUSTMENT | BIGINT | range_adjustment - Ajuste de rango |
| RANGE_NUM_CONTAINER | BIGINT | range_num_containers - Número de contenedores en rango |
| RANGE_CONTAINER_ID | BIGINT | range_container_id - Contenedor de rango |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa SNAPUTIL y función de tabla SNAP_GET_UTIL - Recuperar información de instantánea de grupo de datos lógicos utility_info

La vista administrativa SNAPUTIL y la función de tabla SNAP_GET_UTIL devuelven información de instantáneas acerca de los programas de utilidad desde el grupo de datos lógicos utility_info.

SNAPUTIL, vista administrativa

La vista administrativa SNAPUTIL, que se utiliza junto con la vista administrativa SNAPUTIL_PROGRESS, proporciona la misma información que proporciona el mandato CLP LIST UTILITIES SHOW DETAIL.

El esquema es SYSIBMADM.

Consulte la Tabla 163 en la página 661 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPUTIL
- Privilegio CONTROL en la vista administrativa SNAPUTIL
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE para la función de tabla SNAP_GET_UTIL
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar una lista de programas de utilidad, y los estados de éstos, de todas las particiones de base de datos para todas las bases de datos activas en la instancia que contiene la base de datos conectada.

```
SELECT UTILITY_TYPE, UTILITY_PRIORITY, SUBSTR(UTILITY_DESCRIPTION, 1, 72)
       AS UTILITY_DESCRIPTION, SUBSTR(UTILITY_DBNAME, 1, 17) AS
       UTILITY_DBNAME, UTILITY_STATE, UTILITY_INVOKER_TYPE, DBPARTITIONNUM
FROM SYSIBMADM.SNAPUTIL ORDER BY DBPARTITIONNUM
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
UTILITY_TYPE    UTILITY_PRIORITY ...
-----
LOAD            - ...
LOAD            - ...
LOAD            - ...
```

3 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```
... UTILITY_DESCRIPTION ...
... -----
... ONLINE LOAD DEL AUTOMATIC INDEXING INSERT COPY NO TEST .LOADTEST ...
... ONLINE LOAD DEL AUTOMATIC INDEXING INSERT COPY NO TEST .LOADTEST ...
... ONLINE LOAD DEL AUTOMATIC INDEXING INSERT COPY NO TEST .LOADTEST ...
```


Datos de salida de esta consulta (continuación).

| UTILITY_DBNAME | UTILITY_STATE | UTILITY_INVOKER_TYPE | DBPARTITIONNUM |
|----------------|---------------|----------------------|----------------|
| SAMPLE | EXECUTE | USER | 0 |
| SAMPLE | EXECUTE | USER | 1 |
| SAMPLE | EXECUTE | USER | 2 |

SNAP_GET_UTIL, función de tabla

La función de tabla SNAP_GET_UTIL devuelve la misma información que la vista administrativa SNAPUTIL, pero le permite recuperar la información para una partición de base de datos específica, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_UTIL, que se utiliza junto con la función de tabla SNAP_GET_UTIL_PROGRESS, proporciona la misma información que el mandato CLP LIST UTILITIES SHOW DETAIL.

Consulte la Tabla 163 en la página 661 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
SNAP_GET_UTIL(núm-partición-bd)
```

El esquema es SYSPROC.

Parámetro de la función de tabla

núm-partición-bd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si no se utiliza esta opción de entrada, se devolverán datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *núm-partición-bd* se establece en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_UTIL tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE para la función de tabla SNAP_GET_UTIL
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar una lista de los ID de programa de utilidad, con su tipo y estado, para la partición de base de datos conectada actualmente en la base de datos SAMPLE.

```
SELECT UTILITY_ID, UTILITY_TYPE, STATE
FROM TABLE(SNAP_GET_UTIL(-1)) AS T WHERE UTILITY_DBNAME='SAMPLE'
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| UTILITY_ID | UTILITY_TYPE | STATE |
|------------|--------------|---------|
| 1 | BACKUP | EXECUTE |

1 registro(s) seleccionado(s).

Información devuelta

Tabla 193. Información que devuelve la vista administrativa SNAPUTIL y la función de tabla SNAP_GET_UTIL

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| UTILITY_ID | INTEGER | utility_id - ID de programa de utilidad. Exclusivo de una partición de base de datos. |
| UTILITY_TYPE | VARCHAR(26) | utility_type - Tipo de programa de utilidad. Esta interfaz devuelve un identificador de texto basándose en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • ASYNC_INDEX_CLEANUP • BACKUP • CRASH_RECOVERY • LOAD • REBALANCE • REDISTRIBUTE • REORG • RESTART_RECREATE_INDEX • RESTORE • ROLLFORWARD_RECOVERY • RUNSTATS |
| UTILITY_PRIORITY | INTEGER | utility_priority - Prioridad del programa de utilidad. Prioridad si el programa de utilidad da soporte a la disminución; de lo contrario, nulo. |

Tabla 193. Información que devuelve la vista administrativa SNAPUTIL y la función de tabla SNAP_GET_UTIL (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------------|---------------|---|
| UTILITY_DESCRIPTION | VARCHAR(2048) | utility_description - Descripción del programa de utilidad. Puede ser nulo. |
| UTILITY_DBNAME | VARCHAR(128) | utility_dbname - Base de datos en la que trabaja el programa de utilidad |
| UTILITY_START_TIME | TIMESTAMP | utility_start_time - Hora de inicio del programa de utilidad |
| UTILITY_STATE | VARCHAR(10) | utility_state - Estado del programa de utilidad. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • ERROR • EXECUTE • WAIT |
| UTILITY_INVOKER_TYPE | VARCHAR(10) | utility_invoker_type - Tipo de invocador del programa de utilidad. Esta interfaz devuelve un identificador de texto basándose en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • AUTO • USER |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |
| PROGRESS_LIST_ATTR | VARCHAR(10) | progress_list_attr - Atributos de la lista de progresos actual |
| PROGRESS_LIST_CUR_SEQ_NUM | INTEGER | progress_list_current_seq_num - Número de secuencia de lista de progresos actual |

Vista administrativa SNAPUTIL_PROGRESS y función de tabla SNAP_GET_UTIL_PROGRESS - Recuperar información de instantánea de grupo de datos lógicos de progreso

La vista administrativa SNAPUTIL_PROGRESS y la función de tabla SNAP_GET_UTIL_PROGRESS devuelven información de instantáneas acerca del progreso del programa de utilidad, en concreto el grupo de datos lógicos de progreso.

SNAPUTIL_PROGRESS, vista administrativa

La vista administrativa SNAPUTIL_PROGRESS, que se utiliza junto con la vista administrativa SNAPUTIL, proporciona la misma información que el mandato CLP LIST UTILITIES SHOW DETAIL.

El esquema es SYSIBMADM.

Consulte la Tabla 164 en la página 665 para ver una lista completa de la información que se puede devolver.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa SNAPUTIL_PROGRESS
- Privilegio CONTROL en la vista administrativa SNAPUTIL_PROGRESS
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE para la función de tabla SNAP_GET_UTIL_PROGRESS
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMMAINT
- SYSADM

Ejemplo

Recuperar información detallada acerca de las unidades de progreso totales y completadas por ID de programa de utilidad.

```
SELECT SELECT UTILITY_ID, PROGRESS_TOTAL_UNITS, PROGRESS_COMPLETED_UNITS,  
        DBPARTITIONNUM FROM SYSIBMADM.SNAPUTIL_PROGRESS
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
UTILITY_ID PROGRESS_TOTAL_UNITS PROGRESS_COMPLETED_UNITS DBPARTITIONNU  
-----  
          7                10                5                0  
          9                10                5                1
```

1 registro(s) seleccionado(s).

SNAP_GET_UTIL_PROGRESS, función de tabla

La función de tabla SNAP_GET_UTIL_PROGRESS devuelve la misma información que la vista administrativa SNAPUTIL_PROGRESS, pero le permite recuperar la información para una base de datos específica de una partición de base de datos específica, un agregado de todas las particiones de base de datos o todas las particiones de base de datos.

La función de tabla SNAP_GET_UTIL_PROGRESS, que se utiliza junto con la función de tabla SNAP_GET_UTIL, proporciona la misma información que el mandato CLP LIST UTILITIES SHOW DETAIL.

Consulte la Tabla 164 en la página 665 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
▶▶ SNAP_GET_UTIL_PROGRESS ( [número-partición-bd] ) ▶▶▶▶
```

El esquema es SYSPROC.

Parámetro de la función de tabla

número-partición-bd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si no se utiliza esta opción de entrada, se devolverán datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *número-partición-bd* se establece en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_UTIL_PROGRESS tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio EXECUTE para la función de tabla SNAP_GET_UTIL_PROGRESS
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMOINT
- SYSADM

Ejemplo

Recuperar información detallada acerca del progreso de los programas de utilidad de la partición conectada actualmente.

```
SELECT UTILITY_ID, PROGRESS_TOTAL_UNITS, PROGRESS_COMPLETED_UNITS,  
       DBPARTITIONNUM FROM TABLE(SNAP_GET_UTIL_PROGRESS(-1)) as T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
UTILITY_ID PROGRESS_TOTAL_UNITS PROGRESS_COMPLETED_UNITS DBPARTITIONNUM  
-----  
          7                10                5                0
```

1 registro(s) seleccionado(s).

Información devuelta

Tabla 194. Información devuelta por la vista administrativa SNAPUTIL_PROGRESS y la función de tabla SNAP_GET_UTIL_PROGRESS

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| UTILITY_ID | INTEGER | utility_id - ID de programa de utilidad. Exclusivo de una partición de base de datos. |
| PROGRESS_SEQ_NUM | INTEGER | progress_seq_num - Número de secuencia de progreso. Si es serie, el número de la fase. Si es simultáneo, puede ser NULL. |
| UTILITY_STATE | VARCHAR(16) | utility_state - Estado del programa de utilidad. Esta interfaz devuelve un identificador de texto basándose en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • ERROR • EXECUTE • WAIT |
| PROGRESS_DESCRIPTION | VARCHAR(2048) | progress_description - Descripción del progreso |
| PROGRESS_START_TIME | TIMESTAMP | progress_start_time - Hora de inicio del progreso. Hora de inicio si la fase se ha iniciado; de lo contrario, NULL. |
| PROGRESS_WORK_METRIC | VARCHAR(16) | progress_work_metric - Métrica de trabajo en curso. Esta interfaz devuelve un identificador de texto basándose en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • NOT_SUPPORT • BYTES • EXTENTS • INDEXES • PAGES • ROWS • TABLES |
| PROGRESS_TOTAL_UNITS | BIGINT | progress_total_units - Total de unidades de trabajo en curso |
| PROGRESS_COMPLETED_UNITS | BIGINT | progress_completed_units - Unidades de trabajo en curso completadas |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Procedimiento SNAP_WRITE_FILE

El procedimiento SNAP_WRITE_FILE graba datos de instantánea del sistema en un archivo ubicado en el subdirectorio tmp del directorio de la instancia.

Sintaxis

```
▶▶—SNAP_WRITE_FILE—(—tipo-petición—,—nombre-bd—,—númparticiónbd—)————▶▶
```

El esquema es SYSPROC.

Parámetros del procedimiento

tipo-petición

Argumento de entrada de tipo VARCHAR (32) que especifica un tipo de petición de instantánea válido. Los tipos de petición posibles son identificadores de texto basados en las sentencias define de sqlmon.h y son uno de los siguientes:

- APPL_ALL
- BUFFERPOOLS_ALL
- DB2
- DBASE_ALL
- DBASE_LOCKS
- DBASE_TABLES
- DBASE_TABLESPACES
- DYNAMIC_SQL

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada en el momento en que se llama a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique NULL o una serie vacía para tomar la instantánea de la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica un valor nulo, se establece -1 de forma implícita.

Autorización

Para ejecutar el procedimiento, el usuario debe tener autorización SYSADM, SYSCTRL, SYSMANT o SYSMON. Los usuarios que no tengan autorización SYSADM, SYSCTRL, SYSMANT o SYSMON podrán leer la instantánea guardada pasando valores nulos como entrada a funciones de tabla de la instantánea.

Ejemplo

Tomar una instantánea de la información del gestor de bases de datos especificando un tipo de petición de 'DB2' (que corresponde a SQLMA_DB2) y adoptando los valores por omisión de la base de datos conectada actualmente y la partición de base de datos actual.

```
CALL SYSPROC.SNAP_WRITE_FILE ('DB2', '', -1)
```

Esto dará como resultado que se graben datos de instantánea en el directorio temporal de instancia, que es sql1ib/tmp/SQLMA_DB2.dat en los sistemas operativos UNIX y sql1ib\DB2\tmp\SQLMA_DB2.dat en un sistema operativo Windows.

Notas de uso

Si se proporciona un parámetro de entrada no reconocido, se devuelve el error siguiente: SQL2032N El parámetro "REQUEST_TYPE" no es válido.

Vista administrativa TBSP_UTILIZATION - Recuperar información de configuración y utilización de espacio de tablas

La vista administrativa TBSP_UTILIZATION devuelve información de utilización y configuración de espacio de tablas. La vista es una interfaz de SQL para el mandato CLP LIST TABLESPACES. Su información se basa en las vistas administrativas SNAPTbsp, SNAPTbsp_PART y en la vista de catálogo TABLESPACES.

El esquema es SYSIBMADM.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa TBSP_UTILIZATION
- Privilegio CONTROL en la vista administrativa TBSP_UTILIZATION
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMANT
- SYSADM

Ejemplo

Recuperar el mismo informe que el mandato LIST TABLESPACE de una única base de datos particionada.

```
SELECT TBSP_ID, SUBSTR(TBSP_NAME,1,20) as TBSP_NAME, TBSP_TYPE,  
       TBSP_CONTENT_TYPE, TBSP_STATE FROM SYSIBMADM.TBSP_UTILIZATION
```

El siguiente ejemplo muestra los datos de salida de la consulta.

| TBSP_ID | TBSP_NAME | TBSP_TYPE | ... |
|---------|-------------|-----------|-----|
| 0 | SYSCATSPACE | SMS | ... |


```

1 TEMPSPACE1      SMS      ...
2 USERSPACE1     SMS      ...
3 SYSTOOLSPACE   SMS      ...
4 SYSTOOLSTMPSPACE SMS     ...

```

Datos de salida de esta consulta (continuación).

```

... TBSP_CONTENT_TYPE TBSP_STATE
... -----
... ANY                NORMAL
... SYSTEMP           NORMAL
... ANY                NORMAL
... ANY                NORMAL
... USRTEMP           NORMAL

```

Información devuelta

Tabla 195. Información devuelta por la vista administrativa TBSP_UTILIZATION

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| TBSP_ID | BIGINT | tablespace_id - Identificación de espacio de tablas |
| TBSP_NAME | VARCHAR(128) | tablespace_name - Nombre de espacio de tablas |
| TBSP_TYPE | VARCHAR(10) | tablespace_type - Tipo de espacio de tablas. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlutil.h y es uno de los siguientes: <ul style="list-style-type: none"> • DMS • SMS |
| TBSP_CONTENT_TYPE | VARCHAR(10) | tablespace_content_type - Tipo de contenido de espacio de tablas. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlutil.h y es uno de los siguientes: <ul style="list-style-type: none"> • ANY • LONG • SYSTEMP • USRTEMP |
| TBSP_CREATE_TIME | TIMESTAMP | Hora de creación del espacio de tablas. |

Tabla 195. Información devuelta por la vista administrativa
TBSP_UTILIZATION (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------|---------------|---|
| TBSP_STATE | VARCHAR(256) | <p>tablespace_state - Estado de espacio de tablas. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlutil.h y es una combinación de los siguientes, separados mediante un signo '+':</p> <ul style="list-style-type: none"> • BACKUP_IN_PROGRESS • BACKUP_PENDING • DELETE_PENDING • DISABLE_PENDING • DROP_PENDING • LOAD_IN_PROGRESS • LOAD_PENDING • NORMAL • OFFLINE • PSTAT_CREATION • PSTAT_DELETION • QUIESCED_EXCLUSIVE • QUIESCED_SHARE • QUIESCED_UPDATE • REBAL_IN_PROGRESS • REORG_IN_PROGRESS • RESTORE_IN_PROGRESS • RESTORE_PENDING • • ROLLFORWARD_IN_PROGRESS • ROLLFORWARD_PENDING • STORDEF_ALLOWED • STORDEF_CHANGED • STORDEF_FINAL_VERSION • STORDEF_PENDING • SUSPEND_WRITE |
| TBSP_TOTAL_SIZE_KB | BIGINT | El tamaño total del espacio de tablas en KB, calculado como páginas_total*tamaño_página/1024. |
| TBSP_USABLE_SIZE_KB | BIGINT | El tamaño total utilizable del espacio de tablas en KB, calculado como págs_utilizables*tamaño_pág/1024. |
| TBSP_USED_SIZE_KB | BIGINT | El tamaño total utilizado del espacio de tablas en KB, calculado como págs_utilizables*tamaño_pág/1024. |

Tabla 195. Información devuelta por la vista administrativa
TBSP_UTILIZATION (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|----------------------------|---------------|---|
| TBSP_FREE_SIZE_KB | BIGINT | El tamaño total disponible del espacio de tablas en KB, calculado como $\text{páginas_libres} * \text{tamaño_página} / 1024$. |
| TBSP_UTILIZATION_PERCENT | BIGINT | La utilización del espacio de tablas como porcentaje. Calculado como $(\text{páginas_utilizadas} / \text{páginas_utilizables}) * 100$, si existen páginas_utilizables. De lo contrario, se visualizará -1. |
| TBSP_TOTAL_PAGES | BIGINT | tablespace_total_pages - Total de páginas en espacio de tablas |
| TBSP_USABLE_PAGES | BIGINT | tablespace_usable_pages - Páginas utilizables de espacio de tablas |
| TBSP_USED_PAGES | BIGINT | tablespace_used_pages - Páginas utilizadas en espacio de tablas |
| TBSP_FREE_PAGES | BIGINT | tablespace_free_pages - Páginas libres en espacio de tablas |
| TBSP_PAGE_TOP | BIGINT | tablespace_page_top - Nivel límite de espacio de tablas |
| TBSP_PAGE_SIZE | INTEGER | tablespace_page_size - Tamaño de página de espacio de tablas |
| TBSP_EXTENT_SIZE | INTEGER | tablespace_extent_size - Tamaño de extensión de espacio de tablas |
| TBSP_PREFETCH_SIZE | BIGINT | tablespace_prefetch_size - Tamaño de captación previa de espacio de tablas |
| TBSP_MAX_SIZE | BIGINT | tablespace_max_size - Tamaño máximo de espacio de tablas |
| TBSP_INCREASE_SIZE | BIGINT | tablespace_increase_size - Aumento de tamaño en bytes |
| TBSP_INCREASE_SIZE_PERCENT | SMALLINT | tablespace_increase_size_percent - Aumento de tamaño en porcentaje |
| TBSP_LAST_RESIZE_TIME | TIMESTAMP | tablespace_last_resize_time - Hora del último redimensionamiento satisfactorio |
| TBSP_LAST_RESIZE_FAILED | SMALLINT | tablespace_last_resize_failed - Último intento de redimensionamiento fallido |
| TBSP_USING_AUTO_STORAGE | SMALLINT | tablespace_using_auto_storage - Espacio de tablas habilitado para el almacenamiento automático |
| TBSP_AUTO_RESIZE_ENABLED | SMALLINT | tablespace_auto_resize_enabled - Espacio de tablas habilitado para la modificación de tamaño automática |

Tabla 195. Información devuelta por la vista administrativa
TBSP_UTILIZATION (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------|---------------|---|
| DBPGNAME | VARCHAR(128) | Nombre del grupo de particiones de base de datos para el espacio de tablas. |
| TBSP_NUM_CONTAINERS | BIGINT | tablespace_num_containers - Número de contenedores de espacio de tablas |
| REMARKS | VARCHAR(254) | Comentario suministrado por el usuario. |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Vista administrativa TOP_DYNAMIC_SQL – Recuperar información acerca de las sentencias de SQL dinámico más frecuentes

La vista administrativa TOP_DYNAMIC_SQL devuelve las sentencias de SQL dinámico más frecuentes que pueden clasificarse por número de ejecuciones, tiempo de ejecución medio, número de clasificaciones o clasificaciones por sentencia. Estas son las consultas cuya correcta especificación debería satisfacerse.

El esquema es SYSIBMADM.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa TOP_DYNAMIC_SQL
- Privilegio CONTROL en la vista administrativa TOP_DYNAMIC_SQL
- Autorización DATAACCESS

Asimismo, para acceder a los datos del supervisor de instantáneas, también es necesaria una de las autorizaciones siguientes:

- SYSMON
- SYSCTRL
- SYSMAINT
- SYSADM

Ejemplo

Identificar las cinco consultas de SQL más frecuentes que se han ejecutado.

```
SELECT NUM_EXECUTIONS, AVERAGE_EXECUTION_TIME_S, STMT_SORTS,
       SORTS_PER_EXECUTION, SUBSTR(STMT_TEXT,1,60) AS STMT_TEXT
FROM SYSIBMADM.TOP_DYNAMIC_SQL
ORDER BY NUM_EXECUTIONS DESC FETCH FIRST 5 ROWS ONLY
```

El siguiente ejemplo muestra los datos de salida de la consulta.

```
NUM_EXECUTIONS      AVERAGE_EXECUTION_TIME_S STMT_SORTS      ...
-----
148                0                      0 ...
```

```

123          0          0 ...
  2          0          0 ...
  1          0          0 ...
  1          0          0 ...

```

5 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```

... SORTS_PER_EXECUTION ...
... ----- ...
...          0 ...
...          0 ...
...          0 ...
...          0 ...
...          0 ...

```

Datos de salida de esta consulta (continuación).

```

... STMT_TEXT
... -----
... SELECT A.ID, B.EMPNO, B.FIRSTNME, B.LASTNAME, A.DEPT FROM E
... SELECT A.EMPNO, A.FIRSTNME, A.LASTNAME, B.LOCATION, B.MGRNO
... SELECT A.EMPNO, A.FIRSTNME, A.LASTNAME, B.DEPTNAME FROM EMP
... SELECT ATM.SCHEMA, ATM.NAME, ATM.CREATE_TIME, ATM.LAST_WAIT,
... SELECT * FROM JESSICAE.EMP_RESUME

```

Información devuelta

Tabla 196. Información devuelta por la vista administrativa TOP_DYNAMIC_SQL

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | Indicación de fecha y hora para el informe. |
| NUM_EXECUTIONS | BIGINT | num_compilations - Compilaciones de sentencia |
| AVERAGE_EXECUTION_TIME_S | BIGINT | Tiempo de ejecución medio |
| STMT_SORTS | BIGINT | stmt_sorts - Clasificaciones de sentencias |
| SORTS_PER_EXECUTION | BIGINT | Número de clasificaciones por ejecución de sentencia. |
| STMT_TEXT | CLOB(2 M) | stmt_text - Texto de sentencia de SQL |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Capítulo 15. Rutinas de procedimientos de SQL

Procedimiento ALTER_ROUTINE_PACKAGE

►►—ALTER_ROUTINE_PACKAGE—(—*tipo*—,—*esquema*—,—*módulo*—,—*nombre*—,—*opciones*—)——►►

El esquema es SYSPROC.

Este procedimiento modifica los valores del paquete asociado con una rutina de SQL compilado o un activador compilado, sin necesidad de realizar revinculaciones. Es funcionalmente equivalente al mandato ALTER PACKAGE, excepto en que toma un nombre de objeto, en lugar de un nombre de paquete, como argumento. El procedimiento ALTER_ROUTINE_PACKAGE puede invocarse desde la línea de mandatos o se puede llamar desde una aplicación.

tipo

Argumento de entrada de tipo CHAR(2) que especifica el tipo de rutina o activador compilado mediante uno de los valores siguientes:

- 'P' para un procedimiento
- 'SP' para el nombre específico de un procedimiento
- 'F' para una función compilada
- 'SF' para un nombre específico de una función compilada
- 'T' para un activador compilado

esquema

Argumento de entrada opcional de tipo VARCHAR(128) que especifica el esquema de la rutina o activador. Si no se especifica un esquema, el valor por omisión será el valor del registro especial CURRENT SCHEMA. Este parámetro es sensible a las mayúsculas y minúsculas.

módulo

Argumento de entrada opcional de tipo VARCHAR(128) que especifica el esquema del módulo en el que reside la rutina. Este parámetro no se puede especificar para los activadores. Si no se especifica este parámetro, se omiten las rutinas de módulo. Este parámetro es sensible a las mayúsculas y minúsculas.

nombre

Argumento de entrada de tipo VARCHAR(128) que especifica el nombre de la rutina o activador. Este parámetro es sensible a las mayúsculas y minúsculas.

opciones

Argumento de entrada de tipo VARCHAR(1024) que especifica una lista de opciones soportadas por la sentencia ALTER PACKAGE. Debe proporcionarse al menos una cláusula ALTER PACKAGE dentro del parámetro *opciones*.

Ejemplos

Modificar el paquete subyacente para un procedimiento almacenado existente con el nombre de UPDATE_EMPLOYEE.

```
CALL SYSPROC.ALTER_ROUTINE_PACKAGE ('P', '', 'UPDATE_EMPLOYEE',  
'ACCESS PLAN REUSE YES OPTIMIZATION PROFILE AYYANG.INDEXHINTS')
```

Modificar el paquete para un activador compilado denominado MIN_SALARY, en el esquema DRICARD.

```
CALL SYSPROC.ALTER_ROUTINE_PACKAGE ('T','DRICARD','','MIN_SALARY',  
    'OPTIMIZATION PROFILE AYYANG.INDEXHINTS')
```

Modificar el paquete para una función compilada, utilizando un nombre en tres partes.

```
CALL SYSPROC.ALTER_ROUTINE_PACKAGE ('F','DRICARD','MODULE','FUNCTION','APREUSE YES')
```

RGET_ROUTINE_OPTS

►► GET_ROUTINE_OPTS (—) ◀◀

El esquema es SYSPROC.

La función GET_ROUTINE_OPTS devuelve un valor de serie de caracteres de las opciones que se deben utilizar para la creación de procedimientos SQL en la sesión actual.

El resultado de la función es un valor de serie de caracteres de longitud variable (VARCHAR) con un atributo de longitud de 1024.

Ejemplo:

Devolver las opciones a utilizar para la creación de procedimientos SQL como resultado de una consulta.

```
SELECT GET_ROUTINE_OPTS()  
FROM SYSIBM.SYSDUMMY1
```

GET_ROUTINE_SAR

►► GET_ROUTINE_SAR ◀◀
► (—blob_sar—, —tipo—, —serie-nombre-rutina— [,—ocultar-distintivo-cuerpo—]) ◀◀

El esquema es SYSFUN.

El procedimiento GET_ROUTINE_SAR recupera la información necesaria para instalar la misma rutina en otro servidor de bases de datos que se ejecute en el mismo nivel en el mismo sistema operativo. La información se recupera en una serie BLOB individual que representa un archivo archivador SQL.

Autorización

Los privilegios que posee el ID de autorización debe tener al menos una de las autorizaciones o privilegios siguientes:

- DBADM
- SQLADM
- DATAACCESS
- EXECUTE

blobsar

Argumento de salida de tipo BLOB(3M) que incluye el contenido del archivo SAR de la rutina.

tipo

Argumento de entrada de tipo CHAR(2) que especifica el tipo de rutina, mediante uno de los valores siguientes:

- 'P' para un procedimiento
- 'SP' para el nombre específico de un procedimiento

serie-nombre-rutina

Argumento de entrada de tipo VARCHAR(257) que especifica un nombre calificado de la rutina. Si no se especifica ningún nombre de esquema, el valor por omisión es CURRENT SCHEMA cuando se procesa la rutina. La *serie-nombre-rutina* no puede incluir comillas dobles (").

ocultar-distintivo-cuerpo

Un argumento de entrada de tipo INTEGER que especifica (utilizando uno de los valores siguientes) si debe o no debe ocultarse el cuerpo de la rutina cuando se extraer texto de la rutina de los catálogos. Los valores válidos son:

- 0 Dejar el texto de la rutina intacto. Es el valor por omisión.
- 1 Sustituir el cuerpo de la rutina por un cuerpo vacío cuando el texto de la rutina se extrae de los catálogos.

El nombre calificado de la rutina se utiliza para determinar qué rutina se debe recuperar. La rutina que se encuentre debe ser una rutina SQL. Si no se utiliza un nombre específico, se puede dar lugar a más de una rutina y se producirá un error (SQLSTATE 42725). Si ocurre esto, se debe utilizar el nombre específico de la rutina que se desee.

El archivo SAR debe incluir un archivo de vinculación, que puede no estar disponible en el servidor. Si el archivo de vinculación no se puede encontrar y almacenar en el archivo SAR, se producirá un error (SQLSTATE 55045).

PUT_ROUTINE_SAR

El procedimiento PUT_ROUTINE_SAR pasa el archivo necesario para crear una rutina SQL en el servidor y, a continuación, define la rutina.

Autorización

Los privilegios que posee el ID de autorización debe tener al menos una de las autorizaciones o privilegios siguientes:

- DBADM
- SQLADM
- DATAACCESS
- EXECUTE

```
▶▶—PUT_ROUTINE_SAR—▶▶
▶—(—blobsar—, —propietario-nuevo—, —distintivo-registro-uso—)—▶▶
```

El esquema es SYSFUN.

blobsar

Argumento de entrada de tipo BLOB(3M) que incluye el contenido del archivo SAR de la rutina.

propietario-nuevo

Argumento de entrada de tipo VARCHAR(128) que contiene un nombre-autorización utilizado para la comprobación de autorización de la rutina. El *propietario-nuevo* debe tener los privilegios necesarios para que se defina la rutina. Si no se especifica *propietario-nuevo*, se utilizará el nombre-autorización del definidor de rutina original.

distintivo-registro-uso

Argumento de entrada de tipo INTEGER que indica si los registros especiales CURRENT SCHEMA y CURRENT PATH se utilizan para definir la rutina. Si no se utilizan los registros especiales, los valores para el esquema por omisión y la vía de acceso de SQL son los valores utilizados cuando se ha definido originalmente la rutina. Valores posibles para *distintivo-registro-uso*:

- 0 No utilizar los registros especiales del entorno actual
- 1 Utilizar los registros especiales CURRENT SCHEMA y CURRENT PATH.

Si el valor es 1, se utiliza CURRENT SCHEMA para los nombres de objeto no calificados en la definición de rutina (incluido el nombre de la rutina) y CURRENT PATH se utiliza para resolver rutinas no calificadas y tipos de datos en la definición de rutina. Si no se especifica *distintivo-registro-uso*, el comportamiento es el mismo que si se hubiera especificado un valor de 0.

La información de identificación contenida en *sarblob* se comprueba para confirmar que las entradas son apropiadas para el entorno porque, de lo contrario, se produce un error (SQLSTATE 55046). Entonces el procedimiento PUT_ROUTINE_SAR utiliza el contenido de *sarblob* para definir la rutina en el servidor.

El contenido del argumento *sarblob* se extrae en los archivos independientes que forman el archivo archivador SQL. La biblioteca compartida y los archivos de vinculación se graban en archivos de un directorio temporal. El entorno se establece de forma tal que el proceso de sentencia de definición de rutina esté informado de que la compilación y el enlace no son necesarios y de que la ubicación de la biblioteca compartida y los archivos de vinculación está disponible. Entonces se utiliza el contenido del archivo DDL para ejecutar dinámicamente la sentencia de definición de rutina.

No se puede instalar simultáneamente más de un procedimiento bajo un esquema determinado.

El proceso de esta sentencia puede producir los mismos errores que la ejecución de la sentencia de definición de rutina utilizando otras interfaces. Durante el proceso de definición de rutina, se detecta la presencia de la biblioteca compartida y de los archivos de vinculación y se saltan los pasos de precompilación, compilación y enlace. El archivo de vinculación se utiliza durante el proceso de vinculación y el contenido de ambos archivos se copia en el directorio usual para una rutina SQL.

Si no se puede ejecutar satisfactoriamente una operación GET ROUTINE o PUT ROUTINE (o su procedimiento correspondiente), se devolverá siempre un error (SQLSTATE 38000), junto con texto de diagnóstico que proporciona información

acerca de la causa de la anomalía. Por ejemplo, si el nombre de procedimiento proporcionado a GET ROUTINE no identifica un procedimiento SQL, se devolverá el texto de diagnóstico "-204, 42704", donde "-204" y "42704" son SQLCODE y SQLSTATE, respectivamente, que identifican la causa del problema. En este ejemplo, SQLCODE y SQLSTATE indican que el nombre de procedimiento proporcionado en el mandato GET ROUTINE no está definido.

Procedimiento REBIND_ROUTINE_PACKAGE - volver a vincular un paquete

El procedimiento REBIND_ROUTINE_PACKAGE vuelve a vincular el paquete asociado con un procedimiento de SQL, una rutina, una función compilada o un activador. Es funcionalmente equivalente al mandato REBIND, excepto en que toma el nombre de un procedimiento, en lugar del nombre de un paquete, como argumento. El procedimiento REBIND_ROUTINE_PACKAGE puede invocarse desde la línea de mandatos o se puede llamar desde una aplicación.

Sintaxis

Existen dos métodos igualmente válidos para invocar REBIND_ROUTINE_PACKAGE. La única diferencia entre las dos invocaciones es el método de especificación del nombre de rutina. En el primer caso, la variable *serie-nombre-rutina* está formada por nombres de identificador separados por puntos. En el segundo método, la rutina se identifica mediante valores independientes para cada valor de *esquema*, *módulo* y *nombre*.

Método 1:

```
►► REBIND_ROUTINE_PACKAGE (—tipo—, —serie-nombre-rutina—, —opciones—) ►►
```

Método 2:

```
►► REBIND_ROUTINE_PACKAGE (—tipo—, —————►►
► esquema—, —módulo—, —nombre—, —opción—) ►►
```

El esquema es SYSPROC.

Parámetros del procedimiento

tipo

Argumento de entrada de tipo CHAR(2) que especifica el tipo de rutina o activador compilado mediante uno de los valores siguientes:

- 'P' para un procedimiento
- 'SP' para el nombre específico de un procedimiento
- 'F' para una función compilada
- 'SF' para un nombre específico de una función compilada
- 'T' para un activador compilado

serie-nombre-rutina (método 1 solamente)

Argumento de entrada de tipo VARCHAR(386) que especifica el nombre de la rutina o activador. Los nombres de los activadores constan de dos partes separadas por un punto y tienen el formato *esquema.activador* donde el

esquema es opcional. Los nombres de las rutinas están formados por tres partes separadas por puntos y tienen el formato esquema.módulo.rutina donde el esquema y el módulo son opcionales. Si no se especifica el esquema, el valor por omisión será el valor del registro especial CURRENT SCHEMA. Si se especifica un nombre de dos partes, la primera parte se interpreta inicialmente como el nombre del esquema; si la rutina no se encuentra bajo ese esquema, la primera parte se interpreta como el nombre del módulo y se realiza un intento de encontrar la rutina en un módulo con ese nombre bajo CURRENT SCHEMA. Los nombres de esquema, módulo u objeto no pueden incluir comillas dobles (") o puntos (.).

esquema (**método 2 únicamente**)

Argumento de entrada opcional de tipo VARCHAR(128) que especifica el esquema de la rutina o activador. Si no se especifica un esquema, el valor por omisión será el valor del registro especial CURRENT SCHEMA. Este parámetro es sensible a las mayúsculas y minúsculas.

módulo (**método 2 únicamente**)

Argumento de entrada opcional de tipo VARCHAR(128) que especifica el nombre del módulo en el que reside la rutina. No especifique este parámetro para los activadores. Si no se especifica este parámetro, se omiten las rutinas de módulo. Este parámetro es sensible a las mayúsculas y minúsculas.

nombre (**método 2 únicamente**)

Argumento de entrada de tipo VARCHAR (128) que especifica el nombre de la rutina o activador. Este parámetro es sensible a las mayúsculas y minúsculas.

opciones

Argumento de entrada opcional del tipo VARCHAR(1024) que especifica cualquier lista de opciones de revinculación que siguen a la sintaxis del mandato REBIND. También se acepta un solo valor "ANY" o "CONSERVATIVE" por cuestiones de compatibilidad y se interpreta como el valor para la opción de revinculación RESOLVE.

El nombre calificado de la rutina se utiliza para determinar qué rutina se debe recuperar. La rutina que se encuentre debe ser una rutina SQL; de lo contrario, se devuelve un error (SQLSTATE 428F7). Si no se utiliza un nombre específico, puede encontrarse más de una rutina y se devuelve un error (SQLSTATE 42725). Si ocurre esto, se debe utilizar el nombre específico de la rutina que se desee.

Ejemplos

Ejemplo 1: Volver a vincular el paquete de rutina UPDATE_EMPLOYEE empleando las opciones RESOLVE, REOPT y APREUSE.

```
Método 1:  
CALL SYSPROC.REBIND_ROUTINE_PACKAGE (  
    'P', 'UPDATE_EMPLOYEE', 'RESOLVE ANY REOPT ONCE APREUSE YES')
```

```
Método 2:  
CALL SYSPROC.REBIND_ROUTINE_PACKAGE (  
    'P', '', '', 'UPDATE_EMPLOYEE', 'RESOLVE ANY REOPT ONCE APREUSE YES')
```

Ejemplo 2: Volver a vincular el paquete de rutina UPDATE_EMPLOYEE sin opciones.

```
Método 1:  
CALL SYSPROC.REBIND_ROUTINE_PACKAGE (  
    'P', 'UPDATE_EMPLOYEE', '')
```

```
Método 2:  
CALL SYSPROC.REBIND_ROUTINE_PACKAGE (  
    'P', '', '', 'UPDATE_EMPLOYEE', '')
```

Ejemplo 3: Volver a vincular el paquete de un activador compilado.

```
Método 1:  
CALL SYSPROC.REBIND_ROUTINE_PACKAGE (  
    'T', 'DRICARD.MIN_SALARY', 'REOPT ALWAYS')
```

```
Método 2:  
CALL SYSPROC.REBIND_ROUTINE_PACKAGE (  
    'T', 'DRICARD', '', 'MIN_SALARY', 'REOPT ALWAYS')
```

Ejemplo 4: Volver a vincular el paquete de una función compilada utilizando un nombre en tres partes.

```
Método 1  
CALL SYSPROC.REBIND_ROUTINE_PACKAGE (  
    'F', 'DRICARD.MODULE.FUNCTION', 'REOPT ALWAYS')
```

```
Método 2  
CALL SYSPROC.REBIND_ROUTINE_PACKAGE (  
    'F', 'DRICARD', 'MODULE', 'FUNCTION', 'REOPT ALWAYS')
```

RSET_ROUTINE_OPTS

►►—SET_ROUTINE_OPTS—(—*expresión-carácter*—)—————►►

El esquema es SYSPROC.

El procedimiento SET_ROUTINE_OPTS establece las opciones que se deben utilizar para la creación de procedimientos SQL en la sesión actual. Este valor altera temporalmente el valor de instancia especificado en la variable de registro DB2_SQLROUTINE_PREPOPTS.

expresión-caracteres

Argumento de entrada de tipo VARCHAR(1024) que especifica el valor de las opciones para la sesión actual.

Las opciones especificadas son válidas mientras dura la sesión. Si se especifica el valor nulo como argumento, el valor de la variable de registro DB2_SQLROUTINE_PREPOPTS se restaura como el valor por omisión de las opciones para la sesión actual. Si desea obtener la lista de las opciones permitidas, consulte la descripción de la variable de registro DB2_SQLROUTINE_PREPOPTS en el apartado “Variables del compilador de consultas”.

Capítulo 16. Rutinas de redistribución paso a paso

Procedimiento ANALYZE_LOG_SPACE - Recuperar información de análisis de espacio de anotaciones

El procedimiento ANALYZE_LOG_SPACE devuelve los resultados del análisis del espacio de anotaciones cronológicas para cada una de las particiones de base de datos del grupo de particiones de base de datos indicado.

Sintaxis

```
►►—ANALYZE_LOG_SPACE—(—inDBPGroup—,—inMainTbSchema—,—inMainTable—,—  
►—analysisType—,—inStmgTime—,—addDropOption—,—addDropList—,—pNumber—,—  
►—pWeight—)—————►►
```

El esquema es SYSPROC.

Parámetros del procedimiento

inDBPGroup

Argumento de entrada de tipo VARCHAR (128) que especifica el nombre del grupo de particiones de base de datos.

inMainTbSchema

Argumento de entrada de tipo VARCHAR (128) que especifica el esquema de la tabla principal

inMainTable

Argumento de entrada de tipo VARCHAR (128) que especifica la tabla principal del grupo de particiones de base de datos y que, por lo general, se trata de la tabla más grande del grupo de particiones.

analysisType

Argumento de entrada de tipo SMALLINT que especifica un indicador para el tipo de análisis:

- SWRD_USE_STMG_TABLE (1): indica que la información de las tablas de gestión de almacenamiento se utiliza para saber el número de filas de tabla por partición de base de datos. Este tipo sólo debe utilizarse si las tablas de gestión de almacenamiento están instaladas y se ha tomado como mínimo una instantánea del grupo de particiones de base de datos que se va a redistribuir.
- SWRD_USE_REALTIME_ANALYSIS (2): indica que se utiliza una consulta SELECT para saber el número de filas por partición de base de datos.

inStmgTime

Argumento de entrada de tipo VARCHAR (26) que especifica la indicación de fecha y hora del registro de gestión de almacenamiento. Este parámetro no se tendrá en cuenta cuando *analysisType* esté establecido en SWRD_USE_REALTIME_ANALYSIS.

addDropOption

Argumento de entrada de tipo CHAR (1) que especifica que se están añadiendo o eliminando particiones de base de datos:

- 'A': Se están añadiendo particiones de base de datos.
- 'D': Se están descartando particiones de base de datos.
- 'N': No se están añadiendo ni eliminando.

addDropList

Argumento de entrada de tipo VARCHAR (6000) que especifica las particiones de base de datos que se añadirán o eliminarán. Los números de partición de base de datos se muestran en formato de serie separado por comas y la serie no permite espacios.

pNumber

Argumento de entrada de tipo VARCHAR (6000) que especifica todos los números de partición de base de datos que corresponden a la ponderación de la partición de base de datos. Todos los números de partición de base de datos están comprendidos entre 0 y 999, se especifican en una serie separada por comas y no se permiten espacios en dicha serie.

pWeight

Argumento de entrada de tipo VARCHAR (6000) que especifica todas las ponderaciones de partición de base de datos que el usuario ha especificado correspondientes a los números de partición de base de datos de la serie *pNumber*. Todas las ponderaciones de partición de base de datos son un número comprendido entre 0 y 32767 y las ponderaciones de partición de base de datos se especifican en una serie separada por comas y sin espacios.

Autorización

- SYSADM, SYSMON, SYSCTRL o SYSMAINT
- Privilegio EXECUTE para el procedimiento ANALYZE_LOG_SPACE

Ejemplo

Analizar el efecto de añadir una partición de base de datos sin aplicar los cambios. En el caso siguiente, se parte de la hipótesis de añadir las particiones de base de datos 40, 50 y 60 al grupo de particiones de base de datos y para las particiones de base de datos 10,20,30,40,50 y 60 se utiliza una relación de destino respectiva de 1:2:1:2:1:2. Tenga en cuenta que sólo las particiones 10, 20 y 30 existen realmente en el grupo de particiones de base de datos

```
CALL SYSPROC.ANALYZE_LOG_SPACE('IBMDEFAULTGROUP', 'TEST',  
    'EMP', 2, ' ', 'A', '40,50,60', '10,20,30,40,50,60',  
    '1,2,1,2,1,2')
```

Analizar el efecto de eliminar una partición de base de datos sin aplicar los cambios. En el caso siguiente, la hipótesis consiste en eliminar la partición de base de datos 30 del grupo de particiones de base de datos y de redistribuir los datos de las particiones de base de datos 10 y 20 utilizando la relación de destino respectiva de 1 : 1. Tenga en cuenta que, en este ejemplo, las particiones de base de datos 10, 20 y 30 deben existir en el grupo de particiones de base de datos

```
CALL SYSPROC.ANALYZE_LOG_SPACE('IBMDEFAULTGROUP', 'TEST',  
    'EMP', 2, ' ', 'D', '30', '10,20', '1,1')
```


Notas de uso

“-1” se utiliza como valor de salida para parámetros cuando no se puede obtener su valor.

Las funciones y los procedimientos almacenados de redistribución sólo funcionan en entornos de base de datos particionada, en los que se ha definido una clave de distribución para cada tabla.

Información devuelta

El procedimiento `ANALYZE_LOG_SPACE` devuelve un conjunto de resultados (un cursor abierto) del análisis del espacio de anotaciones cronológicas, que contiene los campos siguientes por cada partición de base de datos del grupo de particiones de base de datos indicado.

Tabla 197. Información devuelta por el procedimiento `ANALYZE_LOG_SPACE`

| Nombre de columna | Tipo de columna | Descripción |
|------------------------------|-----------------------|--|
| <code>PARTITION_NUM</code> | <code>SMALLINT</code> | Número de la partición de base de datos del análisis del espacio de anotación cronológica. |
| <code>TOTAL_LOG_SIZE</code> | <code>BIGINT</code> | Espacio total de anotación cronológica asignado (en bytes); -1 indica un espacio ilimitado. |
| <code>AVAIL_LOG_SPACE</code> | <code>BIGINT</code> | Cantidad de espacio de anotación cronológica en bytes que es libre y que el proceso de redistribución puede utilizar. |
| <code>DATA_SKEW</code> | <code>BIGINT</code> | Valor absoluto en bytes del tamaño de los datos que se desvía del nivel de destino. |
| <code>REQ_LOG_SPACE</code> | <code>BIGINT</code> | Cantidad necesaria de espacio, indicada en bytes, para llegar a la distribución de datos deseada. |
| <code>NUM_OF_STEPS</code> | <code>SMALLINT</code> | Número de pasos necesarios para reducir el desvío de datos a cero. |
| <code>MAX_STEP_SIZE</code> | <code>BIGINT</code> | Cantidad máxima de datos, indicada en bytes, que se puede mover al mismo tiempo, sin producir un error de anotación cronológica llena. |

Procedimiento `GENERATE_DISTFILE` - Generar archivo de distribución de datos

El procedimiento `GENERATE_DISTFILE` genera un archivo de distribución de datos para la tabla indicada y lo guarda con el nombre de archivo indicado.

Sintaxis

```
►► GENERATE_DISTFILE (—inTbSchema—, —inTbName—, —fileName—) ►►
```

El esquema es `SYSPROC`.

Parámetros del procedimiento

inTbSchema

Argumento de entrada de tipo VARCHAR (128) que especifica el nombre del esquema de tabla.

inTbName

Argumento de entrada de tipo VARCHAR (128) que especifica el nombre de la tabla.

fileName

Argumento de entrada o de salida de tipo VARCHAR (255) que especifica el nombre del archivo de distribución de datos. Si el nombre de archivo indicado sólo es un nombre de archivo, el archivo se guardará en el subdirectorio tmp del directorio de instancias y se devolverá el nombre completo de la vía de acceso al archivo en el parámetro.

Autorización

- Privilegio EXECUTE para el procedimiento GENERATE_DISTFILE.
- Privilegio SELECT para SYSCAT.TABLES, SYSCAT.COLUMNS y la tabla especificada.

Asimismo, el ID de usuario delimitado debe poder crear archivos en el subdirectorio tmp del directorio de instancias.

Ejemplo

Generar un archivo de distribución de datos que el proceso de redistribución utilizará.

```
CALL SYSPROC.GENERATE_DISTFILE('TEST', 'EMP',  
    '$HOME/sql1lib/function/SAMPLE.IBMDEFAULTGROUP_swrData.dst')
```

Notas de uso

Las funciones y los procedimientos almacenados de redistribución sólo funcionan en entornos de base de datos particionada, en los que se ha definido una clave de distribución para cada tabla.

Procedimiento GET_SWRD_SETTINGS - Recuperar información de redistribución

El procedimiento GET_SWRD_SETTINGS lee las grabaciones de registros de redistribución existentes para el grupo de particiones de base de datos dado.

Sintaxis

```
►► GET_SWRD_SETTINGS (—dbpgName—, —matchingSpec—, —redistMethod—, —————►  
► pMapFile—, —distFile—, —stepSize—, —totalSteps—, —stageSize—, —————►  
► nextStep—, —processState—, —pNumber—, —pWeight—) —————►◀
```

El esquema es SYSPROC.

Parámetros del procedimiento

dbpgName

Argumento de entrada de tipo VARCHAR(128) que especifica el nombre del grupo de particiones de base de datos en el que se ejecutará el proceso de redistribución.

matchingSpec

Argumento de entrada de tipo SMALLINT que especifica los identificadores del campo a nivel de bit a partir de la Tabla 198, indicando los campos de destino que los parámetros de salida deben devolver. Los parámetros de salida que no se necesitan se pueden establecer en nulos.

Por ejemplo, si *matchingSpec* se ha establecido en 96, que es el valor de entero de (REDIST_STAGE_SIZE | REDIST_NEXT_STEP), el proceso que realiza la llamada de esta función sólo debe proporcionar *stageSize* y *nextStep* para recibir los valores, y los parámetros de salida restantes pueden ser nulos.

Tabla 198. Identificadores de campo a nivel de bit

| Nombre de campo | Valor hexadecimal | Valor decimal |
|---------------------------|-------------------|---------------|
| REDIST_METHOD | 0x0001<<0 | 1 |
| REDIST_PMAP_FILE | 0x0001<<1 | 2 |
| REDIST_DIST_FILE | 0x0001<<2 | 4 |
| REDIST_STEP_SIZE | 0x0001<<3 | 8 |
| REDIST_NUM_STEPS | 0x0001<<4 | 16 |
| REDIST_STAGE_SIZE | 0x0001<<5 | 32 |
| REDIST_NEXT_STEP | 0x0001<<6 | 64 |
| REDIST_PROCESS_STATE | 0x0001<<7 | 128 |
| REDIST_PWEIGHT_START_NODE | 0x0001<<8 | 256 |
| REDIST_PWEIGHT | 0x0001<<9 | 512 |

redistMethod

Argumento de salida de tipo SMALLINT que especifica si la redistribución se realizará utilizando el archivo de distribución de datos o la correlación de distribución de destino. Hay dos valores de retorno posibles:

- 2: indica que el proceso de redistribución funcionará con un archivo de distribución de datos como entrada.
- 3: indica que el proceso de redistribución funcionará con una correlación de distribución de destino como entrada.

pMapFile

Argumento de salida de tipo VARCHAR (255) que especifica la vía de acceso completa de la correlación de distribución de destino en el servidor de base de datos.

distFile

Argumento de salida de tipo VARCHAR (255) que especifica la vía de acceso completa del archivo de distribución de datos en el servidor de base de datos.

stepSize

Argumento de salida de tipo BIGINT que especifica el número máximo de filas que se pueden mover antes de que se deba solicitar una confirmación, para evitar una situación completa de anotaciones cronológicas. El número se puede cambiar en cada paso de la redistribución.

totalSteps

Argumento de salida de tipo SMALLINT que especifica el número de pasos que comporta redistribuir totalmente el grupo de particiones de base de datos dado.

stageSize

Argumento de salida de tipo SMALLINT que especifica el número de pasos que se deben dar de manera consecutiva.

nextStep

Argumento de salida de tipo SMALLINT que especifica el índice, separando los pasos que se han dado ya y los que quedan pendientes.

processState

Argumento de salida de tipo SMALLINT que indica si el proceso de redistribución se detendrá o no en el punto de comprobación siguiente. Se pone un punto de comprobación al principio de cada paso de redistribución. Si este argumento se establece en 1, el paso no se iniciará; si el valor es 0, el paso proseguirá.

pNumber

Argumento de salida de tipo VARCHAR (6000) que puede devolver una lista de números de partición de base de datos separados por comas con formato de serie. Dichos números de partición pueden ser particiones de base de datos que el grupo de particiones de base de datos está utilizando en ese momento, o bien las particiones que se añadirán o descartarán. La secuencia y el recuento de estos números de partición se corresponden con la ponderación de la partición de destino que devuelve la variable *pWeight*.

pWeight

Argumento de salida de tipo VARCHAR (6000) que puede devolver una lista de números de ponderación de partición de base de datos de destino separados por comas. La secuencia y el número de estas ponderaciones de partición se corresponden con los números de partición devueltos por la variable *pNumber*.

Autorización

Privilegio EXECUTE para el procedimiento GET_SWRD_SETTINGS.

Ejemplo

Informar del contenido del plan de redistribución por pasos para el grupo de particiones de base de datos dado.

```
CALL SYSPROC.GET_SWRD_SETTINGS  
('IBMDEFAULTGROUP', 255, ?, ?, ?, ?, ?, ?, ?, ?)
```

Nota sobre uso

Las funciones y los procedimientos almacenados de redistribución sólo funcionan en entornos de base de datos particionada, en los que se ha definido una clave de distribución para cada tabla.

Procedimiento SET_SWRD_SETTINGS - Crear o cambiar registro de redistribución

El procedimiento SET_SWRD_SETTINGS crea o introduce cambios en el registro de redistribución. Si dicho registro no existe, lo crea y le añade registros. Si ya existe, utilizará *overwriteSpec* para identificar los valores de campo sobre los que deberá grabarse. El campo *overwriteSpec* permite que esta función tome las entradas NULL para los campos que no necesiten actualizarse.

Sintaxis

```
► SET_SWRD_SETTINGS (—dbpgName—, —overwriteSpec—, —redistMethod—, —
► pMapFile—, —distFile—, —stepSize—, —totalSteps—, —stageSize—, —
► nextStep—, —processState—, —pNumber—, —pWeight—)
```

El esquema es SYSPROC.

Parámetros del procedimiento

dbpgName

Argumento de entrada de tipo VARCHAR(128) que especifica el nombre del grupo de particiones de base de datos en el que se ejecutará el proceso de redistribución.

overwriteSpec

Identificadores de campo a nivel de bit de la Tabla 199 que indican los campos de destino que se grabarán o sobre los que se grabará en el registro de valores de redistribución.

Tabla 199. Identificadores de campo a nivel de bit

| Nombre de campo | Valor hexadecimal | Valor decimal |
|---------------------------|-------------------|---------------|
| REDIST_METHOD | 0x0001<<0 | 1 |
| REDIST_PMAP_FILE | 0x0001<<1 | 2 |
| REDIST_DIST_FILE | 0x0001<<2 | 4 |
| REDIST_STEP_SIZE | 0x0001<<3 | 8 |
| REDIST_NUM_STEPS | 0x0001<<4 | 16 |
| REDIST_STAGE_SIZE | 0x0001<<5 | 32 |
| REDIST_NEXT_STEP | 0x0001<<6 | 64 |
| REDIST_PROCESS_STATE | 0x0001<<7 | 128 |
| REDIST_PWEIGHT_START_NODE | 0x0001<<8 | 256 |
| REDIST_PWEIGHT | 0x0001<<9 | 512 |

redistMethod

Argumento de entrada de tipo SMALLINT que especifica si la redistribución se ejecutará mediante el archivo de distribución de datos o la correlación de distribución de destino. Los dos valores de entrada válidos son:

- 2: Permite indicar que el proceso de redistribución funcionará con un archivo de distribución de datos como entrada.

- 3: Permite indicar que el proceso de redistribución funcionará con una correlación de distribución de destino como entrada.

pMapFile

Argumento de entrada de tipo VARCHAR (255) que especifica la vía de acceso completa de la correlación de distribución de destino en el servidor de la base de datos.

distFile

Argumento de entrada de tipo VARCHAR (255) que especifica la vía de acceso completa al archivo de distribución de datos en el servidor de base de datos.

stepSize

Argumento de entrada de tipo BIGINT que especifica el número máximo de filas que se pueden mover antes de que deba llamarse a una confirmación para evitar una situación de anotaciones cronológicas completas. El número se puede cambiar en cada paso de redistribución. El valor "-2" se puede utilizar para *stepSize* para indicar que el número es ilimitado.

totalSteps

Argumento de entrada de tipo SMALLINT que especifica el número de pasos que es preciso efectuar para redistribuir completamente el grupo de particiones de base de datos determinado. El valor "-2" se puede utilizar para *totalSteps* para indicar que el número es ilimitado.

stageSize

Argumento de entrada de tipo SMALLINT que especifica el número de pasos que se ejecutarán de forma consecutiva.

nextStep

Argumento de entrada de tipo SMALLINT que especifica el índice que separa los pasos que se han completado y qué queda por ejecutarse.

processState

Argumento de entrada de tipo SMALLINT que indica si el proceso de redistribución se detendrá en el siguiente punto de comprobación. Se pone un punto de comprobación al principio de cada paso de redistribución. Si este argumento se establece en 1, el paso no se iniciará; si el valor es 0, el paso proseguirá.

pNumber

Argumento de entrada de tipo VARCHAR (6000) que puede contener una lista de números de partición de base de datos separados por comas en formato de serie. Dichos números de partición pueden ser particiones de base de datos que el grupo de particiones de base de datos está utilizando en ese momento, o bien las particiones que se añadirán o descartarán. La secuencia y el recuento de estos números de partición se corresponden con la ponderación de la partición de destino que devuelve la variable *pWeight*. Todos los números de partición de base de datos están comprendidos entre 0 y 999 y no se permiten espacios en la serie.

pWeight

Argumento de entrada de tipo VARCHAR (6000) que puede contener una serie separada por comas de todas las ponderaciones de partición de base de datos que ha especificado el usuario, correspondientes a los números de partición de base de datos de la serie *pNumber*. Todas las ponderaciones de partición de base de datos son un número comprendido entre 0 y 32767 y no se permiten espacios en la serie.

Autorización

Privilegio EXECUTE para el procedimiento SET_SWRD_SETTINGS.

Ejemplo

Grabar un plan de redistribución por pasos en un registro. Si se establece *processState* en 1, es posible que un procedimiento almacenado de redistribución por pasos que se esté ejecutando en esos momentos acabe el paso que está ejecutando y se detenga hasta que este parámetro se restablezca en 0 y se vuelva a llamar el procedimiento almacenado de redistribución.

```
CALL SYSPROC.SET_SWRD_SETTINGS('IBMDEFAULTGROUP', 255, 0, ' ',  
    '$HOME/sql1lib/function/TEST.IBMDEFAULTGROUP_swrData.dst', 1000,  
    12, 2, 1, 0, '10,20,30', '50,50,50')
```

Notas de uso

Las funciones y los procedimientos almacenados de redistribución sólo funcionan en entornos de base de datos particionada, en los que se ha definido una clave de distribución para cada tabla.

Procedimiento STEPWISE_REDISTRIBUTE_DBPG - Redistribuir parte del grupo de particiones de base de datos

El procedimiento STEPWISE_REDISTRIBUTE_DBPG redistribuye parte del grupo de particiones de base de datos según las entradas especificadas para el procedimiento y el archivo de configuración creado o actualizado mediante el procedimiento SET_SWRD_SETTINGS.

Sintaxis

```
►►STEPWISE_REDISTRIBUTE_DBPG(—inDBPGroup—,—inStartingPoint—,——————►  
►—inNumSteps—)—————►►
```

El esquema es SYSPROC.

Parámetros del procedimiento

inDBPGroup

Argumento de entrada de tipo VARCHAR (128) que especifica el nombre del grupo de particiones de base de datos de destino.

inStartingPoint

Argumento de entrada de tipo SMALLINT que especifica el punto de inicio que se utilizará. Si el parámetro está configurado en un valor entero positivo y no es NULL, el procedimiento STEPWISE_REDISTRIBUTE_DBPG utilizará este valor y no el valor *nextStep* especificado en el archivo de configuración. Se trata de una opción útil para los casos en los que se desee volver a ejecutar el procedimiento STEPWISE_REDISTRIBUTE_DBPG desde un paso determinado. Si el parámetro está establecido en NULL, se utilizará el valor *nextStep*.

inNumSteps

Argumento de entrada de tipo SMALLINT que especifica el número de pasos que se ejecutarán. Si el parámetro está configurado en un valor entero positivo y no es NULL, el procedimiento STEPWISE_REDISTRIBUTE_DBPG utilizará este

valor y no el valor *stageSize* especificado en el archivo de configuración. Se trata de una opción útil para los casos en los que se desee volver a ejecutar el procedimiento `STEPWISE_REDISTRIBUTE_DBPG` con un número de pasos diferente del especificado en los valores. Por ejemplo, si una etapa planificada tiene cinco pasos y el proceso de redistribución ha dado un resultado anómalo en el paso 3, se puede llamar al proceso `STEPWISE_REDISTRIBUTE_DBPG` para que ejecute los tres pasos restantes una vez se haya corregido la condición de error. Si el parámetro está establecido en `NULL`, se utilizará el valor *stageSize*. El valor `-2` se puede utilizar en este procedimiento para indicar que el número es ilimitado.

Nota: No hay ningún parámetro para especificar la equivalencia a la opción **NOT ROLLFORWARD RECOVERABLE** en el mandato `REDISTRIBUTE DATABASE PARTITION GROUP`. El registro siempre se realiza para la redistribución de datos de fila que se realiza cuando se utiliza el procedimiento `STEPWISE_REDISTRIBUTE_DBPG`.

Autorización

- Privilegio `EXECUTE` para el procedimiento `STEPWISE_REDISTRIBUTE_DBPG`
- `SYSADM`, `SYSCTRL` o `DBADM`

Ejemplo

Redistribuir el grupo de particiones de base de datos "IBMDEFAULTGROUP" según el plan de redistribución almacenado en el registro por el procedimiento `SET_SWRD_SETTINGS`. Empieza en el paso 3 y redistribuye los datos hasta finalizar 2 pasos del plan de redistribución.

```
CALL SYSPROC.STEPWISE_REDISTRIBUTE_DBPG('IBMDEFAULTGROUP', 3, 2)
```

Notas de uso

Si el valor de registro de *processState* se actualiza a 1 siguiendo el procedimiento `SET_SWRD_SETTINGS` después de iniciar la ejecución del procedimiento `STEPWISE_REDISTRIBUTE_DBPG`, el proceso se detendrá al principio del paso siguiente y se devolverá un mensaje de aviso.

Dado que el proceso de redistribución llama a la sentencia de `SQL COMMIT`, no se soporta la ejecución del proceso de redistribución en una conexión de tipo 2.

Capítulo 17. Rutinas de la herramienta de gestión de almacenamiento

Procedimiento `CAPTURE_STORAGEMGMT_INFO` – Recuperar información relacionada con el almacenamiento para un objeto raíz concreto

El procedimiento `CAPTURE_STORAGEMGMT_INFO` intenta recopilar información relacionada con el almacenamiento para un objeto raíz determinado, así como los objetos de almacenamiento definidos dentro del ámbito. Todos los objetos de almacenamiento están especificados en la tabla `SYSTOOLS.STMG_OBJECT_TYPE`.

Tabla 200. Tabla `STMG_OBJECT_TYPE`

| Nombre de columna | Tipo de datos | Posibilidad de nulos | Descripción |
|------------------------|----------------------|----------------------|--|
| <code>OBJ_TYPE</code> | <code>INTEGER</code> | N | Valor entero que corresponde a un tipo de objeto de almacenamiento <ul style="list-style-type: none">• 0 - Base de datos• 1 - Grupo de particiones de base de datos• 2 - Espacio de tablas• 3 - Contenedor de espacio de tablas• 4 - Tabla• 5 - Índice |
| <code>TYPE_NAME</code> | <code>VARCHAR</code> | N | Nombre descriptivo del tipo de objeto de almacenamiento <ul style="list-style-type: none">• <code>STMG_DATABASE</code>• <code>STMG_DBPGROUP</code>• <code>STMG_TABLESPACE</code>• <code>STMG_CONTAINER</code>• <code>STMG_TABLE</code>• <code>STMG_INDEX</code> |

Sintaxis

```
►►—CAPTURE_STORAGEMGMT_INFO—(—in_rootType—,—in_rootSchema—,—  
►—in_rootName—)—
```

El esquema es `SYSPROC`.

Parámetros del procedimiento

in_rootType

Argumento de entrada de tipo `SMALLINT`. Los tipos de opciones válidos son:

- 0 - Base de datos
- 1 - Grupo de particiones de base de datos

- 2 - Espacio de tablas
- 4 - Tabla
- 5 - Índice

El argumento de entrada no puede ser nulo. Si se especifica un valor nulo, se devolverá un error SQL0443 con SQLSTATE 38553 y el símbolo DBA7617.

in_rootSchema

Argumento de entrada de tipo VARCHAR (128) que especifica el nombre de esquema del objeto raíz de la instantánea de almacenamiento.

in_rootName

Argumento de entrada de tipo VARCHAR (128) que especifica el nombre del objeto raíz. El argumento de entrada no puede ser nulo. Si se especifica un valor nulo, se devolverá un error SQL0443 con SQLSTATE 38553 y el símbolo DBA7617.

Autorización

- Privilegio EXECUTE para el procedimiento CAPTURE_STORAGEMGMT_INFO.
- Privilegio EXECUTE para las funciones de tabla SYSPROC.DB_PARTITIONS, SYSPROC.SNAP_GET_CONTAINER, SYSPROC.SNAPSHOT_CNTRFS.
- Privilegio SELECT para SYSCAT.TABLES, SYSCAT.TABLESPACES, SYSCAT.NODEGROUPDEF, SYSCAT.DATABASEPARTITIONS, SYSCAT.DATAPARTITIONEXPRESSION, SYSCAT.INDEXES y SYSCAT.COLUMNS.

Procedimiento CREATE_STORAGEMGMT_TABLES - Crear tablas de gestión de almacenamiento

El procedimiento CREATE_STORAGEMGMT_TABLES crea todas las tablas de gestión de almacenamiento bajo un esquema "DB2TOOLS" fijo, en el espacio de tabla especificado por la entrada.

Sintaxis

```
▶▶—CREATE_STORAGEMGMT_TABLES—(—in_tbspace—)—————▶▶
```

El esquema es SYSPROC.

Parámetros del procedimiento

in_tbspace

Argumento de entrada de tipo VARCHAR (128) que especifica el nombre de espacio de tablas. El argumento de entrada no puede ser nulo. Si se especifica un valor nulo, se devolverá un error SQL0443 con SQLSTATE 38553 y el símbolo DBA7617.

Autorización

Privilegio EXECUTE para el procedimiento CREATE_STORAGEMGMT_TABLES.

Es preciso disponer también del privilegio CREATETAB para la base de datos y del privilegio USE para el espacio de tablas, y de uno de los siguientes:

- Autorización IMPLICIT_SCHEMA para la base de datos, si el nombre de esquema implícito o explícito DB2TOOLS no existe.
- Privilegio CREATEIN para el esquema, si el nombre de esquema de la tabla existe.
- Autorización DBADM

Notas de uso

Se crean las tablas siguientes en el esquema DB2TOOLS:

- STMG_CONTAINER
- STMG_CURR_THRESHOLD
- STMG_DATABASE
- STMG_DBPARTITION
- STMG_DBPGROUP
- STMG_HIST_THRESHOLD
- STMG_INDEX
- STMG_OBJECT
- STMG_OBJECT_TYPE
- STMG_ROOT_OBJECT
- STMG_TABLE
- STMG_TABLESPACE
- STMG_TBPARTITION
- STMG_THRESHOLD_REGISTRY

Procedimiento DROP_STORAGEMGMT_TABLES - Descartar todas las tablas de gestión de almacenamiento

El procedimiento DROP_STORAGEMGMT_TABLES intenta eliminar todas las tablas de gestión de almacenamiento.

Sintaxis

►—DROP_STORAGEMGMT_TABLES—(—*dropSpec*—)—————►

El esquema es SYSPROC.

Parámetros del procedimiento

dropSpec

Argumento de entrada de tipo SMALLINT. Cuando *dropSpec* está establecido en 0, el proceso se detiene cuando encuentra algún error; cuando *dropSpec* está establecido en 1, el proceso continúa sin tener en cuenta los errores que encuentra. El argumento de entrada no puede ser nulo. Si se especifica un valor nulo, se devolverá un error SQL0443 con SQLSTATE 38553 y el símbolo DBA7617.

Autorización

Privilegio EXECUTE para el procedimiento DROP_STORAGEMGMT_TABLES.

El ID de usuario que establece la conexión de base de datos debe ser el definidor de las tablas de gestión de almacenamiento tal como están registradas en la columna DEFINER de SYSCAT.TABLES o debe disponer, como mínimo, de los privilegios siguientes:

- Autorización DBADM
- Privilegio DROPIN para el esquema de dichas tablas
- Privilegio CONTROL para dichas tablas

Capítulo 18. Rutinas de búsqueda de texto

Procedimiento almacenado SYSTS_ADMIN_CMD - Ejecutar mandatos de administración de búsqueda de texto

Las aplicaciones utilizan el procedimiento SYSTS_ADMIN_CMD para ejecutar mandatos administrativos de búsqueda de texto mediante la sentencia CALL de SQL.

Sintaxis

```
►►—SYSTS_ADMIN_CMD—(—serie-mandato—,—entorno_local_mensaje—,—mensaje—)——►►
```

El esquema es SYSPROC.

Parámetro del procedimiento

serie-mandato

Argumento de entrada de tipo VARCHAR (32K) que especifica un único mandato de administración de índice de búsqueda de texto que ha de ejecutarse. La sintaxis del mandato es la misma que la del mandato de Búsqueda de texto de DB2 a excepción de las opciones de conexión. Las opciones de conexión no están soportadas en este procedimiento. Los mandatos que se ejecutan a través de este procedimiento utilizan la conexión actual.

entorno-local-mensaje

Argumento de entrada de tipo VARCHAR (33) que especifica el idioma en que se desea que se devuelva cualquier texto de mensaje de error. Si el argumento es nulo o una serie vacía o los archivos de mensajes del entorno local especificado no están disponibles en el servidor, se utiliza 'en_US'.

mensaje

Argumento de salida de tipo VARCHAR (32K) que especifica un aviso o mensaje informativo para una operación que se considere que se ha completado de modo satisfactorio.

Autorización

Privilegio EXECUTE para el procedimiento SYSTS_ADMIN_CMD.

El procedimiento da soporte actualmente a los siguientes mandatos de Búsqueda de texto de DB2:

- ALTER INDEX
- CLEAR COMMAND LOCKS
- CLEAR EVENTS
- CREATE INDEX
- DISABLE DATABASE
- DROP INDEX
- ENABLE DATABASE
- UPDATE INDEX

Ejemplo

Actualizar el índice de búsqueda de texto MYTEXTINDEX en el esquema DB2TS y devolver los mensajes de error en inglés.

```
CALL SYSPROC.SYSTS_ADMIN_CMD  
( 'UPDATE INDEX DB2TS.MYTEXTINDEX FOR TEXT', 'en_US', ? );
```

El ejemplo siguiente muestra los datos de salida de la consulta.

Valor parámetros de salida

```
-----  
Nombre de parámetro : MESSAGE  
Valor de parámetro  : CIE00001 La operación ha finalizado satisfactoriamente.
```

Estado de devolución = 0

Notas de uso

- Si la ejecución del mandato no ha resultado satisfactoria, se devuelve SQLCODE -20427 y SQLSTATE 38H14 con el mensaje de error específico de búsqueda de texto. Por ejemplo, si ya existe el índice MYTEXTINDEX y se emite la siguiente sentencia:

```
CALL SYSPROC.SYSTS_ADMIN_CMD ('CREATE INDEX MYTEXTINDEX FOR TEXT  
ON DB2TS.TEXTBOOKS (STORY)', 'en_US', ?)
```

la creación del índice fallará con el siguiente mensaje de error.

```
SQL20427N Se ha producido un error durante un procedimiento o mandato de  
administración de búsqueda de texto. El mensaje de error es "CIE00201 El índice  
de búsqueda de texto "DB2TS ".MYTEXTINDEX" ya existe. ". SQLSTATE=38H14
```

- Si el procedimiento devuelve SQLCODE, es posible que el mensaje esté truncado. Puede hallar información completa sobre el mensaje en los archivos de anotaciones cronológicas db2diag.

Procedimiento SYSTS_ALTER - Cambiar las características de actualización de un índice

Este procedimiento cambia las características de actualización de un índice.

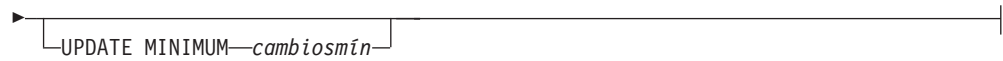
El procedimiento emite un mandato de administración de búsqueda de texto ALTER INDEX en el servidor de base de datos.

Sintaxis

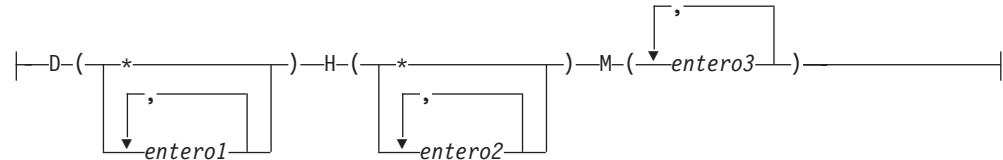
```
►► SYSTS_ALTER (—esquema_índice—, —nombre_índice—, —————►  
► | características de actualización | —————►  
►, —entorno_local_mensaje—, —mensaje—) —————►
```

Características de actualización:

```
|  
| UPDATE FREQUENCY | NONE |  
| frecuencia de actualización |
```



frecuencia de actualización:



El esquema es SYSPROC.

Parámetros del procedimiento

esquema_índice

Argumento de entrada de tipo VARCHAR(128) que especifica el esquema del índice de búsqueda de texto. El *esquema_índice* debe seguir las restricciones sobre nombres para los nombres de esquema de DB2. Si el argumento es nulo o una serie vacía, se utilizará el valor de CURRENT SCHEMA. El *esquema-índice* es sensible a mayúsculas y minúsculas.

nombre-índice

Argumento de entrada de tipo VARCHAR(128) que especifica el nombre del índice. Junto con *esquema_índice*, identifica de modo exclusivo un índice de búsqueda de texto en una base de datos. El *nombre-índice* es sensible a mayúsculas y minúsculas.

Características de actualización

Argumento de entrada de tipo VARCHAR(32K) que especifica las opciones de modificación. Las opciones de modificación permitidas son las siguientes:

UPDATE FREQUENCY

Especifica la frecuencia con la que se efectúan las actualizaciones del índice. El índice se actualizará, si el número de cambios es como mínimo el valor establecido para UPDATE MINIMUM. La frecuencia de actualización NONE indica que no se efectuará ninguna actualización de índices adicional. Esto puede ser útil para una columna de texto de una tabla con datos que no cambiarán. También resulta útil cuando el usuario intenta actualizar manualmente el índice (utilizando el mandato UPDATE INDEX). Las actualizaciones automáticas sólo se pueden realizar si se ha ejecutado el mandato START FOR TEXT y se ejecutan los servicios de la instancia de búsqueda de texto de DB2.

El valor de la frecuencia por omisión se toma de la vista SYSIBMTS.TSDEFAULTS, donde DEFAULTNAME='UPDATEFREQUENCY'.

NONE

No se aplicará ninguna actualización automática al índice de texto. Cualquier actualización de índice posterior tendrá que iniciarse manualmente.

D El día o días de la semana en los que se actualiza el índice.

* Cada día de la semana.

entero1

Días específicos de la semana, de domingo a sábado: de 0 a 6

H La hora u horas del día o días especificados en que se actualiza el índice.

* Cada hora del día.

entero2

Horas específicas del día, desde medianoche a las 11 de la noche: de 0 a 23

M El minuto o minutos de la hora u horas especificadas en que se actualiza el índice.

entero3

Se especifica además de la hora (0), o en múltiplos de incrementos de 5 minutos después de la hora: 0, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50 ó 55

Si no especifica la opción UPDATE FREQUENCY, los valores de frecuencia se dejan sin cambios.

UPDATE MINIMUM *cambiosmín*

Especifica el número mínimo de cambios en los documentos de texto que deben producirse antes de que el índice se actualice de manera incremental. Múltiples cambios en el mismo documento de texto se tratan como cambios separados. Si no especifica la opción UPDATE MINIMUM, el valor se deja sin cambios.

entorno_local_mensaje

Argumento de entrada de tipo VARCHAR(33) que especifica el entorno local que ha de utilizarse para los mensajes de error devueltos. Si el argumento es nulo o una serie vacía o los archivos de mensajes del entorno local especificado no están disponibles en el servidor, se utiliza 'en_US'.

mensaje

Argumento de salida de tipo VARCHAR(32K) que especifica un aviso o mensaje informativo para una operación completada de modo satisfactorio.

Autorización

El ID de autorización de la sentencia debe tener al menos uno de los privilegios siguientes:

- El privilegio CONTROL en la tabla en la que está definido el índice de texto
- Autorización DBADM

Ejemplos

Ejemplo 1: En el ejemplo siguiente, las características de actualización de un índice de búsqueda de texto se están modificando. Este índice se creó originalmente con *esquema_índice* 'db2ts' y *nombre_índice* 'myTextIndex'. Utilizando 'UPDATE FREQUENCY NONE', la intención es la de no hacer más actualizaciones en el índice de búsqueda de texto ya que posiblemente ya no se espera ningún cambio para la columna de tabla asociada. Los mensajes de error han de devolverse en inglés. Cuando el procedimiento resulta satisfactorio, se devuelve al que llama el mensaje de parámetro de salida que indica que la operación ha resultado satisfactoria.


```
CALL SYSPROC.SYSTS_ALTER('db2ts', 'myTextIndex',  
'UPDATE FREQUENCY_NONE', 'en_US', ?)
```

El ejemplo siguiente muestra los datos de salida de la consulta.

Valor parámetros de salida

Nombre de parámetro : MESSAGE

Valor de parámetro : La operación ha finalizado satisfactoriamente.

Estado de devolución = 0

Ejemplo 2: En el ejemplo siguiente, se llama al procedimiento almacenado SYSTS_ALTER para modificar las características de actualización para un índice de búsqueda de texto con *esquema índice* 'db2ts' y *nombre índice* 'myTextIndex'. La intención es la de asegurar que se produzcan actualizaciones en el índice cada hora a la hora en punto. Sin embargo, este índice no existe y da como resultado un error.

```
CALL SYSPROC.SYSTS_ALTER('db2ts', 'myTextIndex',  
'update frequency D(*) H(*) M(0)', 'en_US', ?)
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
SQL20427N Se ha producido un error durante un procedimiento o mandato de  
administración de búsqueda de texto. El mensaje de error es "CIE00316 El índice de  
búsqueda de texto "db2ts"."myTextIndex" no existe. ". SQLSTATE 38H14
```

Notas de uso

- Los procedimientos de administración de búsqueda de texto utilizan una conexión existente con la base de datos. La transacción actual puede confirmarse o retrotraerse en función de que se completen los procedimientos. Como tal, es posible que desee confirmar todos los cambios de transacciones para evitar cualquier impacto inesperado de dicha confirmación o retroacción. Una forma de conseguirlo es la de activar AUTOCOMMIT.
- Varios procedimientos o mandatos no pueden ejecutarse de modo simultáneo en un índice de búsqueda de texto en el caso de que puedan entrar en conflicto. Algunos de los ejemplos de mandatos y procedimientos conflictivos son:
 - Procedimiento SYSTS_ALTER o mandato ALTER INDEX db2ts
 - Procedimiento SYSTS_CLEAR_EVENTS o mandato CLEAR EVENTS FOR INDEX db2ts
 - Procedimiento SYSTS_DISABLE o mandato DISABLE DATABASE FOR TEXT db2ts
 - Procedimiento SYSTS_DROP o mandato DROP INDEX db2ts
 - Mandato STOP FOR TEXT db2ts
 - Procedimiento SYSTS_UPDATE o mandato UPDATE INDEX db2ts

Si se produce un conflicto, el procedimiento devuelve un SQLCODE -20426 y SQLSTATE 38H13.

- Cuando se ejecuta este procedimiento,
 - se actualiza el contenido de la vista de Búsqueda de texto DB2 SYSIBMTS.TSLOCKS.
 - se actualiza la entrada de índice del archivo de datos de índice de Búsqueda de texto. El archivo contiene una representación permanente de las planificaciones de actualización (también las vacías) para cada uno de los índices de la instancia.

Procedimiento SYSTS_CLEAR_COMMANDLOCKS - Eliminar bloqueos de mandato para índices de búsqueda de texto

Este procedimiento elimina todos los bloqueos de mandatos para un índice de búsqueda de texto específico o para todos los índices de búsqueda de texto en la base de datos.

Un bloqueo de mandato se crea al principio de un mandato de índice de búsqueda y se destruye cuando se completa el mandato. Impide un conflicto no deseado entre diferentes mandatos.

Se realiza automáticamente una limpieza de todos los bloqueos asociados a procesos que ya no están activos. Esta acción se lleva a cabo para que un índice de búsqueda de texto se pueda acceder para una nueva petición de búsqueda. La utilización de este procedimiento es necesaria en el caso extraño de que los bloqueos sigan activos debido a un comportamiento inesperado del sistema y que tengan que limpiarse de modo explícito.

Este procedimiento emite el mandato de administración de búsqueda de texto CLEAR COMMAND LOCKS en el servidor de base de datos.

Sintaxis

```
►►SYSTS_CLEAR_COMMANDLOCKS—(—esquema_índice—,—nombre_índice—,——————►  
►—entorno_local_mensaje—,—mensaje—)—————►►
```

El esquema es SYSPROC.

Parámetros del procedimiento

esquema_índice

Argumento de entrada de tipo VARCHAR(128) que especifica el esquema del índice de texto. El *esquema_índice* debe seguir las restricciones sobre nombres para los nombres de esquema de DB2. Si el argumento es nulo o una serie vacía, se utilizará el valor de CURRENT SCHEMA. El *esquema-índice* es sensible a mayúsculas y minúsculas.

nombre-índice

Argumento de entrada de tipo VARCHAR(128) que especifica el nombre del índice. Junto con *esquema_índice*, identifica de modo exclusivo un índice de búsqueda de texto en una base de datos. Si el argumento es nulo o una serie vacía, el procedimiento suprime los bloqueos de mandato para todos los índices de búsqueda de texto de la base de datos. El *nombre-índice* es sensible a mayúsculas y minúsculas.

entorno_local_mensaje

Argumento de entrada de tipo VARCHAR(33) que especifica el entorno local que ha de utilizarse para los mensajes de error devueltos. Si el argumento es nulo o una serie vacía o los archivos de mensajes del entorno local especificado no están disponibles en el servidor, se utiliza 'en_US'.

mensaje

Argumento de salida de tipo VARCHAR(32K) que especifica un aviso o mensaje informativo para una operación completada de modo satisfactorio.

Autorización

El *nombreusuario* para la conexión de base de datos debe tener la autorización DBADM si no se especifica un nombre de índice. Para quitar un bloqueo de mandato en un índice específico, el *nombreusuario* para la conexión de base de datos debe tener el privilegio CONTROL en la tabla para la que se ha creado el índice de búsqueda de texto.

Ejemplos

Ejemplo 1: En el ejemplo siguiente, se emite SYSTS_CLEAR_COMMANDLOCKS para un índice de búsqueda de texto con *esquema_índice* 'db2ts' y *nombre_índice* 'myTextIndex'. Se ha pedido que los mensajes de error se devuelvan en inglés. Cuando el procedimiento resulta satisfactorio, se devuelve al que llama el mensaje de parámetro de salida que indica que la operación ha resultado satisfactoria.

```
CALL SYSPROC.SYSTS_CLEAR_COMMANDLOCKS('db2ts', 'myTextIndex', 'en_US', ?)
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
Valor parámetros de salida
```

```
-----
```

```
Nombre de parámetro : MESSAGE
```

```
Valor de parámetro  : La operación ha finalizado satisfactoriamente.
```

```
Estado de devolución = 0
```

Ejemplo 2: En el ejemplo siguiente, se llama a SYSTS_CLEAR_COMMANDLOCKS para borrar los bloqueos de mandato para un índice de búsqueda de texto con *esquema_índice* 'db2ts' y *nombre_índice* 'myTextIndex'. Este índice no existe y el procedimiento devuelve un mensaje de error.

```
CALL SYSPROC.SYSTS_CLEAR_COMMANDLOCKS('db2ts', 'myTextIndex', 'en_US', ?)
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
SQL20427N Se ha producido un error durante un procedimiento o mandato de
administración de búsqueda de texto. El mensaje de error es "C1E00316 El índice de
búsqueda de texto "db2ts"."myTextIndex" no existe. ". SQLSTATE 38H14
```

Notas de uso

- Los procedimientos de administración de búsqueda de texto utilizan una conexión existente con la base de datos. La transacción actual puede confirmarse o retrotraerse en función de que se completen los procedimientos. Como tal, es posible que desee confirmar todos los cambios de transacciones para evitar cualquier impacto inesperado de dicha confirmación o retroacción. Una forma de conseguirlo es la de activar AUTOCOMMIT.
- La información de hebra y proceso de la vista SYSIBMTS.TSLOCKS puede utilizarse para comprobar si sigue existiendo la hebra o proceso que mantiene el bloqueo. Los bloqueos para los procesos existentes que pertenezcan a un mandato o procedimiento de búsqueda de texto en ejecución (por ejemplo, SYSTS_UPDATE o UPDATE INDEX) no deberían borrarse.
- Ha de invocarse este procedimiento ya que el proceso que posee el bloqueo de mandato está muerto. En este caso, es posible que el mandato (representado por el bloqueo) no se haya completado y que el índice ya no resulte operativo. Deberá realizar la acción adecuada. Por ejemplo, el proceso que ejecuta el mandato DROP INDEX muere de modo repentino. Ha suprimido algunos datos del índice, pero no toda la información de recopilación y catálogo. El bloqueo del mandato se deja intacto. Después de borrar el bloqueo del mandato DROP

INDEX, es posible que desee volver a ejecutar el procedimiento SYSTS_DROP. En otro ejemplo, el proceso que ejecuta el procedimiento SYSTS_CREATE muere de modo repentino. Ha creado alguna información de colección y catálogo de índice, pero no toda. El bloqueo del mandato se deja intacto. Después de borrar el bloqueo del mandato, podrá ejecutar los procedimientos SYSTS_DROP y SYSTS_CREATE.

- Cuando se ejecuta este procedimiento, se actualiza el contenido de la vista de Búsqueda de texto DB2 SYSIBMTS.TSLOCKS.

Procedimiento SYSTS_CLEAR_EVENTS - Suprimir los sucesos de indexación de una tabla de sucesos de índice

Este procedimiento suprime los sucesos de indexación de una tabla de sucesos de índice utilizados para la administración.

El nombre de la tabla de sucesos puede hallarse en la vista SYSIBMTS.TSINDEXES de la columna EVENTVIEWNAME. Cada operación de actualización de índice genera entradas informativas y, en algunos casos, entradas de error en la tabla de sucesos. Para las actualizaciones automáticas, la tabla de sucesos ha de inspeccionarse con regularidad. Los errores específicos en los documentos deben corregirse cambiando el contenido del documento. Después de corregir los errores, los sucesos pueden borrarse (y deberían borrarse al objeto de no consumir demasiado espacio).

El procedimiento emite un mandato de administración de búsqueda de texto CLEAR EVENTS FOR INDEX en el servidor de base de datos.

Sintaxis

```
►►SYSTS_CLEAR_EVENTS—(—esquema_índice—,—nombre_índice—,——————►  
►—entorno_local_mensaje—,—mensaje—)—————►◄
```

El esquema es SYSPROC.

Parámetros del procedimiento

esquema_índice

Argumento de entrada de tipo VARCHAR(128) que especifica el esquema del índice de búsqueda de texto. El *esquema_índice* debe seguir las restricciones sobre nombres para los nombres de esquema de DB2. Si el argumento es nulo o una serie vacía, se utilizará el valor de CURRENT_SCHEMA. El *esquema-índice* es sensible a mayúsculas y minúsculas.

nombre-índice

Argumento de entrada de tipo VARCHAR(128) que especifica el nombre del índice. Junto con *esquema_índice*, identifica de modo exclusivo un índice de búsqueda de texto en una base de datos. El *nombre-índice* es sensible a mayúsculas y minúsculas.

entorno_local_mensaje

Argumento de entrada de tipo VARCHAR(33) que especifica el entorno local que ha de utilizarse para los mensajes de error devueltos. Si el argumento es nulo o una serie vacía o los archivos de mensajes del entorno local especificado no están disponibles en el servidor, se utiliza 'en_US'.

mensaje

Argumento de salida de tipo VARCHAR(32K) que especifica un aviso o mensaje informativo para una operación completada de modo satisfactorio.

Autorización

El ID de autorización de la sentencia debe tener al menos uno de los privilegios siguientes:

- El privilegio CONTROL en la tabla en la que está definido el índice
- Autorización DBADM

Ejemplos

Ejemplo 1: En el ejemplo siguiente, se llama a SYSTS_CLEAR_EVENTS para un índice de búsqueda de texto que se ha creado con *esquema_índice* 'db2ts' y *nombre_índice* 'myTextIndex'. Los mensajes de error han de devolverse en inglés. Cuando el procedimiento resulta satisfactorio, se devuelve al que llama el mensaje de parámetro de salida que indica que la operación ha resultado satisfactoria.

```
CALL
SYSPROC.SYSTS_CLEAR_EVENTS('db2ts', 'myTextIndex', 'en_US', ?)
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
Valor parámetros de salida
-----
Nombre de parámetro : MESSAGE
Valor de parámetro  : La operación ha finalizado satisfactoriamente.
```

```
Estado de devolución = 0
```

Ejemplo 2: En el ejemplo siguiente, se llama a SYSTS_CLEAR_EVENTS para borrar las entradas de tabla de sucesos para un índice de búsqueda de texto con *esquema_índice* 'db2ts' y *nombre_índice* 'myTextIndex'. Este índice no existe y da como resultado un error.

```
CALL
SYSPROC.SYSTS_CLEAR_EVENTS('db2ts', 'myTextIndex', 'en_US', ?)
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
SQL20427N Se ha producido un error durante un procedimiento o mandato de
administración de búsqueda de texto. El mensaje de error es "CIE00316 El índice de
búsqueda de texto "db2ts"."myTextIndex" no existe. ". SQLSTATE 38H14
```

Notas de uso

- Los procedimientos de administración de búsqueda de texto utilizan una conexión existente con la base de datos. La transacción actual puede confirmarse o retrotraerse en función de que se completen los procedimientos. Como tal, es posible que desee confirmar todos los cambios de transacciones para evitar cualquier impacto inesperado de dicha confirmación o retroacción. Una forma de conseguirlo es la de activar AUTOCOMMIT.
- Varios procedimientos o mandatos no pueden ejecutarse de modo simultáneo en un índice de búsqueda de texto en el caso de que puedan entrar en conflicto. Algunos de los ejemplos de mandatos y procedimientos conflictivos son:
 - Procedimiento SYSTS_ALTER o mandato ALTER INDEX db2ts
 - Procedimiento SYSTS_DISABLE o mandato DISABLE DATABASE FOR TEXT db2ts

- Procedimiento SYSTS_DROP o mandato DROP INDEX db2ts
- Mandato STOP FOR TEXT db2ts
- Procedimiento SYSTS_UPDATE o mandato UPDATE INDEX db2ts

Si se produce un conflicto, el procedimiento devuelve un SQLCODE -20426 y SQLSTATE 38H13.

- Cuando se planifican actualizaciones regulares (consulte las opciones UPDATE FREQUENCY en los procedimientos SYSTS_CREATE o SYSTS_ALTER), la tabla de sucesos debería comprobarse con regularidad.
- Para limpiar la tabla de sucesos de Búsqueda de texto DB2 para un índice de búsqueda de texto, utilice el procedimiento SYSTS_CLEAR_EVENTS o el mandato CLEAR EVENTS FOR INDEX db2ts una vez haya comprobado el motivo del suceso y eliminado el origen del error.
- Asegúrese de que se hayan efectuado los cambios en todas las filas a las que hace referencia la tabla de sucesos. Al cambiar las filas en la tabla del usuario, se asegura de que al volver a ejecutar el procedimiento SYSTS_UPDATE o el mandato UPDATE INDEX db2ts, se volverá a efectuar un intento de indexar los documentos erróneos.
- Cuando se emite este mandato, se borra la tabla de sucesos.

Procedimiento SYSTS_CREATE - Crear un índice de búsqueda de texto en una columna

Este procedimiento crea un índice de búsqueda de textos para una columna de texto que permite que se efectúen búsquedas en los datos de la columna utilizando las funciones de búsqueda de texto.

Una vez que se cree el índice de búsqueda de texto, pueden efectuarse búsquedas en la columna utilizando funciones de búsqueda de texto en consultas. El índice no contendrá ningún dato hasta que el usuario haya explícitamente ejecutado el procedimiento SYSTS_UPDATE o el mandato UPDATE INDEX de búsqueda de texto o hasta que los servicios de nivel de instancia de búsqueda de texto los hayan ejecutado implícitamente, con arreglo a la frecuencia de actualización definida para el índice.

El procedimiento emite un mandato de administración de búsqueda de texto CREATE INDEX en el servidor de base de datos.

Sintaxis

```

▶▶ SYSTS_CREATE (—esquema_índice—, —nombre_índice—, —
▶ | fuente de texto |, —opciones | —
▶, —entorno_local_mensaje—, —mensaje—)

```

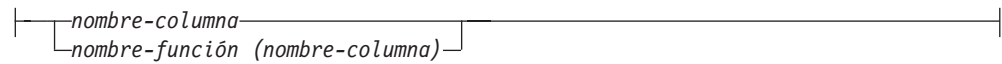
fuentes de texto:

```

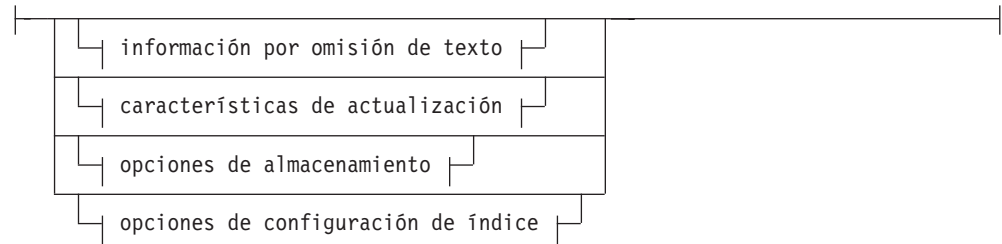
|—nombre-tabla— (— | nombre de columna de texto | —), —

```

nombre de columna de texto:



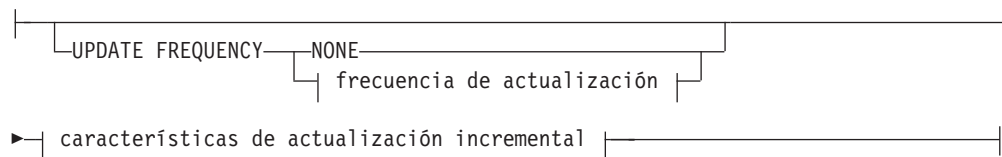
options:



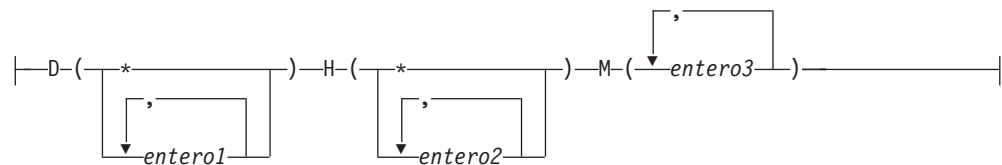
información por omisión de texto:



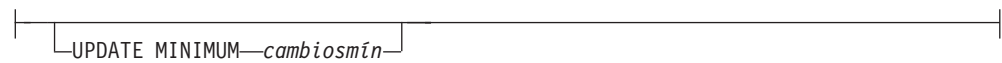
Características de actualización:



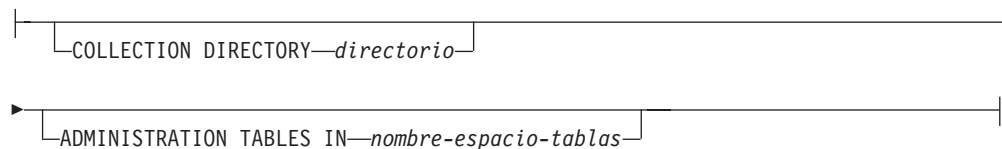
frecuencia de actualización:



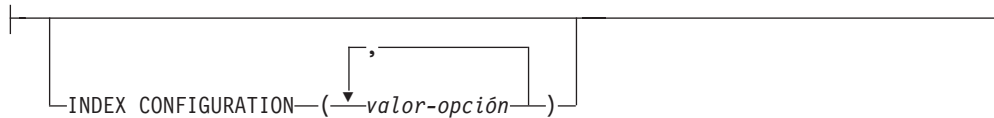
características de actualización incremental:



opciones de almacenamiento:



opciones de configuración de índice:



El esquema es SYSPROC.

Parámetros del procedimiento

esquema_índice

Argumento de entrada de tipo VARCHAR(128) que especifica el esquema del índice de búsqueda de texto. El *esquema_índice* debe seguir las restricciones sobre nombres para los nombres de esquema de DB2. Si el argumento es nulo o una serie vacía, se utilizará el valor de CURRENT SCHEMA. El *esquema-índice* es sensible a mayúsculas y minúsculas.

nombre-índice

Argumento de entrada de tipo VARCHAR(128) que especifica el nombre del índice. Junto con *esquema_índice*, identifica de modo exclusivo un índice de búsqueda de texto en una base de datos. El *nombre-índice* es sensible a mayúsculas y minúsculas.

fFuente de texto

Argumento de entrada de tipo VARCHAR (1024) que especifica el nombre de la columna que ha de indexarse. Las opciones son:

nombre-tabla

El nombre de la tabla que contiene la columna de texto. Los índices de búsqueda de texto no se pueden crear en las tablas siguientes:

- tablas particionadas por rangos
- tablas federadas
- tablas de consulta materializada
- vistas

El *nombre-tabla* es sensible a mayúsculas y minúsculas.

nombre de columna de texto

El nombre de la columna que se debe indexar.

nombre-columna

La columna debe tener uno de los siguientes tipos de datos: CHAR, VARCHAR, LONG VARCHAR, CLOB, DBCLOB, BLOB, GRAPHIC, VARGRAPHIC, LONG VARGRAPHIC o XML. Si el tipo de datos de la columna no es ninguno de ellos, utilice una función de transformación que se haya especificada con *esquema-función.nombre-función* para convertir el tipo de columna a uno de los tipos válidos. Consulte la sintaxis y los detalles en *nombre-función (nombre-columna)*.

Alternativamente, puede especificar una función externa definida por el usuario que acceda a los documentos de texto que deben indexarse. Sólo puede crearse un único índice de búsqueda de texto para una columna. El *nombre-columna* es sensible a las mayúsculas y minúsculas.

nombre-función (nombre-columna)

Especifica el nombre calificado mediante esquema, conforme a los convenios de denominación de DB2, de una función escalar externa que acceda a los documentos de texto de una columna que no tenga el

tipo soportado para la búsqueda de textos. Realiza una conversión del tipo de datos de dicho valor y devuelve el valor como uno de los tipos de datos soportados para la búsqueda de textos. Su tarea es realizar una conversión del tipo de columna. Esta función debe tomar un único parámetro y devolver un único valor. El *nombre-función* (*nombre-columna*) es sensible a las mayúsculas y minúsculas.

opciones

Argumento de entrada de tipo VARCHAR(32K) que especifica las opciones que han de utilizarse. Si no se necesitan opciones, el argumento puede ser un nulo o una serie vacía. Las opciones disponibles son:

CODEPAGE *página-códigos*

Especifica la página de códigos (CODEPAGE) de DB2 que se debe utilizar al indexar documentos de texto. El valor por omisión se especifica mediante el valor de la vista SYSIBMTS.TSDEFAULTS, donde DEFAULTNAME='CODEPAGE' (que resulta ser la página de códigos de la base de datos). Este argumento sólo se aplica a los tipos de datos binarios, por ejemplo, el tipo de columna o el tipo de retorno de una función de transformación debe ser BLOB o del tipo de caracteres FOR BIT DATA.

LANGUAGE *lenguaje*

Especifica el idioma que la búsqueda de texto de DB2 debe utilizar para el proceso específico de un documento durante la indexación. Si no especifica un entorno local, el territorio de la base de datos se utilizará para determinar el valor por omisión de LANGUAGE. Si desea que sus documentos se exploren automáticamente para determinar el entorno local, especifique el entorno local como AUTO.

FORMAT *formato*

Especifica el formato de los documentos de texto de la columna. Los formatos soportados incluyen: TEXT, XML y HTML. La búsqueda de texto de DB2 necesita esta información cuando se indexan documentos. Si no se especifica el formato, se utiliza el valor por omisión. El valor por omisión está en la vista SYSIBMTS.TSDEFAULTS, donde DEFAULTNAME='FORMAT'. Para las columnas de tipo de datos XML, se utiliza el formato por omisión 'XML', sin tener en cuenta el valor de DEFAULTNAME.

UPDATE FREQUENCY

Especifica la frecuencia con la que se realizan las actualizaciones de índice. El índice se actualizará, si el número de cambios es como mínimo el valor establecido para UPDATE MINIMUM. La frecuencia de actualización NONE indica que no se efectuará ninguna actualización de índices adicional. Esto puede ser útil para una columna de texto de una tabla con datos que no cambiarán. También es útil cuando el usuario intenta actualizar manualmente el índice (utilizando el mandato UPDATE INDEX). Las actualizaciones automáticas sólo se pueden realizar si se ha ejecutado el mandato START FOR TEXT y se ejecutan los servicios de la instancia de búsqueda de texto de DB2.

El valor de la frecuencia por omisión se toma de la vista SYSIBMTS.TSDEFAULTS, donde DEFAULTNAME='UPDATEFREQUENCY'.

NONE

No se realizan más actualizaciones de índice. La actualización se debe iniciar manualmente.

D El día o días de la semana en los que se actualiza el índice.

* Cada día de la semana.

entero1

Días específicos de la semana, de domingo a sábado: de 0 a 6

H La hora u horas del día o días especificados en que se actualiza el índice.

* Cada hora del día.

entero2

Horas específicas del día, desde medianoche a las 11 de la noche: de 0 a 23

M El minuto o minutos de la hora u horas especificadas en que se actualiza el índice.

entero3

Se especifica además de la hora (0), o en múltiplos de incrementos de 5 minutos después de la hora: 0, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50 ó 55

UPDATE MINIMUM *cambiosmín*

Especifica el número mínimo de cambios en los documentos de texto antes de que el índice se actualice de manera incremental a la hora especificada en UPDATE FREQUENCY. Sólo están permitidos valores enteros positivos. El valor por omisión se toma de la vista SYSIBMTS.TSDEFAULTS, donde DEFAULTNAME='UPDATEMINIMUM'.

Nota: Este valor se pasa por alto durante un mandato UPDATE INDEX (a menos que se utilice la opción USING UPDATE MINIMUM allí). Un valor pequeño aumenta la coherencia entre la columna de tabla y el índice de búsqueda de texto. Sin embargo, también produce una actividad general superior del rendimiento.

COLLECTION DIRECTORY *directorio*

El directorio en el que se almacena el índice de búsqueda de texto. Por omisión, los datos de recopilación se ubicarán en *DBPATH/NODExxxx/SQLxxxx/db2collections/identificador índice/data*. Debe especificar la vía de acceso absoluta. La longitud máxima del nombre de vía de acceso absoluta es 215 caracteres.

ADMINISTRATION TABLES IN *nombre-espacio-tablas*

Especifica el nombre de un espacio de tablas regular existente para las tablas de administración creadas para el índice. Si no se especifica, se utiliza el espacio de tablas de la tabla base para la que se crea el índice.

INDEX CONFIGURATION *(valor-opción)*

Especifica valores adicionales relacionados con el índice como pares de serie de opción y valor. Estos valores deben especificarse entre comillas simples.

Nota: Un carácter de comillas simples dentro de un valor de serie debe representarse mediante dos comillas simples seguidas. Están soportados los valores siguientes:

Tabla 201. Especificaciones para opción-valor

| Opción | Valores permitidos (por omisión) | Significado |
|---------|---------------------------------------|---|
| COMMENT | Valor de serie de menos de 512 bytes. | Añade un valor de comentario de serie a la columna REMARKS en la vista TSINDEXES del catálogo de búsqueda de texto de DB2. También añade el valor de comentario de serie como la descripción de la colección. |

Ejemplo:

INDEX CONFIGURATION (COMMENT 'Index on User''s Guide column')

entorno_local_mensaje

Argumento de entrada de tipo VARCHAR(33) que especifica el entorno local que ha de utilizarse para los mensajes de error devueltos. Si el argumento es nulo o una serie vacía o los archivos de mensajes del entorno local especificado no están disponibles en el servidor, se utiliza 'en_US'.

mensaje

Argumento de salida de tipo VARCHAR(32K) que especifica un aviso o mensaje informativo para una operación completada de modo satisfactorio.

Autorización

El ID de autorización de la sentencia debe tener al menos uno de los privilegios siguientes:

Uno de los siguientes:

- El privilegio CONTROL en la tabla en la que está definido el índice
- El privilegio INDEX en la tabla en la que está definido el índice y uno de los siguientes:
 - Autorización IMPLICIT_SCHEMA en la base de datos, si el nombre de esquema implícito o explícito del índice no existe
 - Privilegio CREATEIN para el esquema, si el nombre de esquema del índice hace referencia a un esquema existente
- Autorización DBADM

Ejemplos

Ejemplo 1: En el ejemplo siguiente, se crea un índice de búsqueda de texto *esquema_índice* 'db2ts' y *nombre_índice* 'myTextIndex' utilizando el procedimiento SYSTS_CREATE. La opción 'UPDATE MINIMUM 10' especifica que al menos deberían efectuarse 10 cambios en los documentos de texto asociados con el índice antes de efectuar una actualización incremental del índice. Los mensajes de error han de devolverse en inglés. Cuando el mandato de búsqueda de texto subyacente se ejecuta de modo satisfactorio, se establece el parámetro de salida message para indicar el estado de la ejecución del mandato.

```
CALL SYSPROC.SYSTS_CREATE('db2ts', 'myTextIndex',
    'myUserSchema.myBaseTable (myTextColumn)', 'UPDATE MINIMUM 10',
    'en_US', ?)
```

El ejemplo siguiente muestra los datos de salida de la consulta.

Valor parámetros de salida

```
-----  
Nombre de parámetro : MESSAGE  
Valor de parámetro  : La operación ha finalizado satisfactoriamente.  
Estado de devolución = 0
```

Ejemplo 2: En el ejemplo siguiente, se llama a SYSTS_CREATE para crear un índice de búsqueda de texto con *esquema_índice* 'db2ts' y *nombre_índice* 'myTextIndex'. No se especifica ninguna opción. En este ejemplo, el índice ya existe lo que da como resultado que se devuelva un mensaje de error al llamador.

```
CALL SYSPROC.SYSTS_CREATE('db2ts', 'myTextIndex',  
    'myUserSchema.myBaseTable (myTextColumn)', '', 'en_US', ?)
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
SQL20427N Se ha producido un error durante un procedimiento o mandato de  
administración de búsqueda de texto. El mensaje de error es "CIE00201 El índice  
de búsqueda de texto "db2ts"."myTextIndex" ya existe."
```

Notas de uso

- Los procedimientos de administración de búsqueda de texto utilizan una conexión existente con la base de datos. La transacción actual puede confirmarse o retrotraerse en función de que se completen los procedimientos. Como tal, es posible que desee confirmar todos los cambios de transacciones para evitar cualquier impacto inesperado de dicha confirmación o retroacción. Una forma de conseguirlo es la de activar AUTOCOMMIT.
- Varios procedimientos o mandatos no pueden ejecutarse de modo simultáneo en un índice de búsqueda de texto en el caso de que puedan entrar en conflicto. Algunos de los ejemplos de mandatos y procedimientos conflictivos son:
 - Procedimiento SYSTS_ALTER o mandato ALTER INDEX db2ts
 - Procedimiento SYSTS_CLEAR_EVENTS o mandato CLEAR EVENTS FOR INDEX db2ts
 - Procedimiento SYSTS_DISABLE o mandato DISABLE DATABASE FOR TEXT db2ts
 - Mandato STOP FOR TEXT db2ts
 - Procedimiento SYSTS_UPDATE o mandato UPDATE INDEX db2ts

Si se produce un conflicto, el procedimiento devuelve un SQLCODE -20426 y SQLSTATE 38H13.

- Con la ejecución satisfactoria del mandato CREATE INDEX:
 - Se actualizan datos del servidor de búsqueda de texto de DB2. Se crea una colección con el nombre *instancia_nombre-basedatos_identificador-índice_número*, como en el ejemplo siguiente:
tigertail_MYTSDB_TS250517_0000

El nombre de colección se puede recuperar de la vista SYSIBMTS.TSCOLLECTIONNAMES (columna COLLECTIONNAME).

- Se actualiza la información del catálogo de la búsqueda de texto de DB2. Se crea una tabla de base de índice en el espacio de tablas especificado con los índices de DB2 adecuados. Además, se crea una tabla de sucesos de índice en el espacio de tablas especificado.
- El índice de búsqueda de texto que se acaba de crear no se llena automáticamente. El procedimiento SYSTS_UPDATE o el mandato UPDATE INDEX deben ejecutarse manualmente o automáticamente (como consecuencia de que se haya definido una planificación de actualización para

el índice por medio de la especificación de la opción UPDATE FREQUENCY) para que se llene el índice de búsqueda de texto.

- Se actualiza el archivo de datos de índice de búsqueda de texto del servidor de bases de datos de DB2. Se registra la información de actualización planificada para cada índice de la instancia.

Restricciones de uso:

- Se debe definir una clave primaria para la tabla. En la búsqueda de texto de DB2, se puede utilizar una clave primaria de DB2 de múltiples columnas sin limitaciones de tipo. El número de columnas de claves primarias se limita a 2 menos que el número de columnas de claves primarias permitidas por DB2.
- La longitud total de todas las columnas de claves primarias para una tabla con índices de búsqueda de texto de DB2 se limita a 15 bytes menos que la longitud total de claves primarias permitida por DB2. Consulte las restricciones de DB2 de la sentencia DB2 CREATE INDEX.

Procedimiento SYSTS_DISABLE - Inhabilitar base de datos actual para búsqueda de textos

El procedimiento inhabilita la búsqueda de texto de DB2 para la base de datos actual.

Una vez que se haya inhabilitado la característica de Búsqueda de texto, los mandatos e índices de búsqueda de texto ya no estarán disponibles para su utilización con la base de datos.

El procedimiento emite un mandato de administración de búsqueda de texto DISABLE DATABASE FOR TEXT en el servidor de base de datos.

Sintaxis

►►SYSTS_DISABLE(—*opciones*—, —*entorno_local_mensaje*—, —*mensaje*—)◀◀

El esquema es SYSPROC.

Parámetros del procedimiento

opciones

Argumento de entrada de tipo VARCHAR(128) que especifica las opciones que han de utilizarse al inhabilitar la base de datos. El argumento puede establecerse en FORCE. Cuando se especifica este valor, se descartan todos los índices y se fuerza la inhabilitación de la característica de Búsqueda de texto. No se conserva ningún índice de búsqueda y no se devuelve ningún aviso o mensaje de error. Si el argumento es nulo o una serie vacía, se efectúa un intento de inhabilitar la característica de la Búsqueda de texto para la base de datos.

entorno_local_mensaje

Argumento de entrada de tipo VARCHAR(33) que especifica el entorno local que ha de utilizarse para los mensajes de error devueltos. Si el argumento es nulo o una serie vacía o los archivos de mensajes del entorno local especificado no están disponibles en el servidor, se utiliza 'en_US'.

mensaje

Argumento de salida de tipo VARCHAR(32K) que especifica un aviso o mensaje informativo para una operación completada de modo satisfactorio.

Autorización

Los privilegios que tiene el ID de autorización de la sentencia deben incluir la autorización DBADM.

Ejemplos

Ejemplo 1: En el ejemplo siguiente, la Búsqueda de texto se ha inhabilitado para una base de datos utilizando el procedimiento SYSTS_DISABLE. La opción FORCE se especifica para asegurar la inhabilitación de la característica incluso en el caso de que sigan existiendo índices de búsqueda de texto en las tablas de la base de datos. Se ha especificado que los mensajes de error han de devolverse en inglés. El parámetro de salida *mensaje* se ha establecido en una serie de mensajes informativos.

```
CALL SYSPROC.SYSTS_DISABLE('FORCE', 'en_US', ?)
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
Valor parámetros de salida
```

```
-----
```

```
Nombre de parámetro : MESSAGE
```

```
Valor de parámetro  : La operación ha finalizado satisfactoriamente.
```

```
Estado de devolución = 0
```

Ejemplo 2: En el ejemplo siguiente, la Búsqueda de texto se ha inhabilitado para una base de datos con los índices de búsqueda de texto existentes utilizando el procedimiento SYSTS_DISABLE sin especificar la opción FORCE. Esto da como resultado un mensaje de error al llamador. Resulta preferible descartar todos los índices de búsqueda de texto existentes antes de inhabilitar la característica de Búsqueda de texto o alternativamente especificar la opción FORCE para el valor de parámetro de entrada *opciones*.

```
CALL SYSPROC.SYSTS_DISABLE('', 'en_US', ?)
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
SQL20427N Se ha producido un error durante un procedimiento o mandato de
administración de búsqueda de texto. El mensaje de error es "C1E00326 El
índice de búsqueda de texto está activo en la base de datos especificada
o en la predeterminada.". SQLSTATE 38H14
```

Notas de uso

- Los procedimientos de administración de búsqueda de texto utilizan una conexión existente con la base de datos. La transacción actual puede confirmarse o retrotraerse en función de que se completen los procedimientos. Como tal, es posible que desee confirmar todos los cambios de transacciones para evitar cualquier impacto inesperado de dicha confirmación o retroacción. Una forma de conseguirlo es la de activar AUTOCOMMIT.
- Varios procedimientos o mandatos no pueden ejecutarse de modo simultáneo en un índice de búsqueda de texto en el caso de que puedan entrar en conflicto. Algunos de los ejemplos de mandatos y procedimientos conflictivos son:
 - Procedimiento SYSTS_ALTER o mandato ALTER INDEX db2ts

- Procedimiento SYSTS_CLEAR_EVENTS o mandato CLEAR EVENTS FOR INDEX db2ts
- Procedimiento SYSTS_DISABLE o mandato DISABLE DATABASE FOR TEXT db2ts
- Mandato STOP FOR TEXT db2ts
- Procedimiento SYSTS_UPDATE o mandato UPDATE INDEX db2ts

Si se produce un conflicto, el procedimiento devuelve un SQLCODE -20426 y SQLSTATE 38H13.

- Cuando se ejecuta este procedimiento,
 - se actualiza la información del catálogo de Búsqueda de texto de DB2. Se descartan las tablas de suceso y registro del índice. Se suprimen los activadores de la tabla de texto del usuario.
 - si se especifica la opción FORCE, se elimina toda la información de índice de texto de la base de datos y se suprimen todas las recopilaciones asociadas. Además, se actualiza el servicio de texto para eliminar cualquier información de planificación de actualización que quede. Consulte el "Mandato db2ts DROP INDEX" o el "Procedimiento SYSTS_DROP" para más información.
- Este procedimiento no influye en el estado de habilitación del DB2 Net Search Extender de la base de datos. Suprime las vistas y tablas de catálogos de la Búsqueda de texto de DB2 creadas mediante el procedimiento SYSTS_ENABLE o el mandato ENABLE FOR TEXT.
- Antes de descartar una base de datos DB2 que tenga definiciones de índice de búsqueda de texto, ejecute este procedimiento y asegúrese de que las recopilaciones e índices de texto se hayan eliminado de modo satisfactorio.
- Si no se han podido suprimir algunos índices utilizando la opción FORCE, los nombres de recopilación se grabarán en los archivos de anotaciones cronológicas db2diag. Si no se ejecutan el procedimiento de índice de búsqueda de texto SYSTS_DISABLE o el mandato DISABLE DATABASE FOR TEXT antes que el mandato CLP DROP DATABASE, los servicios de índice de búsqueda de texto también deberán limpiarse utilizando el mandato CLEANUP FOR TEXT. Consulte el procedimiento SYSTS_DROP o el mandato DROP INDEX para obtener más información sobre el modo de descartar índices y el mandato CLEANUP FOR TEXT para obtener más información sobre las recopilaciones de búsqueda de texto y su relación con los índices de búsqueda de texto.

Nota: Al usuario no se le aconseja utilizar dichos resultados en recopilaciones huérfanas, es decir, en recopilaciones que sigan definidas en el servidor de búsqueda de texto pero que no sean utilizadas por DB2. A continuación se describen algunos casos que pueden ocasionar recopilaciones huérfanas:

- Cuando un mandato CLP DROP DATABASE o una sentencia DROP TABLE se ejecuta sin ejecutar el procedimiento SYSTS_DISABLE o un mandato DISABLE DATABASE FOR TEXT.
- Cuando se ejecuta el procedimiento SYSTS_DISABLE o se ejecuta un mandato DISABLE DATABASE FOR TEXT sin utilizar la opción FORCE.
- Algunas condiciones de error adicionales. El mandato CLEANUP FOR TEXT puede utilizarse en algunos escenarios.

Procedimiento SYSTS_DROP - Descartar un índice de búsqueda de texto

Este procedimiento descarta un índice de búsqueda de texto existente asociado a cualquier columna de tabla.

Después de ejecutar este procedimiento de forma satisfactoria, las consultas de búsqueda de texto no podrán ejecutarse en la columna.

El procedimiento emite un mandato de administración de búsqueda de texto DROP INDEX en el servidor de base de datos.

Sintaxis

```
►►—SYSTS_DROP—(—esquema_índice—,—nombre_índice—,——————►  
►—entorno_local_mensaje—,—mensaje—)—————►◄
```

El esquema es SYSPROC.

Parámetros del procedimiento

esquema_índice

Argumento de entrada de tipo VARCHAR(128) que especifica el esquema del índice de búsqueda de texto. El *esquema_índice* debe seguir las restricciones sobre nombres para los nombres de esquema de DB2. Si el argumento es nulo o una serie vacía, se utilizará el valor de CURRENT SCHEMA. El *esquema-índice* es sensible a mayúsculas y minúsculas.

nombre-índice

Argumento de entrada de tipo VARCHAR(128) que especifica el nombre del índice. Junto con *esquema_índice*, identifica de modo exclusivo un índice de búsqueda de texto en una base de datos. El *nombre-índice* es sensible a mayúsculas y minúsculas.

entorno_local_mensaje

Argumento de entrada de tipo VARCHAR(33) que especifica el entorno local que ha de utilizarse para los mensajes de error devueltos. Si el argumento es nulo o una serie vacía o los archivos de mensajes del entorno local especificado no están disponibles en el servidor, se utiliza 'en_US'.

mensaje

Argumento de salida de tipo VARCHAR(32K) que especifica un aviso o mensaje informativo para una operación completada de modo satisfactorio.

Autorización

El ID de autorización de la sentencia debe tener al menos uno de los privilegios siguientes:

- El privilegio CONTROL en la tabla en la que está definido el índice
- Autorización DBADM

Ejemplos

Ejemplo 1: En el ejemplo siguiente, se descarta un índice de búsqueda de texto creado con *esquema_índice* 'db2ts' y *nombre_índice* 'myTextIndex'. Los mensajes de error han de devolverse en inglés. Cuando el procedimiento resulta satisfactorio, se devuelve al que llama el mensaje de parámetro de salida que indica que la operación ha resultado satisfactoria.

```
CALL SYSPROC.SYSTS_DROP('db2ts', 'myTextIndex', 'en_US', ?)
```

El ejemplo siguiente muestra los datos de salida de la consulta.

Valor parámetros de salida

Nombre de parámetro : MESSAGE
Valor de parámetro : La operación ha finalizado satisfactoriamente.

Estado de devolución = 0

Ejemplo 2: En el ejemplo siguiente, se llama a SYSTS_DROP para descartar un índice de búsqueda de texto con *esquema_índice* 'db2ts' y *nombre_índice* 'myTextIndex'. Este índice no existe y da como resultado un error.

```
CALL SYSPROC.SYSTS_DROP('db2ts', 'myTextIndex', 'en_US', ?)
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
SQL20427N Se ha producido un error durante un procedimiento o mandato de
administración de búsqueda de texto. El mensaje de error es "CIE00316 El índice de
búsqueda de texto "db2ts"."myTextIndex" no existe. ". SQLSTATE 38H14
```

Notas de uso

- Los procedimientos de administración de búsqueda de texto utilizan una conexión existente con la base de datos. La transacción actual puede confirmarse o retrotraerse en función de que se completen los procedimientos. Como tal, es posible que desee confirmar todos los cambios de transacciones para evitar cualquier impacto inesperado de dicha confirmación o retroacción. Una forma de conseguirlo es la de activar AUTOCOMMIT.
- Varios procedimientos o mandatos no pueden ejecutarse de modo simultáneo en un índice de búsqueda de texto en el caso de que puedan entrar en conflicto. Algunos de los ejemplos de mandatos y procedimientos conflictivos son:
 - Procedimiento SYSTS_ALTER o mandato ALTER INDEX db2ts
 - Procedimiento SYSTS_CLEAR_EVENTS o mandato CLEAR EVENTS FOR INDEX db2ts
 - Procedimiento SYSTS_DISABLE o mandato DISABLE DATABASE FOR TEXT db2ts
 - Mandato STOP FOR TEXT db2ts
 - Procedimiento SYSTS_UPDATE o mandato UPDATE INDEX db2ts

Si se produce un conflicto, el procedimiento devuelve un SQLCODE -20426 y SQLSTATE 38H13.

- Descartar la tabla de usuario en DB2 no desencadena que se descarten los índices, éstos deberán descartarse manualmente antes o después de descartar la tabla.
- Cuando se ejecuta este procedimiento,
 - se actualiza la información del catálogo de búsqueda de texto. Se descartan las tablas de suceso y etapas del índice. Se suprimen los activadores de la tabla del usuario.
 - se suprime la entrada de índice del archivo de datos de índice de búsqueda de texto. El archivo contiene una representación permanente de las planificaciones de actualización (también las vacías) para cada uno de los índices de la instancia.
 - se elimina la recopilación asociada con la definición de índice de búsqueda de texto.
- Si, después de descartar un índice de búsqueda de texto, piensa crear uno nuevo en la misma columna de texto, en primer lugar deberá volverse a conectar con la base de datos antes de crear el nuevo índice de texto.

Procedimiento SYSTS_ENABLE - Habilitar base de datos actual para búsqueda de texto

Este procedimiento habilita la búsqueda de texto de DB2 para la base de datos actual.

Este procedimiento debe emitirse de modo satisfactorio antes de que puedan crearse índices de búsqueda de texto en las columnas de las tablas de la base de datos.

Este procedimiento emite el mandato de administración de búsqueda de texto ENABLE DATABASE FOR TEXT en el servidor de base de datos.

Sintaxis

```
►►—SYSTS_ENABLE—(—entorno_local_mensaje—,—mensaje—)—————►◄
```

El esquema es SYSPROC.

Parámetros del procedimiento

entorno_local_mensaje

Argumento de entrada de tipo VARCHAR(33) que especifica el entorno local que ha de utilizarse para los mensajes de error devueltos. Si el argumento es nulo o una serie vacía o los archivos de mensajes del entorno local especificado no están disponibles en el servidor, se utiliza 'en_US'.

mensaje

Argumento de salida de tipo VARCHAR(32K) que especifica un aviso o mensaje informativo para una operación completada de modo satisfactorio.

Autorización

El usuario debe tener el privilegio DBADM para ejecutar el mandato ENABLE DATABASE.

Ejemplos

Ejemplo 1: Habilitar la base de datos para la búsqueda de texto y devolver los mensajes de error en inglés.

```
CALL SYSPROC.SYSTS_ENABLE('en_US', ?)
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
Valor parámetros de salida
```

```
-----
```

```
Nombre de parámetro : MESSAGE
```

```
Valor de parámetro  : La operación ha finalizado satisfactoriamente.
```

```
Estado de devolución = 0
```

Ejemplo 2: En el ejemplo siguiente, se llama a SYSTS_ENABLE en una base de datos que ya ha sido habilitada para la búsqueda de texto. Esto da como resultado un mensaje de error al llamador.

```
CALL SYSPROC.SYSTS_ENABLE('en_US', ?)
```

El ejemplo siguiente muestra los datos de salida de la consulta.

SQL20427N Se ha producido un error durante un procedimiento o mandato de administración de búsqueda de texto. El mensaje de error del producto de búsqueda de texto es "CIE00322 La base de datos especificada o la predeterminada ya está habilitada para el texto.". SQLSTATE 38H14

Notas de uso

- Los procedimientos de administración de búsqueda de texto utilizan una conexión existente con la base de datos. La transacción actual puede confirmarse o retrotraerse en función de que se completen los procedimientos. Como tal, es posible que desee confirmar todos los cambios de transacciones para evitar cualquier impacto inesperado de dicha confirmación o retrotracción. Una forma de conseguirlo es la de activar AUTOCOMMIT.
- Cuando se ejecuta este procedimiento,
 - Este procedimiento crea objetos de base de datos, como por ejemplo vistas y tablas de catálogos de administración de búsqueda de texto, en el esquema SYSIBMTS. Estos objetos se colocan en el espacio de tablas por omisión de la base de datos (IBMDEFAULTGROUP).
 - las bases de datos establecidas por omisión para el índice de búsqueda de texto están disponibles en la vista SYSIBMTS.TSDEFAULTS.
 - y cuando el mandato se ha completado satisfactoriamente, se crean y están disponibles las vistas y tablas de catálogo de búsqueda de texto.

Procedimiento SYSTS_UPDATE - Actualizar el índice de búsqueda de texto

Este procedimiento actualiza el índice de búsqueda de textos para reflejar el contenido actual de la columna de texto a la que está asociado el índice.

Mientras se realiza la actualización, es posible efectuar una búsqueda. Hasta que finalice la actualización, la búsqueda trabaja sobre un índice parcialmente actualizado.

El procedimiento emite un mandato de administración de búsqueda de texto UPDATE INDEX en el servidor de base de datos.

Sintaxis

```
►►SYSTS_UPDATE(—esquema_índice—,—nombre_índice—,——————►  
►—opciones_actualización—,—entorno_local_mensaje—,—mensaje—)—————►
```

El esquema es SYSPROC.

Parámetros del procedimiento

esquema_índice

Argumento de entrada de tipo VARCHAR(128) que especifica el esquema del índice de búsqueda de texto. El *esquema_índice* debe seguir las restricciones sobre nombres para los nombres de esquema de DB2. Si el argumento es nulo o una serie vacía, se utilizará el valor de CURRENT SCHEMA. El *esquema-índice* es sensible a mayúsculas y minúsculas.

nombre-índice

Argumento de entrada de tipo VARCHAR(128) que especifica el nombre del

índice. Junto con *esquema_índice*, identifica de modo exclusivo un índice de búsqueda de texto en una base de datos. El *nombre-índice* es sensible a mayúsculas y minúsculas.

opciones-actualización

Argumento de entrada de tipo VARCHAR(32K) que especifica las opciones de actualización. Los valores posibles son:

- USING UPDATE MINIMUM: este valor respeta los valores UPDATE MINIMUM del mandato de administración de búsqueda de texto CREATE INDEX y del procedimiento SYSTS_CREATE.
- NULL o serie vacía (''): la actualización se inicia incondicionalmente cuando se llama el procedimiento.

entorno_local_mensaje

Argumento de entrada de tipo VARCHAR(33) que especifica el entorno local que ha de utilizarse para los mensajes de error devueltos. Si el argumento es nulo o una serie vacía o los archivos de mensajes del entorno local especificado no están disponibles en el servidor, se utiliza 'en_US'.

mensaje

Argumento de salida de tipo VARCHAR(32K) que especifica un aviso o mensaje informativo para una operación completada de modo satisfactorio.

Autorización

El ID de autorización de la sentencia debe tener al menos uno de los privilegios siguientes:

- El privilegio CONTROL en la tabla en la que está definido el índice de texto
- Autorización DATAACCESS

Ejemplos

Ejemplo 1: En el ejemplo siguiente, se actualiza un índice de búsqueda de texto creado con *esquema_índice* 'db2ts' y *nombre_índice* 'myTextIndex'. Un valor NULL en lugar de *opciones_actualización* significa que una actualización se inicia incondicionalmente cuando se llama el procedimiento almacenado. Los mensajes de error han de devolverse en inglés. Cuando el procedimiento resulta satisfactorio, se devuelve al que llama el mensaje de parámetro de salida que indica que la operación ha resultado satisfactoria.

```
CALL SYSPROC.SYSTS_UPDATE('db2ts', 'myTextIndex', '', 'en_US', ?)
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
Valor parámetros de salida
-----
Nombre de parámetro : MESSAGE
Valor de parámetro  : La operación ha finalizado satisfactoriamente.

Estado de devolución = 0
```

Ejemplo 2: En el ejemplo siguiente, se llama a SYSTS_UPDATE para actualizar un índice de búsqueda de texto con *esquema_índice* 'db2ts' y *nombre_índice* 'myTextIndex'. Este índice no existe y da como resultado un error.

```
CALL SYSPROC.SYSTS_UPDATE('db2ts', 'myTextIndex', 'USING UPDATE MINIMUM',
'en_US', ?)
```

El ejemplo siguiente muestra los datos de salida de la consulta.

SQL20427N Se ha producido un error durante un procedimiento o mandato de administración de búsqueda de texto. El mensaje de error es "CIE00316 El índice de búsqueda de texto "db2ts"."myTextIndex" no existe. ". SQLSTATE 38H14

Notas de uso

- Los procedimientos de administración de búsqueda de texto utilizan una conexión existente con la base de datos. La transacción actual puede confirmarse o retrotraerse en función de que se completen los procedimientos. Como tal, es posible que desee confirmar todos los cambios de transacciones para evitar cualquier impacto inesperado de dicha confirmación o retrotracción. Una forma de conseguirlo es la de activar AUTOCOMMIT.
- Varios procedimientos o mandatos no pueden ejecutarse de modo simultáneo en un índice de búsqueda de texto en el caso de que puedan entrar en conflicto. Algunos de los ejemplos de mandatos y procedimientos conflictivos son:
 - Procedimiento SYSTS_ALTER o mandato ALTER INDEX db2ts
 - Procedimiento SYSTS_CLEAR_EVENTS o mandato CLEAR EVENTS FOR INDEX db2ts
 - Procedimiento SYSTS_DISABLE o mandato DISABLE DATABASE FOR TEXT db2ts
 - Procedimiento SYSTS_DROP o mandato DROP INDEX db2ts
 - Mandato STOP FOR TEXT db2ts
 - Procedimiento SYSTS_UPDATE o mandato UPDATE INDEX db2ts

Si se produce un conflicto, el procedimiento devuelve un SQLCODE -20426 y SQLSTATE 38H13.

- Este procedimiento no se devuelve hasta que se completen todos los procesos de actualización del índice. La duración depende del número de documentos que se deben indexar y el número de documentos que ya se han indexado. El nombre de colección para el índice se puede recuperar de la vista SYSIBMTS.TSCOLLECTIONNAMES (columna COLLECTIONNAME).
- Cuando haya errores de documento individuales, los documentos deberán corregirse. Las claves primarias de los documentos erróneos se pueden buscar en la tabla de sucesos para el índice. Cambiando las filas correspondientes en la tabla del usuario, la siguiente llamada a SYSTS_UPDATE volverá a procesar estos documentos.
- Cuando se ejecuta este procedimiento,
 - las filas se insertan en la tabla de sucesos (incluyendo la información de errores del analizador). La información se suprime de la tabla de etapas del índice en caso de actualizaciones incrementales. Antes de la primera actualización, crea activadores en la tabla del usuario.
 - la recopilación se ha actualizado: los documentos nuevos o modificados se actualizan e indexan y los documentos suprimidos se descartan del índice.

Capítulo 19. Rutinas de gestión de cargas de trabajo

WLM_CANCEL_ACTIVITY - Cancelar una actividad

Este procedimiento cancela una actividad determinada. Si la cancelación tiene lugar, se devolverá un mensaje de error a la aplicación que ha enviado la actividad que se ha cancelado.

Sintaxis

```
►►—WLM_CANCEL_ACTIVITY—(—descriptorcontexto_aplicación—,—id_uow—,—id_actividad—)————►◄
```

El esquema es SYSPROC.

Parámetros del procedimiento

descriptorcontexto_aplicación

Argumento de entrada de tipo BIGINT que especifica el descriptor de contexto de aplicación cuya actividad se va a cancelar. Si el argumento es nulo, no se encontrará ninguna actividad y se devolverá SQL4702N con SQLSTATE 5U035.

id_uow

Argumento de entrada de tipo INTEGER que especifica el ID de la unidad de trabajo de la actividad que se va a cancelar. Si el argumento es nulo, no se encontrará ninguna actividad y se devolverá SQL4702N con SQLSTATE 5U035.

id_actividad

Argumento de entrada de tipo INTEGER que especifica el ID de actividad que identifica de forma exclusiva la actividad dentro de la unidad de trabajo que se va a cancelar. Si el argumento es nulo, no se encontrará ninguna actividad y se devolverá SQL4702N con SQLSTATE 5U035.

Autorización

Privilegio EXECUTE para el procedimiento WLM_CANCEL_ACTIVITY

Ejemplo

Un administrador puede utilizar la función de tabla WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES para buscar el descriptor de contexto de aplicación, el ID de unidad de trabajo y el ID de actividad de una actividad. Para cancelar una actividad con el descriptor de contexto de aplicación 1, el ID de unidad de trabajo 2 y el ID de actividad 3:

```
CALL WLM_CANCEL_ACTIVITY(1, 2, 3)
```

Notas de uso

- Si no se encuentra ninguna actividad, se devuelve un SQL4702N con SQLSTATE 5U035.
- Si la actividad no se puede cancelar porque no está en el estado correcto (no inicializada), se devuelve un SQL4703N (código de razón 1) con SQLSTATE 5U016.

- Si la actividad se cancela satisfactoriamente, se devuelve un SQL4725N con SQLSTATE 57014 a la aplicación cancelada.
- Si, en el momento de la cancelación, el coordinador está procesando una solicitud para otra actividad o está desocupado, la actividad se coloca en el estado CANCEL_PENDING y se cancelará cuando el coordinador procese la siguiente solicitud correspondiente a la actividad.

WLM_CAPTURE_ACTIVITY_IN_PROGRESS - Recopilar información para supervisor de sucesos de actividades

El procedimiento WLM_CAPTURE_ACTIVITY_IN_PROGRESS recopila información sobre una actividad especificada y la graba en el supervisor de sucesos de actividades activo.

Cuando se aplica este procedimiento a una actividad con actividades hijo, el procedimiento genera de forma recurrente un registro correspondiente a cada actividad hijo. Esta información se recopila y se envía al invocar el procedimiento; el procedimiento no espera hasta que la actividad padre finalice la ejecución. El registro de la actividad se marca como un registro parcial en el supervisor de sucesos.

Sintaxis

```

▶▶—WLM_CAPTURE_ACTIVITY_IN_PROGRESS—(—descriptorcontexto_aplicación—, —————▶
▶—id_uow—, —id_actividad—)—————▶▶

```

El esquema es SYSPROC.

Parámetros del procedimiento

Si no se especifica ninguno de los parámetros siguientes, no se encuentra ninguna actividad y se devuelve SQL4702N con SQLSTATE 5U035.

descriptorcontexto_aplicación

Argumento de entrada de tipo BIGINT que especifica el descriptor de contexto de aplicación cuya información de actividad se va a capturar.

id_uow

Argumento de entrada de tipo INTEGER que especifica el ID de la unidad de trabajo de la actividad que se va a capturar.

id_actividad

Argumento de entrada de tipo INTEGER que especifica el ID de actividad que identifica de forma exclusiva la actividad dentro de la unidad de trabajo cuya información se va a capturar.

Autorización

Privilegio EXECUTE para el procedimiento
WLM_CAPTURE_ACTIVITY_IN_PROGRESS

Ejemplo

Supongamos que un usuario se queja de que el procedimiento almacenado MYSHEMA.MYSLOWSTP parece estar ejecutándose más lentamente de lo

habitual. El administrador desea investigar la causa de la lenta ejecución. Investigar mientras se está ejecutando el procedimiento almacenado puede resultar poco práctico, así que el administrador decide capturar la información de la actividad del procedimiento almacenado y cualquiera de las actividades anidadas en la misma.

Se ha activado un supervisor de sucesos para las actividades de DB2 llamadas DB2ACTIVITIES. El administrador utiliza la función WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES para buscar el descriptor de contexto de aplicación, el ID de unidad de trabajo y el ID de actividad para la llamada de este procedimiento almacenado. Suponiendo que la actividad se identifica mediante el descriptor de contexto de aplicación 1, el ID de unidad de trabajo 2 y el ID de actividad 3, ahora el administrador puede emitir la llamada a WLM_CAPTURE_ACTIVITY_IN_PROGRESS del siguiente modo:

```
CALL WLM_CAPTURE_ACTIVITY_IN_PROGRESS(1,2,3)
```

Una vez completado el procedimiento, el administrador puede utilizar la siguiente función de tabla para ver dónde ha empleado su tiempo la actividad. La función recupera la información del supervisor de sucesos DB2ACTIVITIES.

```
CREATE FUNCTION SHOWCAPTUREDACTIVITY(APPHNDL BIGINT,
                                     UOWID INTEGER,
                                     ACTIVITYID INTEGER)
  RETURNS TABLE (UOW_ID INTEGER, ACTIVITY_ID INTEGER, STMT_TEXT VARCHAR(40),
                 LIFE_TIME DOUBLE)
  LANGUAGE SQL
  READS SQL DATA
  NO EXTERNAL ACTION
  DETERMINISTIC
  RETURN WITH RAH (LEVEL, APPL_ID, PARENT_UOW_ID, PARENT_ACTIVITY_ID,
                  UOW_ID, ACTIVITY_ID, STMT_TEXT, ACT_EXEC_TIME) AS
  (SELECT 1, ROOT.APPL_ID, ROOT.PARENT_UOW_ID,
          ROOT.PARENT_ACTIVITY_ID, ROOT.UOW_ID, ROOT.ACTIVITY_ID,
          ROOTSTMT.STMT_TEXT, ACT_EXEC_TIME
   FROM ACTIVITY_DB2ACTIVITIES ROOT, ACTIVITYSTMT_DB2ACTIVITIES ROOTSTMT
   WHERE ROOT.APPL_ID = ROOTSTMT.APPL_ID AND ROOT.AGENT_ID = APPHNDL
        AND ROOT.UOW_ID = ROOTSTMT.UOW_ID AND ROOT.UOW_ID = UOWID
        AND ROOT.ACTIVITY_ID = ROOTSTMT.ACTIVITY_ID AND ROOT.ACTIVITY_ID = ACTIVITYID
   UNION ALL
   SELECT PARENT.LEVEL +1, CHILD.APPL_ID, CHILD.PARENT_UOW_ID,
          CHILD.PARENT_ACTIVITY_ID, CHILD.UOW_ID,
          CHILD.ACTIVITY_ID, CHILDSMT.STMT_TEXT, CHILD.ACT_EXEC_TIME
   FROM RAH PARENT, ACTIVITY_DB2ACTIVITIES CHILD,
          ACTIVITYSTMT_DB2ACTIVITIES CHILDSMT
   WHERE PARENT.APPL_ID = CHILD.APPL_ID AND
        CHILD.APPL_ID = CHILDSMT.APPL_ID AND
        PARENT.UOW_ID = CHILD.PARENT_UOW_ID AND
        CHILD.UOW_ID = CHILDSMT.UOW_ID AND
        PARENT.ACTIVITY_ID = CHILD.PARENT_ACTIVITY_ID AND
        CHILD.ACTIVITY_ID = CHILDSMT.ACTIVITY_ID AND
        PARENT.LEVEL < 64
  )
  SELECT UOW_ID, ACTIVITY_ID, SUBSTR(STMT_TEXT,1,40),
         ACT_EXEC_TIME AS
         LIFE_TIME
  FROM RAH
```

La siguiente consulta de ejemplo utiliza la función de tabla:

```
SELECT * FROM TABLE(SHOWCAPTUREDACTIVITY(1, 2, 3))
  AS ACTS ORDER BY UOW_ID, ACTIVITY_ID
```

Notas de uso

Si no hay ningún supervisor de sucesos de actividades activo, se devuelve un SQL1633W con SQLSTATE 01H53.

La información de actividad sólo se recopila en la partición coordinadora para la actividad.

WLM_COLLECT_STATS - Coleccionar y restaurar estadísticas de gestión de carga de trabajo

El procedimiento WLM_COLLECT_STATS reúne estadísticas para clases de servicio, cargas de trabajo, clases de trabajo y colas de umbral y las graba en el supervisor de eventos de estadísticas. El procedimiento también restablece las estadísticas para clases de servicio, cargas de trabajo, clases de trabajo y colas de umbral. Si no hay ningún supervisor de eventos de estadísticas activos, el procedimiento sólo restaura las estadísticas.

Sintaxis

►►—WLM_COLLECT_STATS—(—)—————►►

El esquema es SYSPROC.

Autorización

Privilegio EXECUTE en el procedimiento WLM_COLLECT_STATS

Ejemplos

Ejemplo 1: Llamar WLM_COLLECT_STATS para reunir y restaurar estadísticas.

```
CALL WLM_COLLECT_STATS()
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
Estado de devolución = 0
```

Ejemplo 2: Llamar WLM_COLLECT_STATS para reunir y restaurar estadísticas mientras otra llamada está en proceso.

```
CALL WLM_COLLECT_STATS()
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
SQL1632W La solicitud de reunir y restaurar las estadísticas se ha ignorado porque ya hay otra solicitud para reunir y restaurar estadísticas en proceso.
```

Notas de uso

El procedimiento WLM_COLLECT_STATS realiza la misma tarea de recolección (enviar estadísticas al supervisor de eventos de estadísticas activo) y restauración de las operaciones que se da de forma automática en el intervalo definido por el parámetro de configuración de bases de datos **wlm_collect_int**.

Si llama al procedimiento mientras otra petición de recopilación y restauración está en proceso (por ejemplo, mientras otra invocación del procedimiento se está

ejecutando o mientras se realiza la recopilación automática), se devuelve SQL1632W con SQLSTATE 01H53 y se pasa por alto la nueva petición.

El procedimiento WLM_COLLECT_STATS sólo inicia el proceso de recolección y restauración. Es posible que el procedimiento vuelva al que llama antes de que se graben todas las estadísticas en el supervisor de eventos de estadísticas. En función de la rapidez con que se realice la recolección y restauración de las estadísticas, el procedimiento WLM_COLLECT_STATS (que es en sí una actividad y se incluirá en la estadística de actividades) puede que se cuente en las estadísticas del intervalo de recopilación previo o bien del nuevo intervalo de recopilación que se acaba de iniciar.

Función de tabla WLM_GET_QUEUE_STATS - Devolver estadísticas de cola de umbral

La función WLM_GET_QUEUE_STATS devuelve estadísticas básicas de una o más colas de umbral en todas las particiones activas. Esta función devuelve una fila de estadísticas para cada cola de umbral.

Sintaxis

```
►►—WLM_GET_QUEUE_STATS—(—predicado_umbral—,—dominio_umbral—,——————►  
►—nombre_umbral—,—id_umbral—)——————►►
```

El esquema es SYSPROC.

Parámetros de la función de tabla

predicado_umbral

Argumento de entrada de tipo VARCHAR(27) que especifica un predicado de umbral. Los valores posibles son los siguientes:

CONCDBC

Umbral de actividades de coordinador de base de datos simultáneo

DBCONN

Umbral total de conexiones de partición de base de datos

SCCONN

Umbral total de conexiones de partición de clase de servicio

Si el argumento es nulo o la serie está vacía, se devuelven datos para todos los umbrales que cumplan los demás criterios.

Los valores de *predicado_umbral* coinciden con los de la columna THRESHOLDPREDICATE en la vista SYSCAT.THRESHOLDS.

dominio_umbral

Argumento de entrada de tipo VARCHAR(18) que especifica un dominio de umbral. Los valores posibles son los siguientes:

DB Base de datos

SB Subclase de servicio

SP Superclase de servicio

WA Conjunto de acciones de trabajo

Si el argumento es nulo o la serie está vacía, se devuelven datos para todos los umbrales que cumplan los demás criterios.

Los valores de *dominio_umbral* coinciden con los de la columna DOMAIN en la vista SYSCAT.THRESHOLDS.

nombre_umbral

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de umbral. Si el argumento es nulo o la serie está vacía, se devuelven datos para todos los umbrales que cumplan los demás criterios. Los valores de *nombre_umbral* coinciden con los de la columna THRESHOLDNAME en la vista SYSCAT.THRESHOLDS.

id_umbral

Argumento de entrada de tipo INTEGER que especifica un ID de umbral. Si el argumento es -1, se devuelven datos para todos los umbrales que cumplan los demás criterios. Los valores de *id_umbral* coinciden con los de la columna THRESHOLDID en la vista SYSCAT.THRESHOLDS.

Autorización

Privilegio EXECUTE en la función WLM_GET_QUEUE_STATS.

Ejemplo

La siguiente consulta muestra las estadísticas básicas correspondientes a todas las colas de un sistema, en todas las particiones:

```
SELECT substr(THRESHOLD_NAME, 1, 6) THRESHNAME,
       THRESHOLD_PREDICATE,
       THRESHOLD_DOMAIN,
       DBPARTITIONNUM PART,
       QUEUE_SIZE_TOP,
       QUEUE_TIME_TOTAL,
       QUEUE_ASSIGNMENTS_TOTAL QUEUE_ASSIGN
FROM table(WLM_GET_QUEUE_STATS(' ', ' ', ' ', -1)) as QSTATS
```

Un ejemplo de la salida es el siguiente:

| THRESHNAME | THRESHOLD_PREDICATE | THRESHOLD_DOMAIN | ... | |
|------------|---------------------|------------------|------------------|--------------|
| LIMIT1 | CONCDBC | DB | ... | |
| LIMIT2 | SCCONN | SP | ... | |
| LIMIT3 | DBCONN | DB | ... | |
| ... | PART | QUEUE_SIZE_TOP | QUEUE_TIME_TOTAL | QUEUE_ASSIGN |
| ... | 0 | 12 | 1238540 | 734 |
| ... | 0 | 4 | 741249 | 24 |
| ... | 0 | 7 | 412785 | 128 |

Nota sobre uso

La función no realiza ninguna agregación de datos entre colas (de una partición) ni entre particiones (para una o más colas). Sin embargo, puede utilizar consultas de SQL para agregar datos, como se muestra en el ejemplo anterior.

Información devuelta

Tabla 202. Información devuelta para WLM_GET_QUEUE_STATS

| Nombre de columna | Tipo de datos | Descripción |
|-------------------------|---------------|--|
| THRESHOLD_PREDICATE | VARCHAR(27) | <p>Predicado del umbral responsable de esta cola. Los valores posibles son los siguientes:</p> <p><i>CONCDBC</i> Umbral de actividades de coordinador de base de datos simultáneo</p> <p><i>DBCONN</i> Umbral total de conexiones de partición de base de datos</p> <p><i>SCCONN</i> Umbral total de conexiones de partición de clase de servicio</p> <p>Los valores del predicado de umbral coinciden con los de la columna THRESHOLDPREDICATE en la vista SYSCAT.THRESHOLDS.</p> |
| THRESHOLD_DOMAIN | VARCHAR(18) | <p>Dominio del umbral responsable de esta cola. Los valores posibles son los siguientes:</p> <p><i>DB</i> Base de datos</p> <p><i>SB</i> Subclase de servicio</p> <p><i>SP</i> Superclase de servicio</p> <p><i>WA</i> Conjunto de acciones de trabajo</p> <p>Los valores del dominio de umbral coinciden con los de la columna DOMAIN en la vista SYSCAT.THRESHOLDS.</p> |
| THRESHOLD_NAME | VARCHAR(128) | <p>Nombre exclusivo del umbral responsable de esta cola. El valor de nombre de umbral coincide con los de la columna THRESHOLDNAME en la vista SYSCAT.THRESHOLDS.</p> |
| THRESHOLD_ID | INTEGER | <p>ID exclusivo del umbral responsable de esta cola. El valor de ID de umbral coincide con el de la columna THRESHOLDID en la vista SYSCAT.THRESHOLDS.</p> |
| DBPARTITIONNUM | SMALLINT | <p>Número de partición de la que se ha recopilado este registro.</p> |
| SERVICE_SUPERCLASS_NAME | VARCHAR(128) | <p>Nombre de la superclase de servicio que es el dominio del umbral responsable de esta cola. El valor de la columna es nulo si el dominio del umbral no es ninguna superclase o subclase de servicio.</p> |

Tabla 202. Información devuelta para WLM_GET_QUEUE_STATS (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------------|---------------|---|
| SERVICE_SUBCLASS_NAME | VARCHAR(128) | Nombre de la subclase de servicio que es el dominio del umbral responsable de esta cola. El valor de la columna es nulo si el dominio del umbral no es ninguna subclase de servicio. |
| WORK_ACTION_SET_NAME | VARCHAR(128) | Nombre del conjunto de acciones de trabajo que es el dominio del umbral responsable de esta cola. El valor de la columna es nulo si el dominio del umbral no es ningún conjunto de acciones de trabajo. |
| WORK_CLASS_NAME | VARCHAR(128) | Nombre de la clase de trabajo cuya acción de trabajo pertenece al conjunto de acciones de trabajo que es el dominio del umbral responsable de esta cola. El valor de la columna es nulo si el dominio del umbral no es ningún conjunto de acciones de trabajo. |
| WORKLOAD_NAME | VARCHAR(128) | Nombre de la carga de trabajo que es el dominio del umbral responsable de esta cola. El valor de la columna es nulo si el dominio del umbral no es ninguna carga de trabajo. |
| LAST_RESET | TIMESTAMP | <p>Hora en que se restauraron las estadísticas por última vez. Se pueden producir cuatro sucesos que causarán un restablecimiento de estadísticas:</p> <ul style="list-style-type: none"> • Se invoca el procedimiento WLM_COLLECT_STATS. • El parámetro de configuración wlm_collect_int genera una recopilación y un restablecimiento. • Se reactiva la base de datos. • Se modifica el umbral para el que se notifican estadísticas y se confirma el cambio. <p>La indicación de fecha y hora de LAST_RESET está en el horario local.</p> |
| QUEUE_SIZE_TOP | INTEGER | Mayor número de conexiones o actividades en la cola desde la última restauración. |
| QUEUE_TIME_TOTAL | BIGINT | Suma del tiempo transcurrido en la cola por todas las conexiones o actividades colocadas en dicha cola desde la última restauración. Las unidades son milisegundos. |
| QUEUE_ASSIGNMENTS_TOTAL | BIGINT | Número de conexiones o actividades asignadas a esta cola desde la última restauración. |
| QUEUE_SIZE_CURRENT | INTEGER | Número de conexiones o actividades en la cola. |

Tabla 202. Información devuelta para WLM_GET_QUEUE_STATS (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------------------|---------------|--|
| QUEUE_TIME_LATEST | BIGINT | Tiempo transcurrido en la cola por la última conexión o actividad para dejar la cola. Las unidades son milisegundos. |
| QUEUE_EXIT_TIME_LATEST | TIMESTAMP | Hora en que la última conexión o actividad dejó la cola. |
| THRESHOLD_CURRENT_CONCURRENCY | INTEGER | Número de conexiones o actividades que se están ejecutando actualmente según el umbral. |
| THRESHOLD_MAX_CONCURRENCY | INTEGER | Número máximo de conexiones o actividades que se está ejecutando actualmente con el permiso del umbral. |

Función de tabla WLM_GET_SERVICE_CLASS_AGENTS_V97 - Listar los agentes que se ejecutan en una clase de servicio

La función WLM_GET_SERVICE_CLASS_AGENTS_V97 devuelve la lista de agentes, procesos en modalidad delimitada (procesos db2fmp) y entidades del sistema de la partición especificada que se están ejecutando en una clase de servicio dada o en nombre de una aplicación dada. Las entidades del sistema son procesos y hebras sin agente, como limpiadores de páginas y captadores previos.

Sintaxis

```
►►—WLM_GET_SERVICE_CLASS_AGENTS_V97—(—nombre_superclase_servicio—, —————►
►—nombre_subclase_servicio—,—descriptorcontexto_aplicación—,—númparticiónbd—)————►►
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombre_superclase_servicio

Argumento de entrada de tipo VARCHAR(128) que especifica el nombre de una superclase de servicio de la base de datos actualmente conectada. Si el argumento es nulo o una serie vacía, se recuperan datos correspondientes a todas las superclases de la base de datos.

nombre_subclase_servicio

Argumento de entrada de tipo VARCHAR(128) que hace referencia a una determinada subclase dentro de una superclase. Si el argumento es nulo o una serie vacía, se recuperan datos correspondientes a todas las subclases de la base de datos.

descriptorcontexto_aplicación

Argumento de entrada de tipo BIGINT que especifica el descriptor de contexto de aplicación para el que se debe devolver información del agente. Si el argumento es nulo, se recuperan datos correspondientes a todas las aplicaciones de la base de datos. El descriptor de contexto de aplicación 0 sólo devolverá entidades del sistema.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica el número de partición en la misma instancia que la base de datos conectada actualmente. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos. Si se especifica un valor nulo, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE sobre la función WLM_GET_SERVICE_CLASS_AGENTS_V97

Ejemplo

Ejemplo 1

La consulta siguiente devuelve una lista de agentes asociados al descriptor de contexto de aplicación 1 para todas las particiones de la base de datos. Puede determinar el descriptor de contexto de aplicación mediante el mandato LIST APPLICATIONS o la función de tabla

WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES_V97.

```
SELECT SUBSTR(CHAR(APPLICATION_HANDLE),1,7) AS APPHANDLE,
       SUBSTR(CHAR(DBPARTITIONNUM),1,4) AS PART,
       SUBSTR(CHAR(AGENT_TID),1,9) AS AGENT_TID,
       SUBSTR(AGENT_TYPE,1,11) AS AGENTTYPE,
       SUBSTR(AGENT_STATE,1,10) AS AGENTSTATE,
       SUBSTR(REQUEST_TYPE,1,12) AS REQTYPE,
       SUBSTR(CHAR(UOW_ID),1,6) AS UOW_ID,
       SUBSTR(CHAR(ACTIVITY_ID),1,6) AS ACT_ID
FROM TABLE(WLM_GET_SERVICE_CLASS_AGENTS_V97(CAST(NULL AS VARCHAR(128)),
      CAST(NULL AS VARCHAR(128)), 1, -2)) AS SCDETAILS
ORDER BY APPHANDLE, PART, AGENT_TID
```

Un ejemplo de la salida es el siguiente:

| APPHANDLE | PART | AGENT_TID | AGENTTYPE | AGENTSTATE | REQTYPE | UOW_ID | ACT_ID |
|-----------|------|-----------|-------------|------------|--------------|--------|--------|
| 1 | 0 | 3 | COORDINATOR | ACTIVE | FETCH | 1 | 5 |
| 1 | 0 | 4 | SUBAGENT | ACTIVE | SUBSECTION:1 | 1 | 5 |
| 1 | 1 | 2 | SUBAGENT | ACTIVE | SUBSECTION:2 | 1 | 5 |

La salida muestra un agente de coordinación y un subagente en la partición 0, así como un subagente en la partición 1 que opera en nombre de una actividad con el ID de UOW 1 y el ID de actividad 5. La columna AGENTTYPE con un valor de COORDINATOR tiene un valor FETCH para la columna REQTYPE (que indica el tipo de petición principal o inicial). Esto significa que el tipo de petición es una petición de captación para el agente de coordinación.

Ejemplo 2

La consulta siguiente determina el bloqueo en el que espera un agente:

```
db2 select event_object, event_type, event_state, varchar(event_object_name, 30)
as event_object_name
from table(wlm_get_service_class_agents_v97('',' ',cast(NULL as bigint), -1)) as t
```

Un ejemplo de la salida es el siguiente:

| EVENT_OBJECT | EVENT_TYPE | EVENT_STATE | EVENT_OBJECT_NAME |
|--------------|------------|-------------|-------------------|
| REQUEST | PROCESS | EXECUTING | - |
| REQUEST | PROCESS | EXECUTING | - |
| REQUEST | PROCESS | EXECUTING | - |
| REQUEST | PROCESS | EXECUTING | - |


```

REQUEST      PROCESS      EXECUTING      -
REQUEST      PROCESS      EXECUTING      -
REQUEST      PROCESS      EXECUTING      -
REQUEST      PROCESS      EXECUTING      -
REQUEST      WAIT         IDLE           -
LOCK         ACQUIRE     IDLE           020005000000000000000000054
ROUTINE      PROCESS      EXECUTING      -
REQUEST      PROCESS      EXECUTING      -
REQUEST      PROCESS      EXECUTING      -
REQUEST      PROCESS      EXECUTING      -
REQUEST      PROCESS      EXECUTING      -
REQUEST      PROCESS      EXECUTING      -
REQUEST      PROCESS      EXECUTING      -
REQUEST      PROCESS      EXECUTING      -
REQUEST      PROCESS      EXECUTING      -
REQUEST      PROCESS      EXECUTING      -
REQUEST      PROCESS      EXECUTING      -
REQUEST      PROCESS      EXECUTING      -

```

21 registro(s) seleccionado(s).

Utilizando la misma consulta posteriormente, determina qué umbral de WLM ha puesto en cola un agente:

```

EVENT_OBJECT  EVENT_TYPE  EVENT_STATE  EVENT_OBJECT_NAME
-----
REQUEST       PROCESS     EXECUTING    -
REQUEST       PROCESS     EXECUTING    -
REQUEST       PROCESS     EXECUTING    -
REQUEST       PROCESS     EXECUTING    -
REQUEST       PROCESS     EXECUTING    -
REQUEST       PROCESS     EXECUTING    -
REQUEST       PROCESS     EXECUTING    -
REQUEST       PROCESS     EXECUTING    -
REQUEST       PROCESS     EXECUTING    -
WLM_QUEUE    WAIT        IDLE         MYCONCDBCOORDTH
ROUTINE      PROCESS     EXECUTING    -
REQUEST       PROCESS     EXECUTING    -
REQUEST       PROCESS     EXECUTING    -
REQUEST       PROCESS     EXECUTING    -
REQUEST       PROCESS     EXECUTING    -
REQUEST       PROCESS     EXECUTING    -
REQUEST       PROCESS     EXECUTING    -
REQUEST       PROCESS     EXECUTING    -
REQUEST       PROCESS     EXECUTING    -
REQUEST       PROCESS     EXECUTING    -
REQUEST       PROCESS     EXECUTING    -
REQUEST       PROCESS     EXECUTING    -

```

21 registro(s) seleccionado(s).

Nota sobre uso

Los parámetros tienen el efecto de sumarse (operación AND). Es decir, si se especifican parámetros de entrada conflictivos como una superclase de servicio SUP_A y una subclase de servicio SUB_B de modo que SUB_B no es una subclase de SUP_A, no se devuelve ninguna fila.

Información devuelta

Tabla 203. Información devuelta por WLM_GET_SERVICE_CLASS_AGENTS_V97

| Nombre de columna | Tipo de datos | Descripción |
|-------------------------|------------------|---|
| SERVICE_SUPERCLASS_NAME | VARCHAR (128) | Nombre de la superclase de servicio de la que se ha recopilado este registro. |

Tabla 203. Información devuelta por WLM_GET_SERVICE_CLASS_AGENTS_V97 (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|------------------------|---------------|---|
| SERVICE_SUBCLASS_NAME | VARCHAR (128) | Nombre de la subclase de servicio de la que se ha recopilado este registro. |
| APPLICATION_HANDLE | BIGINT | ID exclusivo a nivel del sistema para la aplicación. En una base de datos de una sola partición, este identificador consiste en un recuento de 16 bits. En una base de datos de varias particiones, este identificador consiste en el número de partición de coordinación concatenado con un recuento de 16 bits. Además, este identificador es el mismo en cada partición en la que la aplicación realiza una conexión secundaria. |
| DBPARTITIONNUM | SMALLINT | Número de partición de la que se ha recopilado este registro. |
| ENTITY | VARCHAR (32) | Uno de los valores siguientes: <ul style="list-style-type: none"> • Si el tipo de entidad es un agente, el valor es db2agent. • Si el tipo de entidad es un proceso en modalidad delimitada, el valor es db2fmp (<i>pid</i>) donde <i>pid</i> es el ID del proceso en modalidad delimitada. • De lo contrario, el valor es el nombre de la entidad del sistema. |
| WORKLOAD_NAME | VARCHAR (128) | Nombre de la carga de trabajo de la que se ha recopilado este registro. |
| WORKLOAD_OCCURRENCE_ID | INTEGER | ID de la aparición de carga de trabajo. Este ID no identifica la aparición de carga de trabajo de forma exclusiva a no ser que esté combinado con el número de partición de base de datos de coordinación y con el nombre de la carga de trabajo. |
| UOW_ID | INTEGER | ID único de la unidad de trabajo en la que se ha iniciado esta actividad. |
| ACTIVITY_ID | INTEGER | ID de actividad exclusivo dentro de una unidad de trabajo. |
| PARENT_UOW_ID | INTEGER | ID único de la unidad de trabajo en la que se ha iniciado la actividad padre de la actividad. El valor de la columna es nulo si esta actividad no tiene actividad padre. |
| PARENT_ACTIVITY_ID | INTEGER | ID exclusivo de actividad dentro de una unidad de trabajo correspondiente a la actividad padre de la actividad cuyo ID es el mismo que <i>activity_id</i> . El valor de esta columna es nulo si esta actividad no tiene actividad padre. |
| AGENT_TID | BIGINT | ID de hebra del agente o de la entidad del sistema. Si este ID no está disponible, el valor de la columna es nulo. |
| AGENT_TYPE | VARCHAR (32) | Tipo de agente. Los tipos de agente son los siguientes: <ul style="list-style-type: none"> • COORDINATOR • OTHER • PDBSUBAGENT • SMPSUBAGENT Si el valor es COORDINATOR, el ID de agente puede variar en entorno de concentrador. |
| SMP_COORDINATOR | INTEGER | Indica si el agente es o no de coordinación SMP; 1 indica que lo es, 0 que no lo es. |

Tabla 203. Información devuelta por WLM_GET_SERVICE_CLASS_AGENTS_V97 (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| AGENT_SUBTYPE | VARCHAR (32) | Subtipo de agente. Los subtipos posibles son los siguientes: <ul style="list-style-type: none"> • DSS • OTHER • RPC • SMP |
| AGENT_STATE | VARCHAR (32) | Indica si el agente está asociado o activo. Los valores posibles son: <ul style="list-style-type: none"> • ASSOCIATED • ACTIVE |
| EVENT_TYPE | VARCHAR (32) | El tipo del último suceso que ha procesado este agente. Los valores posibles son los siguientes: <ul style="list-style-type: none"> • ACQUIRE • PROCESS • WAIT |
| EVENT_OBJECT | VARCHAR (32) | El objeto del último suceso que ha procesado este agente. Los valores posibles son los siguientes: <ul style="list-style-type: none"> • COMPRESSION_DICTIONARY_BUILD • IMPLICIT_REBIND • INDEX_RECREATE • LOCK • LOCK_ESCALATION • QP_QUEUE • REMOTE_REQUEST • REQUEST • ROUTINE • WLM_QUEUE |
| EVENT_STATE | VARCHAR (32) | El estado del último suceso que ha procesado este agente. Los valores posibles son los siguientes: <ul style="list-style-type: none"> • EXECUTING • IDLE |
| REQUEST_ID | VARCHAR(64) | ID de petición. Este valor es único solamente si se combina con el valor de <i>descriptorcontexto_aplicación</i> . Puede utilizar esta combinación para distinguir entre una petición que tarda mucho tiempo y varias peticiones; por ejemplo, para distinguir varias captaciones de una captación larga. |

Tabla 203. Información devuelta por WLM_GET_SERVICE_CLASS_AGENTS_V97 (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| REQUEST_TYPE | VARCHAR (32) | <p>Tipo de petición. Los valores posibles son los siguientes:</p> <ul style="list-style-type: none"> • Para agentes de coordinación: <ul style="list-style-type: none"> – CLOSE – COMMIT – COMPILE – DESCRIBE – EXCSQLSET – EXECIMMD – EXECUTE – FETCH – INTERNAL <i>número</i>, donde <i>número</i> es el valor de la constante interna – OPEN – PREPARE – REBIND – REDISTRIBUTE – REORG – ROLLBACK – RUNSTATS • Para subagentes con un AGENT_SUBTYPE de DSS o SMP: <ul style="list-style-type: none"> – Si el número de subsección es distinto de cero, el número de subsección está en el formato <i>SUBSECCIÓN:número subsección</i>; de lo contrario, devuelve NULL. |

Tabla 203. Información devuelta por WLM_GET_SERVICE_CLASS_AGENTS_V97 (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-----------------------------|---------------|--|
| REQUEST_TYPE (continuación) | VARCHAR (32) | <ul style="list-style-type: none"> • Para subagentes con un AGENT_SUBTYPE de RPC: <ul style="list-style-type: none"> - ABP - CATALOG - INTERNAL - REORG - RUNSTATS - WLM • Para subagentes con un SUBTYPE de OTHER: <ul style="list-style-type: none"> - ABP - APP_RBSVPT - APP_RELSVPT - BACKUP - CLOSE - EXTERNAL_RBSVPT - EVMON - FORCE - FORCE_ALL - INTERNAL <i>número</i>, donde <i>número</i> es el valor de la constante interna - INTERRUPT - NOOP (si no hay petición) - QP - REDISTRIBUTE - STMT_RBSVPT - STOP_USING - UPDATE_DBM_CFG - WLM |
| NESTING_LEVEL | INTEGER | Nivel de anidamiento de la actividad cuyo ID es activity_id. El nivel de anidamiento es la profundidad a la que se anida esta actividad dentro de la actividad principal. |
| INVOCATION_ID | INTEGER | ID de invocación, que distingue una determinada invocación de una actividad de otras del mismo nivel de anidamiento. |
| ROUTINE_ID | INTEGER | ID único para una rutina. El valor de esta columna es nulo si la actividad no forma parte de una rutina. |
| EVENT_OBJECT_NAME | VARCHAR(1024) | Nombre de objeto de suceso. Si el valor de EVENT_OBJECT es LOCK, el valor de esta columna es el nombre del bloqueo en el que espera el agente. Si el valor de EVENT_OBJECT es WLM_QUEUE, el valor de la columna es el nombre del umbral de WLM que el agente ha puesto en cola. De lo contrario, el valor es NULL. |
| APPLICATION_NAME | VARCHAR (128) | Reservado para una utilización futura. |
| APPLICATION_ID | VARCHAR (128) | Reservado para una utilización futura. |
| CLIENT_PID | BIGINT | Reservado para una utilización futura. |
| SESSION_AUTH_ID | VARCHAR (128) | Reservado para una utilización futura. |

Tabla 203. Información devuelta por WLM_GET_SERVICE_CLASS_AGENTS_V97 (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|------------------------------|-----------------------------|---|
| REQUEST_START_TIME | TIMESTAMP | Reservado para una utilización futura. |
| AGENT_STATE_LAST_UPDATE_TIME | TIMESTAMP | Reservado para una utilización futura. |
| EXECUTABLE_ID | VARCHAR(32) FOR BIT DATA | Símbolo binario generado en el servidor de datos que identifica de forma exclusiva la sección en la que está trabajando el agente. Se puede usar el ID ejecutable como entrada para distintas interfaces de supervisión para obtener datos sobre la sección. Se devolverá un valor NULL si el agente no está trabajando en una sección. |

WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES_V97 - Lista de apariciones de carga de trabajo

La función WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES_V97 devuelve la lista de todas las apariciones de carga de trabajo que se ejecutan en una determinada clase de servicio de una determinada partición. Una aparición de carga de trabajo es una conexión de base de datos específica cuyos atributos coinciden con la definición de una carga de trabajo y por lo tanto está asociada a la carga de trabajo o asignada a la misma.

Sintaxis

```
►►—WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES_V97—(—nombre_superclase_servicio—,—————►
►—nombre_subclase_servicio—,—númparticiónbd—)—————►►
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombre_superclase_servicio

Argumento de entrada de tipo VARCHAR(128) que especifica el nombre de una superclase de servicio de la base de datos actualmente conectada. Si el argumento es nulo o una serie vacía, se recuperan datos correspondientes a todas las superclases de la base de datos que coinciden con los valores de los otros parámetros.

nombre_subclase_servicio

Argumento de entrada de tipo VARCHAR(128) que especifica el nombre de una subclase de servicio de la base de datos actualmente conectada. Si el argumento es nulo o una serie vacía, se recuperan datos correspondientes a todas las subclases de la base de datos que coinciden con los valores de los otros parámetros.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica el número de partición en la misma instancia que la base de datos conectada actualmente. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos. Si se especifica el valor nulo, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE sobre la función
WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES_V97

Ejemplo

Si un administrador desea ver qué apariciones de carga de trabajo se están ejecutando en el sistema como un conjunto, puede llamar a la función WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES_V97 especificando un valor nulo o una serie vacía para *nombre_superclase_servicio* y *nombre_subclase_servicio* y -2 para *dbpartitionnum*:

```
SELECT SUBSTR(SERVICE_SUPERCLASS_NAME,1,19) AS SUPERCLASS_NAME,
       SUBSTR(SERVICE_SUBCLASS_NAME,1,18) AS SUBCLASS_NAME,
       SUBSTR(CHAR(DBPARTITIONNUM),1,4) AS PART,
       SUBSTR(CHAR(COORD_PARTITION_NUM),1,4) AS COORDPART,
       SUBSTR(CHAR(APPLICATION_HANDLE),1,7) AS APPHNDL,
       SUBSTR(WORKLOAD_NAME,1,22) AS WORKLOAD_NAME,
       SUBSTR(CHAR(WORKLOAD_OCCURRENCE_ID),1,6) AS WLO_ID
FROM TABLE(WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES_V97
(CAST(NULL AS VARCHAR(128)), CAST(NULL AS VARCHAR(128)), -2))
AS SCINFO
ORDER BY SUPERCLASS_NAME, SUBCLASS_NAME, PART, APPHNDL,
       WORKLOAD_NAME, WLO_ID
```

Si el sistema tiene cuatro particiones de base de datos y está ejecutando dos cargas de trabajo en este momento, la consulta anterior genera resultados como los siguientes:

| SUPERCLASS_NAME | SUBCLASS_NAME | PART | COORDPART | ... |
|---------------------|--------------------|------------------------|-----------|------|
| ---- | ----- | ---- | ----- | ---- |
| SYSDEFAULTMAINTENAN | SYSDEFAULTSUBCLASS | 0 | 0 | ... |
| SYSDEFAULTSYSTEMCLA | SYSDEFAULTSUBCLASS | 0 | 0 | ... |
| SYSDEFAULTUSERCLASS | SYSDEFAULTSUBCLASS | 0 | 0 | ... |
| SYSDEFAULTUSERCLASS | SYSDEFAULTSUBCLASS | 0 | 0 | ... |
| SYSDEFAULTUSERCLASS | SYSDEFAULTSUBCLASS | 1 | 0 | ... |
| SYSDEFAULTUSERCLASS | SYSDEFAULTSUBCLASS | 1 | 0 | ... |
| SYSDEFAULTUSERCLASS | SYSDEFAULTSUBCLASS | 2 | 0 | ... |
| SYSDEFAULTUSERCLASS | SYSDEFAULTSUBCLASS | 2 | 0 | ... |
| SYSDEFAULTUSERCLASS | SYSDEFAULTSUBCLASS | 3 | 0 | ... |
| SYSDEFAULTUSERCLASS | SYSDEFAULTSUBCLASS | 3 | 0 | ... |
| ... | APPHNDL | WORKLOAD_NAME | WLO_ID | |
| ... | ---- | ----- | ----- | |
| ... | - | - | - | |
| ... | - | - | - | |
| ... | 1 | SYSDEFAULTUSERWORKLOAD | 1 | |
| ... | 2 | SYSDEFAULTUSERWORKLOAD | 2 | |
| ... | 1 | SYSDEFAULTUSERWORKLOAD | 1 | |
| ... | 2 | SYSDEFAULTUSERWORKLOAD | 2 | |
| ... | 1 | SYSDEFAULTUSERWORKLOAD | 1 | |
| ... | 2 | SYSDEFAULTUSERWORKLOAD | 2 | |
| ... | 1 | SYSDEFAULTUSERWORKLOAD | 1 | |
| ... | 2 | SYSDEFAULTUSERWORKLOAD | 2 | |

Nota sobre uso

Los parámetros tienen el efecto de sumarse (operación AND). Es decir, si se especifican parámetros de entrada conflictivos como una superclase de servicio SUP_A y una subclase de servicio SUB_B de modo que SUB_B no es una subclase de SUP_A, no se devuelve ninguna fila.

Nota: Las estadísticas notificadas para la aparición de carga de trabajo (por ejemplo, coord_act_completed_total) se restablecen al principio de cada unidad de trabajo cuando se combinan con las estadísticas correspondientes de carga de trabajo.

Información devuelta

Tabla 204. Información devuelta para WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES_V97

| Nombre de columna | Tipo de datos | Descripción |
|-------------------------|---------------|---|
| SERVICE_SUPERCLASS_NAME | VARCHAR(128) | Nombre de la superclase de servicio de la que se ha recopilado este registro. |
| SERVICE_SUBCLASS_NAME | VARCHAR(128) | Nombre de la subclase de servicio de la que se ha recopilado este registro. |
| DBPARTITIONNUM | SMALLINT | Número de partición de la que se ha recopilado este registro. |
| COORD_PARTITION_NUM | SMALLINT | Número de partición de la partición de coordinación de la aparición de carga de trabajo dada. |
| APPLICATION_HANDLE | BIGINT | ID exclusivo a nivel del sistema para la aplicación. En una base de datos de una sola partición, este identificador consiste en un recuento de 16 bits. En una base de datos de varias particiones, este identificador consiste en el número de partición de coordinación concatenado con un recuento de 16 bits. Además, este identificador es el mismo en cada partición en la que la aplicación realiza una conexión secundaria. |
| WORKLOAD_NAME | VARCHAR(128) | Nombre de la carga de trabajo de la que se ha recopilado este registro. |
| WORKLOAD_OCCURRENCE_ID | INTEGER | ID de la aparición de carga de trabajo. Este ID no identifica la aparición de carga de trabajo de forma exclusiva a no ser que esté combinado con el número de partición de base de datos de coordinación y con el nombre de la carga de trabajo. |

Tabla 204. Información devuelta para
WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES_V97 (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|---------------------------|---------------|--|
| WORKLOAD_OCCURRENCE_STATE | VARCHAR(32) | <p>Estado de aparición de carga de trabajo. Los valores son los siguientes:</p> <p><i>DECOUPLED</i> La aparición de la carga de trabajo no tiene ningún agente de coordinación asignado (caso de concentrador).</p> <p><i>DISCONNECTPEND</i> La aparición de la carga de trabajo se desconecta de la base de datos.</p> <p><i>FORCED</i> Se ha forzado la desconexión de la base de datos de la aparición de la carga de trabajo.</p> <p><i>INTERRUPTED</i> Se ha interrumpido la aparición de la carga de trabajo.</p> <p><i>QUEUED</i> Query Patroller o un umbral de colas de gestión de carga de trabajo ha puesto en cola el agente coordinador de apariciones de carga de trabajo. En un entorno de característica de particionamiento de base de datos (DPF), es posible que este estado indique que el agente coordinador ha realizado un RPC en la partición de catálogo para obtener tíquets de umbral y aún no ha recibido una respuesta.</p> <p><i>TRANSIENT</i> La aparición de la carga de trabajo aún no se ha correlacionado con una superclase de servicio.</p> <p><i>UOWEXEC</i> La aparición de la carga de trabajo está procesando una petición.</p> <p><i>UOWWAIT</i> La aparición de la carga de trabajo está a la espera de una solicitud procedente del cliente.</p> |
| UOW_ID | INTEGER | ID único de la unidad de trabajo en la que se ha iniciado esta aparición de carga de trabajo. |
| SYSTEM_AUTH_ID | VARCHAR(128) | ID de autorización del sistema bajo el que la aparición de la carga de trabajo se ha insertado en el sistema. |
| SESSION_AUTH_ID | VARCHAR(128) | ID de autorización de la sesión bajo el que la aparición de la carga de trabajo se ha insertado en el sistema. |
| APPLICATION_NAME | VARCHAR(128) | Nombre de la aplicación que ha creado esta aparición de la carga de trabajo. |

Tabla 204. Información devuelta para
WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES_V97 (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|---------------------------|---------------|---|
| CLIENT_WRKSTNNAME | VARCHAR(255) | Valor actual del registro especial CLIENT_WRKSTNNAME correspondiente a esta aparición de la carga de trabajo. |
| CLIENT_ACCTNG | VARCHAR(255) | Valor actual del registro especial CLIENT_ACCTNG correspondiente a esta aparición de la carga de trabajo. |
| CLIENT_USER | VARCHAR(255) | Valor actual del registro especial CLIENT_USERID correspondiente a esta aparición de la carga de trabajo. |
| CLIENT_APPLNAME | VARCHAR(255) | Valor actual del registro especial CLIENT_APPLNAME correspondiente a esta aparición de la carga de trabajo. |
| COORD_ACT_COMPLETED_TOTAL | INTEGER | Número de actividades de coordinación a cualquier nivel de anidamiento completadas hasta el momento en la unidad de trabajo actual de esta aparición de la carga de trabajo. Esta estadística se actualiza cada vez que una actividad de esta aparición de la carga de trabajo se completa y se restablece al principio de cada unidad de trabajo. |
| COORD_ACT_ABORTED_TOTAL | INTEGER | Número de actividades de coordinación terminadas anormalmente hasta el momento en la unidad de trabajo actual de esta aparición de la carga de trabajo. Esta estadística se actualiza cada vez que una actividad de esta aparición de la carga de trabajo termina anormalmente y se restablece al principio de cada unidad de trabajo. |
| COORD_ACT_REJECTED_TOTAL | INTEGER | Número de actividades de coordinación rechazadas hasta el momento en la unidad de trabajo actual de esta aparición de la carga de trabajo. Una actividad se cuenta como rechazada cuando se impide su ejecución mediante una acción que impide el trabajo de ejecución o mediante un umbral de predicción. Esta estadística se actualiza cada vez que una actividad de esta aparición de la carga de trabajo se rechaza y se restablece al principio de cada unidad de trabajo. |
| CONCURRENT_ACT_TOP | INTEGER | Número más alto de actividades simultáneas en cualquier nivel de anidamiento que están en estado de ejecución (lo que incluye las desocupadas y las en espera) y en cola que se ha alcanzado para esta aparición de la carga de trabajo en la unidad de trabajo actual. Esta estadística se restablece al principio de cada unidad de trabajo. |
| ADDRESS | VARCHAR(255) | La dirección IP o el nombre de dominio seguro que ha creado esta aparición de la carga de trabajo. Los nombres de dominio seguros se indican convertidos en direcciones IP. |

Función de tabla WLM_GET_SERVICE_SUBCLASS_STATS_V97 - Devolver estadísticas de subclases de servicio

La función WLM_GET_SERVICE_SUBCLASS_STATS_V97 devuelve estadísticas básicas para una o varias subclases de servicio.

Sintaxis

```
►—WLM_GET_SERVICE_SUBCLASS_STATS_V97—(—nombre_superclase_servicio—,——►  
►—nombre_subclase_servicio—, —númparticiónbd—)——►
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombre_superclase_servicio

Argumento de entrada de tipo VARCHAR(128) que especifica el nombre de una superclase de servicio de la base de datos actualmente conectada. Si el argumento es nulo o una serie vacía, se recuperan datos correspondientes a todas las superclases de la base de datos.

nombre_subclase_servicio

Argumento de entrada de tipo VARCHAR(128) que especifica el nombre de una subclase de servicio de la base de datos actualmente conectada. Si el argumento es nulo o una serie vacía, se recuperan datos correspondientes a todas las subclases de la base de datos.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número de partición válido en la misma instancia que la base de datos conectada actualmente. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos. Si se especifica el valor nulo, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE sobre la función WLM_GET_SERVICE_SUBCLASS_STATS_V97.

Ejemplos

Ejemplo 1: Puesto que cada actividad se tiene que correlacionar con una clase de servicio de DB2 antes de que se ejecute, se puede supervisar el estado global del sistema utilizando las funciones de tabla de estadísticas de clase de servicio y consultando todas las clases de servicio de todas las particiones. En el ejemplo siguiente, se pasa un valor nulo para *nombre_superclase_servicio* y *nombre_subclase_servicio* para devolver estadísticas para todas las clases de servicio, y se especifica el valor -2 para *númparticiónbd* para devolver estadísticas para todas las particiones:

```
SELECT SUBSTR(SERVICE_SUPERCLASS_NAME,1,19) AS SUPERCLASS_NAME,  
       SUBSTR(SERVICE_SUBCLASS_NAME,1,18) AS SUBCLASS_NAME,  
       SUBSTR(CHAR(DBPARTITIONNUM),1,4) AS PART,  
       CAST(COORD_ACT_LIFETIME_AVG / 1000 AS DECIMAL(9,3))  
       AS AVGLIFETIME,  
       CAST(COORD_ACT_LIFETIME_STDDEV / 1000 AS DECIMAL(9,3))  
       AS STDDEVLIFETIME,
```

```

SUBSTR(CAST(LAST_RESET AS VARCHAR(30)),1,16) AS LAST_RESET
FROM TABLE(WLM_GET_SERVICE_SUBCLASS_STATS_V97(CAST(NULL AS VARCHAR(128)),
CAST(NULL AS VARCHAR(128)), -2)) AS SCSTATS
ORDER BY SUPERCLASS_NAME, SUBCLASS_NAME, PART

```

La sentencia devuelve estadísticas de clase de servicio como duración media de la actividad y desviación estándar en segundos, tal como se muestra en el siguiente ejemplo de salida:

```

SUPERCLASS_NAME  SUBCLASS_NAME  PART ...
-----
SYSDEFAULTUSERCLASS SYSDEFAULTSUBCLASS 0 ...
SYSDEFAULTUSERCLASS SYSDEFAULTSUBCLASS 1 ...
SYSDEFAULTUSERCLASS SYSDEFAULTSUBCLASS 2 ...
SYSDEFAULTUSERCLASS SYSDEFAULTSUBCLASS 3 ...
... AVGLIFETIME STDDEVLIFETIME LAST_RESET
... -----
... 691.242 34.322 2006-07-24-11.44
... 644.740 22.124 2006-07-24-11.44
... 612.431 43.347 2006-07-24-11.44
... 593.451 28.329 2006-07-24-11.44

```

Ejemplo 2: La misma función de tabla también puede dar el valor más alto correspondiente a simultaneidad promedia de actividades de coordinación que se ejecutan en la clase de servicio de cada partición:

```

SELECT SUBSTR(SERVICE_SUPERCLASS_NAME,1,19) AS SUPERCLASS_NAME,
SUBSTR(SERVICE_SUBCLASS_NAME,1,18) AS SUBCLASS_NAME,
SUBSTR(CHAR(DBPARTITIONNUM),1,4) AS PART,
CONCURRENT_ACT_TOP AS ACTTOP,
CONCURRENT_WLO_TOP AS CONNTOP
FROM TABLE(WLM_GET_SERVICE_SUBCLASS_STATS_V97(CAST(NULL AS VARCHAR(128)),
CAST(NULL AS VARCHAR(128)), -2)) AS SCSTATS
ORDER BY SUPERCLASS_NAME, SUBCLASS_NAME, PART

```

Un ejemplo de la salida es el siguiente:

```

SUPERCLASS_NAME  SUBCLASS_NAME  PART ACTTOP  CONNTOP
-----
SYSDEFAULTUSERCLASS SYSDEFAULTSUBCLASS 0 10 7
SYSDEFAULTUSERCLASS SYSDEFAULTSUBCLASS 1 0 0
SYSDEFAULTUSERCLASS SYSDEFAULTSUBCLASS 2 0 0
SYSDEFAULTUSERCLASS SYSDEFAULTSUBCLASS 3 0 0

```

Mediante la comprobación de los tiempos promedios de ejecución y los números de actividades en la salida de esta función de tabla, obtendrá una buena vista a nivel alto de la carga de cada partición para una determinada base de datos. Cualquier variación significativa de los indicadores de nivel alto que devuelve esta función de tabla puede indicar un cambio en la carga del sistema.

Ejemplo 3: Si una actividad utiliza umbrales con acciones REMAP ACTIVITY TO, la actividad puede dedicar tiempo a más de una clase de servicio durante su vida útil. Se puede determinar cuántas actividades han pasado a través de un conjunto de clases de servicio consultando las columnas ACTIVITIES_MAPPED_IN y ACTIVITIES_MAPPED_OUT, como se muestra en el ejemplo siguiente:

```

SELECT SUBSTR(SERVICE_SUPERCLASS_NAME,1,19) AS SUPERCLASS_NAME,
SUBSTR(SERVICE_SUBCLASS_NAME,1,18) AS SUBCLASS_NAME,
ACTIVITIES_MAPPED_IN AS MAPPED_IN,
ACTIVITIES_MAPPED_OUT AS MAPPED_OUT
FROM TABLE(WLM_GET_SERVICE_SUBCLASS_STATS_V97(CAST(NULL AS VARCHAR(128)),
CAST(NULL AS VARCHAR(128)), -2)) AS SCSTATS
ORDER BY SUPERCLASS_NAME, SUBCLASS_NAME

```

Un ejemplo de la salida es el siguiente:

| SUPERCLASS_NAME | SUBCLASS_NAME | MAPPED_IN | MAPPED_OUT |
|---------------------|--------------------|-----------|------------|
| SYSDEFAULTUSERCLASS | SYSDEFAULTSUBCLASS | 0 | 0 |
| SUPERCLASS1 | SYSDEFAULTSUBCLASS | 0 | 0 |
| SUPERCLASS1 | SUBCLASS1 | 0 | 7 |
| SUPERCLASS1 | SUBCLASS2 | 7 | 0 |

Notas de uso

Algunas estadísticas sólo se devolverán si se establecen los parámetros COLLECT AGGREGATE ACTIVITY DATA y COLLECT AGGREGATE REQUEST DATA para la subclase de servicio correspondiente en un valor que no sea "NONE".

La función de tabla WLM_GET_SERVICE_SUBCLASS_STATS_V97 devuelve una fila de datos por subclase de servicio y por partición. La función no realiza ninguna agregación de datos entre clases de servicio (de una partición) ni entre particiones (para una o más clases de servicio). Sin embargo, puede utilizar consultas de SQL para agregar datos.

Los parámetros tienen el efecto de sumarse (operación AND). Es decir, si se especifican parámetros de entrada conflictivos, como un nombre de superclase llamado SUPA y una subclase llamada SUBB de modo que SUBB no es una subclase de SUPA, no se devolverá ninguna fila.

Información devuelta

Tabla 205. Información devuelta para WLM_GET_SERVICE_SUBCLASS_STATS_V97

| Nombre de columna | Tipo de datos | Descripción |
|-------------------------|---------------|---|
| SERVICE_SUPERCLASS_NAME | VARCHAR(128) | Nombre de la superclase de servicio de la que se ha recopilado este registro. |
| SERVICE_SUBCLASS_NAME | VARCHAR(128) | Nombre de la subclase de servicio de la que se ha recopilado este registro. |
| DBPARTITIONNUM | SMALLINT | Número de partición de la que se ha recopilado este registro. |
| LAST_RESET | TIMESTAMP | <p>Hora en que se restauraron las estadísticas por última vez. Se pueden producir cuatro sucesos que causarán un restablecimiento de estadísticas:</p> <ul style="list-style-type: none"> • Se invoca el procedimiento WLM_COLLECT_STATS. • El parámetro de configuración wlm_collect_int genera una recopilación y un restablecimiento. • Se reactiva la base de datos. • Se modifica la subclase de servicio para la que se notifican estadísticas y se confirma el cambio. <p>La indicación de fecha y hora de LAST_RESET está en el horario local.</p> |

Tabla 205. Información devuelta para WLM_GET_SERVICE_SUBCLASS_STATS_V97 (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|---------------------------|---------------|--|
| COORD_ACT_COMPLETED_TOTAL | BIGINT | <p>Número total de actividades de coordinación que se han enviado desde el último restablecimiento y que se han completado satisfactoriamente. Esta cuenta se actualiza a medida que se completa cada actividad.</p> <p>Si correlaciona una actividad con una subclase de servicio diferente, esta actividad contará únicamente para establecer el total de la subclase en la que se completa.</p> |
| COORD_ACT_ABORTED_TOTAL | BIGINT | <p>Número total de actividades de coordinación que se han enviado desde el último restablecimiento y que se han completado con errores. Esta cuenta se actualiza a medida que termina anormalmente cada actividad.</p> <p>Si correlaciona una actividad con una subclase de servicio diferente, esta actividad contará únicamente para establecer el total de la subclase en la que se aborta.</p> |
| COORD_ACT_REJECTED_TOTAL | BIGINT | <p>Número total de actividades de coordinación que se han enviado desde el último restablecimiento y que se han rechazado antes de la ejecución. Una actividad se cuenta como rechazada cuando se impide su ejecución mediante una acción que impide el trabajo de ejecución o mediante un umbral de predicción. Esta cuenta se actualiza a medida que se rechaza cada actividad.</p> |
| CONCURRENT_ACT_TOP | INTEGER | <p>Número más alto de actividades simultáneas en cualquier nivel de anidamiento que se están ejecutando (lo que incluye las desocupadas y las en espera) que se ha alcanzado para esta subclase de servicio.</p> |

Tabla 205. Información devuelta para WLM_GET_SERVICE_SUBCLASS_STATS_V97 (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|------------------------|---------------|---|
| COORD_ACT_LIFETIME_TOP | BIGINT | <p>Indicador de nivel alto correspondiente a la duración de actividad de coordinación, evaluado a todos los niveles de anidamiento. Si el parámetro COLLECT AGGREGATE ACTIVITY DATA de la clase de servicio está establecido en NONE, el valor de la columna es nulo. Las unidades son milisegundos.</p> <p>Para utilizar esta estadística eficazmente cuando la clase de servicio incluye subclases correlacionadas, es necesario agregar la marca de límite superior COORD_ACT_LIFETIME_TOP de la subclase de servicio con la del resto de subclases afectadas por el mismo umbral o umbrales de correlación. Debe agregar estos valores ya que una actividad puede completarse después de que una subclase se haya correlacionado con una subclase de servicio distinta. El tiempo que dedica la actividad en otras subclases de servicio antes de correlacionarse sólo cuenta para establecer el total de la clase de servicio en la que se completa.</p> |
| COORD_ACT_LIFETIME_AVG | DOUBLE | <p>Media aritmética de duración para actividades de coordinación a nivel de anidamiento 0 asociadas a esta subclase de servicio desde el último restablecimiento. Si se ha desbordado el promedio del que se hace un seguimiento interno, se devuelve el valor -2. Si el parámetro COLLECT AGGREGATE ACTIVITY DATA de la clase de servicio está establecido en NONE, el valor de la columna es nulo. Las unidades son milisegundos.</p> <p>El valor de COORD_ACT_LIFETIME_AVG de una subclase de servicio no se ve afectado por las actividades que pasan a través de la subclase pero que están correlacionadas con una subclase distinta antes de completarse.</p> |

Tabla 205. Información devuelta para WLM_GET_SERVICE_SUBCLASS_STATS_V97 (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|---------------------------|---------------|---|
| COORD_ACT_LIFETIME_STDDEV | DOUBLE | <p>Desviación estándar de duración para actividades de coordinación a nivel de anidamiento 0 asociadas a esta subclase de servicio desde el último restablecimiento. Si el parámetro COLLECT AGGREGATE ACTIVITY DATA de la clase de servicio está establecido en NONE, el valor de la columna es nulo. Las unidades son milisegundos.</p> <p>Esta desviación estándar se calcula a partir del histograma de duración de las actividades de coordinación y puede resultar imprecisa si el histograma no se ha dimensionado correctamente para dar cabida a los datos. Se devuelve el valor -1 si alguno de los valores cae dentro de la bandeja del último histograma.</p> <p>El valor de COORD_ACT_LIFETIME_STDDEV de una subclase de servicio no se ve afectado por las actividades que pasan a través de la subclase de servicio pero que están correlacionadas con una subclase distinta antes de completarse.</p> |
| COORD_ACT_EXEC_TIME_AVG | DOUBLE | <p>Media aritmética de tiempos de ejecución para actividades de coordinación a nivel de anidamiento 0 asociadas a esta subclase de servicio desde el último restablecimiento. Si se ha desbordado el promedio del que se hace un seguimiento interno, se devuelve el valor -2. Si el parámetro COLLECT AGGREGATE ACTIVITY DATA de la clase de servicio está establecido en NONE, el valor de la columna es nulo. Las unidades son milisegundos.</p> <p>El promedio del tiempo de ejecución de una subclase de servicio no se ve afectado por las actividades que pasan a través de la subclase pero que están correlacionadas con una subclase distinta antes de completarse.</p> |

Tabla 205. Información devuelta para WLM_GET_SERVICE_SUBCLASS_STATS_V97 (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|----------------------------|---------------|---|
| COORD_ACT_EXEC_TIME_STDDEV | DOUBLE | <p>Desviación estándar de tiempos de ejecución para actividades de coordinación a nivel de anidamiento 0 asociadas a esta subclase de servicio desde el último restablecimiento. Las unidades son milisegundos.</p> <p>Esta desviación estándar se calcula a partir del histograma de tiempo de ejecución de las actividades de coordinación y puede resultar imprecisa si el histograma no se ha dimensionado correctamente para dar cabida a los datos. Se devuelve el valor -1 si alguno de los valores cae dentro de la bandeja del último histograma.</p> <p>La desviación estándar del tiempo de ejecución de una subclase de servicio no se ve afectada por las actividades que pasan a través de la subclase pero que están correlacionadas con una subclase distinta antes de completarse.</p> |
| COORD_ACT_QUEUE_TIME_AVG | DOUBLE | <p>Media aritmética de tiempo en cola para actividades de coordinación a nivel de anidamiento 0 asociadas a esta subclase de servicio desde el último restablecimiento. Si se ha desbordado el promedio del que se hace un seguimiento interno, se devuelve el valor -2. Si el parámetro COLLECT AGGREGATE ACTIVITY DATA de la clase de servicio está establecido en NONE, el valor de la columna es nulo. Las unidades son milisegundos.</p> <p>El promedio del tiempo de cola sólo se cuenta para establecer el total de la subclase de servicio en la que se ha puesto en cola la actividad.</p> |

Tabla 205. Información devuelta para WLM_GET_SERVICE_SUBCLASS_STATS_V97 (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-----------------------------|---------------|---|
| COORD_ACT_QUEUE_TIME_STDDEV | DOUBLE | <p>Desviación estándar del tiempo de cola para actividades de coordinación a nivel de anidamiento 0 asociadas a esta subclase de servicio desde el último restablecimiento. Si el parámetro COLLECT AGGREGATE ACTIVITY DATA de la clase de servicio está establecido en NONE, el valor de la columna es nulo. Las unidades son milisegundos.</p> <p>Esta desviación estándar se calcula a partir del histograma de tiempo de cola de las actividades de coordinación y puede resultar imprecisa si el histograma no se ha dimensionado correctamente para dar cabida a los datos. Se devuelve el valor -1 si alguno de los valores cae dentro de la bandeja del último histograma.</p> <p>La desviación estándar del tiempo de cola sólo se cuenta para establecer el total de la subclase de servicio en la que se ha puesto en cola la actividad.</p> |
| NUM_REQUESTS_ACTIVE | BIGINT | <p>Número de peticiones que se están ejecutando en la subclase de servicio en el momento de ejecutar esta función de tabla.</p> |
| NUM_REQUESTS_TOTAL | BIGINT | <p>Número de peticiones que se han acabado de ejecutar en esta subclase de servicio desde el último restablecimiento. Este estado acabado se aplica a cualquier petición, independientemente de su pertenencia a una actividad. Si el parámetro COLLECT AGGREGATE ACTIVITY DATA de la clase de servicio está establecido en NONE, el valor de la columna es nulo.</p> <p>El valor NUM_REQUESTS_TOTAL de una subclase de servicio no se ve afectado por peticiones que pasan a través de la subclase de servicio pero que no se completan en la misma.</p> |

Tabla 205. Información devuelta para WLM_GET_SERVICE_SUBCLASS_STATS_V97 (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|--------------------------|---------------|---|
| REQUEST_EXEC_TIME_AVG | DOUBLE | <p>Media aritmética de tiempos de ejecución para peticiones asociadas a esta subclase de servicio desde el último restablecimiento. Las unidades son milisegundos. Si se ha desbordado el promedio del que se hace un seguimiento interno, se devuelve el valor -2. Si el parámetro COLLECT AGGREGATE REQUEST DATA en esta clase de servicio está establecido en NONE, el valor de esta columna es NULL.</p> <p>El promedio del tiempo de ejecución de una subclase de servicio no se ve afectada por peticiones que pasan a través de la subclase pero que no se completan en la misma.</p> |
| REQUEST_EXEC_TIME_STDDEV | DOUBLE | <p>Desviación estándar de tiempos de ejecución para peticiones asociadas a esta subclase de servicio desde el último restablecimiento. Las unidades son milisegundos. Si el parámetro COLLECT AGGREGATE REQUEST DATA en la clase de servicio está establecido en NONE, el valor de esta columna es NULL.</p> <p>Esta desviación estándar se calcula a partir del histograma de tiempo de ejecución de peticiones y puede resultar imprecisa si el histograma no se ha dimensionado correctamente para dar cabida a los datos. Se devuelve el valor -1 si alguno de los valores cae dentro de la bandeja del último histograma.</p> <p>La desviación estándar del tiempo de ejecución de una subclase de servicio no se ve afectada por peticiones que pasan a través de la subclase pero que no se completan en la misma.</p> |

Tabla 205. Información devuelta para WLM_GET_SERVICE_SUBCLASS_STATS_V97 (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------------|---------------|--|
| REQUEST_EXEC_TIME_TOTAL | BIGINT | Suma de los tiempos de ejecución para peticiones asociadas a esta subclase de servicio desde el último restablecimiento. Las unidades son milisegundos. Si el parámetro COLLECT AGGREGATE REQUEST DATA en la clase de servicio está establecido en NONE, el valor de esta columna es NULL. Este total se calcula a partir del histograma de tiempos de ejecución de peticiones y puede resultar impreciso si el histograma no se ha dimensionado correctamente para dar cabida a los datos. Se devuelve el valor -1 si alguno de los valores cae dentro de la bandeja del último histograma. El tiempo total de ejecución de una subclase de servicio no se ve afectada por peticiones que pasan a través de la subclase pero que no se completan en la misma. |
| ACT_REMAPPED_IN | BIGINT | Número de actividades que se han correlacionado con esta subclase de servicio por una acción REMAP ACTIVITY de umbral desde el último restablecimiento. |
| ACT_REMAPPED_OUT | BIGINT | Número de actividades que se han correlacionado fuera de esta subclase de servicio por una acción REMAP ACTIVITY de umbral desde el último restablecimiento. |
| CONCURRENT_WLO_TOP | INTEGER | Número más alto de apariciones simultáneas de una determinada carga de trabajo de esta partición desde el último restablecimiento. |
| UOW_TOTAL_TIME_TOP | BIGINT | Reservado para una utilización futura. |

WLM_GET_SERVICE_SUPERCLASS_STATS - Devolver estadísticas de superclases de servicio

La función WLM_GET_SERVICE_SUPERCLASS_STATS devuelve estadísticas básicas para una o varias superclases de servicio.

Sintaxis

```

▶▶—WLM_GET_SERVICE_SUPERCLASS_STATS—(—nombre_superclase_servicio—, —————▶
▶—númparticiónbd—)—————▶▶
    
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombre_superclase_servicio

Argumento de entrada de tipo VARCHAR(128) que especifica el nombre de una superclase de servicio de la base de datos actualmente conectada. Si el argumento es nulo o una serie vacía, se recuperan datos correspondientes a todas las superclases de la base de datos.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número de partición válido en la misma instancia que la base de datos conectada actualmente. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos. Si se especifica el valor nulo, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE sobre la función WLM_GET_SERVICE_SUPERCLASS_STATS.

Ejemplo

La siguiente consulta muestra las estadísticas básicas correspondientes a todas las superclases de servicio del sistema de todas las particiones de la base de datos:

```
SELECT SUBSTR(SERVICE_SUPERCLASS_NAME, 1, 26) SERVICE_SUPERCLASS_NAME,  
       DBPARTITIONNUM,  
       LAST_RESET,  
       CONCURRENT_CONNECTION_TOP CONCURRENT_CONN_TOP  
FROM TABLE(WLM_GET_SERVICE_SUPERCLASS_STATS(' ', -2)) as SCSTATS
```

Un ejemplo de la salida es el siguiente:

```
SERVICE_SUPERCLASS_NAME  DBPARTITIONNUM ...  
-----  
SYSDEFAULTSYSTEMCLASS      0 ...  
SYSDEFAULTMAINTENANCECLASS 0 ...  
SYSDEFAULTUSERCLASS        0 ...  
... LAST_RESET              CONCURRENT_CONN_TOP  
... -----  
... 2006-09-05-09.38.44.396788 0  
... 2006-09-05-09.38.44.396795 0  
... 2006-09-05-09.38.44.396796 1
```

Nota sobre uso

La función de tabla WLM_GET_SERVICE_SUPERCLASS_STATS devuelve una fila de datos por superclase de servicio y por partición. La función no realiza ninguna agregación de datos entre superclases de servicio (de una partición) ni entre particiones (para una o más superclases de servicio). Sin embargo, puede utilizar consultas de SQL para agregar datos, como se muestra en el ejemplo anterior.

Información devuelta

Tabla 206. Información devuelta para WLM_GET_SERVICE_SUPERCLASS_STATS

| Nombre de columna | Tipo de datos | Descripción |
|-------------------------|---------------|---|
| SERVICE_SUPERCLASS_NAME | VARCHAR(128) | Nombre de la superclase de servicio de la que se ha recopilado este registro. |
| DBPARTITIONNUM | SMALLINT | Número de partición de la que se ha recopilado este registro. |

Tabla 206. Información devuelta para WLM_GET_SERVICE_SUPERCLASS_STATS (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|---------------------------|---------------|--|
| LAST_RESET | TIMESTAMP | Hora en que se restauraron las estadísticas por última vez. Se pueden producir cuatro sucesos que causarían un restablecimiento de estadísticas: <ul style="list-style-type: none"> • Se invoca el procedimiento WLM_COLLECT_STATS. • El parámetro de configuración wlm_collect_int genera una recopilación y un restablecimiento. • Se reactiva la base de datos. • Se modifica la superclase de servicio para la que se notifican estadísticas y se confirma el cambio. La indicación de fecha y hora de LAST_RESET está en el horario local. |
| CONCURRENT_CONNECTION_TOP | INTEGER | Número más alto de conexiones simultáneas de coordinación en esta clase desde el último restablecimiento. |

WLM_GET_WORK_ACTION_SET_STATS - Devolver estadísticas de conjunto de acciones de trabajo

La función WLM_GET_WORK_ACTION_SET_STATS devuelve las estadísticas correspondientes a un conjunto de acciones de trabajo.

Sintaxis

```

▶▶—WLM_GET_WORK_ACTION_SET_STATS—(—nombre_conjunto_acciones_trabajo—, —————▶
▶—númparticiónbd—)—————▶▶

```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombre_conjunto_acciones_trabajo

Argumento de entrada de tipo VARCHAR (128) que especifica el conjunto de acciones de trabajo para el que hay que devolver estadísticas. Si el argumento es nulo o una serie vacía, se devuelven estadísticas correspondientes a todos los conjuntos de acción de trabajo.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número de partición válido en la misma instancia que la base de datos conectada actualmente. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos. Si se especifica el valor nulo, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE sobre la función WLM_GET_WORK_ACTION_SET_STATS.

Ejemplo

Supongamos que hay tres clases de trabajo: ReadClass, WriteClass y LoadClass. Hay una acción de trabajo asociada a ReadClass y una asociada a LoadClass, pero no hay ninguna acción de trabajo asociada a WriteClass. En la partición 0, el número de actividades que está actualmente en ejecución o en cola es el siguiente:

- Clase ReadClass: ocho
- Clase WriteClass: cuatro
- Clase LoadClass: dos
- Sin asignar: tres

```
ELECT SUBSTR(WORK_ACTION_SET_NAME,1,18) AS WORK_ACTION_SET_NAME,
      SUBSTR(CHAR(DBPARTITIONNUM),1,4) AS PART,
      SUBSTR(WORK_CLASS_NAME,1,15) AS WORK_CLASS_NAME,
      LAST_RESET,
      SUBSTR(CHAR(WLO_ACT_TOTAL),1,14) AS ACT_TOTAL
FROM TABLE(WLM_GET_WORK_ACTION_SET_STATS
(CAST(NULL AS VARCHAR(128)), -2)) AS WASSTATS
ORDER BY WORK_ACTION_SET_NAME, WORK_CLASS_NAME, PART
```

Un ejemplo de la salida es el siguiente. Puesto que no hay ninguna acción de trabajo asociada a la clase de trabajo WriteClass, las cuatro actividades a las que se aplican se cuentan en la clase artificial indicada con un asterisco (*) en la salida. Las tres actividades que no estaban asignadas a ninguna clase de trabajo también se incluyen en la clase artificial.

| WORK_ACTION_SET_NAME | PART | WORK_CLASS_NAME | LAST_RESET | ACT_TOTAL |
|----------------------|------|-----------------|----------------------------|-----------|
| AdminActionSet | 0 | ReadClass | 2005-11-25-18.52.49.343000 | 8 |
| AdminActionSet | 1 | ReadClass | 2005-11-25-18.52.50.478000 | 0 |
| AdminActionSet | 0 | LoadClass | 2005-11-25-18.52.49.343000 | 2 |
| AdminActionSet | 1 | LoadClass | 2005-11-25-18.52.50.478000 | 0 |
| AdminActionSet | 0 | * | 2005-11-25-18.52.49.343000 | 7 |
| AdminActionSet | 1 | * | 2005-11-25-18.52.50.478000 | 0 |

Información devuelta

Tabla 207. Información devuelta para WLM_GET_WORK_ACTION_SET_STATS

| Nombre de columna | Tipo de datos | Descripción |
|----------------------|---------------|--|
| WORK_ACTION_SET_NAME | VARCHAR(128) | Nombre del conjunto de acciones de trabajo. Se devuelve un nombre solamente si se habilita el conjunto de acciones de trabajo. |
| DBPARTITIONNUM | SMALLINT | Número de partición de la que se ha recopilado este registro. |
| LAST_RESET | TIMESTAMP | Hora en que se restauraron las estadísticas por última vez. Se pueden producir cuatro sucesos que causarán un restablecimiento de estadísticas: <ul style="list-style-type: none"> • Se invoca el procedimiento WLM_COLLECT_STATS. • El parámetro de configuración wlm_collect_int genera una recopilación y un restablecimiento. • Se reactiva la base de datos. • Se modifica el conjunto de acciones de trabajo para el que se notifican estadísticas y se confirma el cambio. La indicación de fecha y hora de LAST_RESET está en el horario local. |

Tabla 207. Información devuelta para WLM_GET_WORK_ACTION_SET_STATS (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| WORK_CLASS_NAME | VARCHAR(128) | Nombre de la clase de trabajo relacionada con el conjunto de acciones de trabajo especificado. Se devuelve un nombre de clase de trabajo solamente si se habilita una acción de trabajo asociada con la clase de trabajo. El asterisco (*) representa una clase de trabajo artificial creada para contar todas las actividades que no pertenecen a las otras clases de trabajo a las que el usuario ha asociado una o varias acciones de trabajo. |
| ACT_TOTAL | BIGINT | Número de actividades de cualquier nivel de anidamiento que se han asignado a la clase de trabajo especificada por WORK_CLASS_NAME. |

WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES_V97 - Devolver una lista de actividades

La función WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES_V97 devuelve la lista de todas las actividades enviadas por la aplicación especificada en la partición especificada y que aún no se han completado.

Sintaxis

```

▶▶—WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES_V97—(—descriptorcontexto_aplicación—, —————▶
▶—númparticiónbd—)—————▶▶

```

El esquema es SYSPROC.

Parámetros de la función de tabla

descriptorcontexto_aplicación

Argumento de entrada de tipo BIGINT que especifica un descriptor de contexto de aplicación para el que se debe devolver una lista de actividades. Si el argumento es nulo, se recuperan datos correspondientes a todas las aplicaciones de la base de datos.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número de partición válido en la misma instancia que la base de datos conectada actualmente. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos. Si se especifica el valor nulo, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE sobre la función
WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES_V97.

Ejemplos

Ejemplo 1: Actividades actualmente en ejecución con un descriptor de contexto de aplicación

Después de identificar el descriptor de contexto de aplicación, se pueden buscar todas las actividades que se ejecutan actualmente en dicha aplicación. Por ejemplo, supongamos que un administrador desea tener una lista de las actividades de una aplicación cuyo descriptor de contexto de aplicación, determinado mediante el mandato LIST APPLICATIONS, es 1. El administrador ejecuta la consulta siguiente:

```
SELECT SUBSTR(CHAR(COORD_PARTITION_NUM),1,5) AS COORD,
       SUBSTR(CHAR(DBPARTITIONNUM),1,4) AS PART,
       SUBSTR(CHAR(UOW_ID),1,5) AS UOWID,
       SUBSTR(CHAR(ACTIVITY_ID),1,5) AS ACTID,
       SUBSTR(CHAR(PARENT_UOW_ID),1,8) AS PARUOWID,
       SUBSTR(CHAR(PARENT_ACTIVITY_ID),1,8) AS PARACTID,
       ACTIVITY_TYPE AS ACTTYPE,
       SUBSTR(CHAR(NESTING_LEVEL),1,7) AS NESTING
FROM TABLE(WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES_V97(1, -2)) AS WLOACTS
ORDER BY PART, UOWID, ACTID
```

Un ejemplo de resultado de la consulta es el siguiente:

```
COORD PART UOWID ACTID PARUOWID PARACTID ACTTYPE NESTING
-----
0 0 2 3 - - CALL 0
0 0 2 5 2 3 READ_DML 1
```

Ejemplo 2: Actividades actualmente en ejecución en el sistema

La consulta siguiente une la salida de WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES_V97 con la salida de MON_GET_PKG_CACHE_STMT en EXECUTABLE_ID para proporcionar texto de sentencia para todas las actividades SQL que se están ejecutando actualmente en el sistema:

```
SELECT t.application_handle,
       t.uow_id,
       t.activity_id,
       varchar(p.stmt_text, 256) as stmt_text
FROM table(wlm_get_workload_occurrence_activities_v97(NULL, -1)) as t,
     table(mon_get_pkg_cache_stmt(NULL, NULL, NULL, -1)) as p
WHERE t.executable_id = p.executable_id
```

Un ejemplo de la salida es el siguiente:

```
APPLICATION_HANDLE UOW_ID ACTIVITY_ID STMT_TEXT
-----
1 1 1 SELECT * FROM SYSCAT.TABLES
47 1 36 INSERT INTO T1 VALUES(123)
```

Información devuelta

Tabla 208. Información devuelta por WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES_V97

| Nombre de columna | Tipo de datos | Descripción |
|--------------------|---------------|---|
| APPLICATION_HANDLE | BIGINT | ID exclusivo a nivel del sistema para la aplicación. En una base de datos de una sola partición, este identificador consiste en un recuento de 16 bits. En una base de datos de varias particiones, este identificador consiste en el número de partición de coordinación concatenado con un recuento de 16 bits. Además, este identificador es el mismo en cada partición en la que la aplicación realiza una conexión secundaria. |
| DBPARTITIONNUM | SMALLINT | Número de partición de la que se ha recopilado este registro. |

Tabla 208. Información devuelta por WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES_V97 (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|---------------------|---------------|--|
| COORD_PARTITION_NUM | SMALLINT | Número de partición de coordinación de la actividad. |
| LOCAL_START_TIME | TIMESTAMP | La hora a la que esta actividad ha empezado a funcionar en la partición. El valor de la columna es nulo si la actividad ha entrado en el sistema pero está en cola y aún no se ha empezado a ejecutar. |
| UOW_ID | INTEGER | ID único de la unidad de trabajo original en la que se ha iniciado la actividad. |
| ACTIVITY_ID | INTEGER | ID de actividad exclusivo dentro de una unidad de trabajo. |
| PARENT_UOW_ID | INTEGER | ID único de la unidad de trabajo original en la que se ha iniciado la actividad padre de la actividad. El valor de esta columna es nulo si la actividad no tiene actividad padre o si está en una partición remota. |
| PARENT_ACTIVITY_ID | INTEGER | ID exclusivo de actividad dentro de una unidad de trabajo correspondiente a la actividad padre de la actividad cuyo ID está especificado por ACTIVITY_ID. El valor de esta columna es nulo si la actividad no tiene actividad padre o si está en una partición remota. |
| ACTIVITY_STATE | VARCHAR(32) | Estado de la actividad. Los valores posibles son los siguientes: <i>CANCEL_PENDING</i> La actividad se ha cancelado porque no había ningún agente trabajando de forma activa en una petición correspondiente a la actividad. La siguiente vez que se envíe una petición como parte de la actividad, la actividad se cancelará y se generará un error SQL4725N. <i>EXECUTING</i> Los agentes están trabajando de forma activa en una petición correspondiente a la actividad. <i>IDLE</i> No hay ningún agente procesando de forma activa una petición correspondiente a la actividad. <i>INITIALIZING</i> La actividad se ha enviado pero aún no se ha empezado a ejecutar. Durante el estado de inicialización, se aplican umbrales de predicción a la actividad para determinar si se permitirá que se ejecute la actividad. |

Tabla 208. Información devuelta por WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES_V97 (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------------------|---------------|---|
| ACTIVITY_STATE (continuación) | VARCHAR(32) | <p>Estado de la actividad. Los valores posibles son los siguientes:</p> <p><i>QP_CANCEL_PENDING</i> Este estado es el mismo que el estado <i>CANCEL_PENDING</i>, excepto que la actividad ha sido cancelada por Query Patroller en lugar de mediante el procedimiento <i>WLM_CANCEL_ACTIVITY</i>.</p> <p><i>QP_QUEUED</i> Query Patroller ha colocado la actividad en cola.</p> <p><i>QUEUED</i> Un umbral de colas de gestión de carga de trabajo pone en cola la actividad. En un entorno de característica de particionamiento de base de datos (DPF), es posible que este estado indique que el agente coordinador ha realizado un RPC en la partición de catálogo para obtener tiquets de umbral y aún no ha recibido una respuesta. Este estado puede indicar que un umbral de colas de gestión de carga de trabajo ha colocado en cola la actividad o, si no ha transcurrido mucho tiempo, puede indicar que la actividad está en proceso de obtener sus tiquets. Para obtener una imagen más precisa de si la actividad se está colocando realmente en cola, determine qué agente está trabajando en la actividad y busque si el valor <i>EVENT_OBJECT</i> del objeto de la partición de catálogo tiene el valor <i>WLM_QUEUE</i>.</p> <p><i>TERMINATING</i> Ha finalizado la ejecución de la actividad y se está eliminando del sistema.</p> |

Tabla 208. Información devuelta por WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES_V97 (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-----------------------------|---------------|--|
| ACTIVITY_TYPE | VARCHAR(32) | Tipo de actividad. Los valores posibles son los siguientes: <ul style="list-style-type: none"> • CALL • DDL • LOAD • OTHER • READ_DML • WRITE_DML <p>Consulte “Identificar tipos de trabajo con clases de trabajo” en <i>Workload Manager Guide and Reference</i> para ver una descripción de los distintos tipos de sentencias de SQL asociados a cada tipo de actividad.</p> |
| NESTING_LEVEL | INTEGER | Profundidad a la que se anida esta actividad dentro de la actividad principal. |
| INVOCATION_ID | INTEGER | ID de invocación, que distingue una determinada invocación de esta actividad de otras del mismo nivel de anidamiento. |
| ROUTINE_ID | INTEGER | ID único de la rutina. |
| UTILITY_ID | INTEGER | Uno de los valores siguientes: <ul style="list-style-type: none"> • Si la actividad es un programa de utilidad, el valor es el ID del programa de utilidad. • Si la actividad no es un programa de utilidad, el valor es nulo. |
| SERVICE_CLASS_ID | INTEGER | ID exclusivo de la clase de servicio a la que pertenece esta actividad. |
| DATABASE_WORK_ACTION_SET_ID | INTEGER | Uno de los valores siguientes: <ul style="list-style-type: none"> • Si esta actividad se ha clasificado en una clase de trabajo de ámbito de base de datos, el valor es el ID del conjunto de la clase de trabajo del que esta clase de trabajo es miembro. • Si esta actividad no se ha clasificado en una clase de trabajo de ámbito de base de datos, el valor es nulo. |
| DATABASE_WORK_CLASS_ID | INTEGER | Uno de los valores siguientes: <ul style="list-style-type: none"> • Si esta actividad se ha clasificado en una clase de trabajo de ámbito de base de datos, el valor es el ID del conjunto de la clase de trabajo. • Si esta actividad no se ha clasificado en una clase de trabajo de ámbito de base de datos, el valor es nulo. |

Tabla 208. Información devuelta por WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES_V97 (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|----------------------------------|--------------------------|---|
| SERVICE_CLASS_WORK_ACTION_SET_ID | INTEGER | Uno de los valores siguientes: <ul style="list-style-type: none"> • Si esta actividad se ha clasificado en una clase de trabajo de ámbito de clase de servicio, el valor es el ID del conjunto de la acción de trabajo asociado al conjunto de la clase de trabajo al que pertenece esta clase de trabajo. • Si esta actividad no se ha clasificado en una clase de trabajo de ámbito de clase de servicio, el valor es nulo. |
| SERVICE_CLASS_WORK_CLASS_ID | INTEGER | Uno de los valores siguientes: <ul style="list-style-type: none"> • Si esta actividad se ha clasificado en una clase de trabajo de ámbito de clase de servicio, el valor es el ID de la clase de trabajo asociado a esta actividad. • Si esta actividad no se ha clasificado en una clase de trabajo de ámbito de clase de servicio, el valor es nulo. |
| EXECUTABLE_ID | VARCHAR(32) FOR BIT DATA | Símbolo binario opaco generado en el servidor de datos que identifica de forma exclusiva la sección. Se puede usar el ID ejecutable como entrada para distintas interfaces de supervisión para obtener datos sobre la sección. Para actividades que no son SQL, como LOAD, se devuelve un valor NULL. |
| TOTAL_CPU_TIME | BIGINT | Reservado para una utilización futura. |
| ROWS_READ | BIGINT | Reservado para una utilización futura. |
| ROWS_RETURNED | BIGINT | Reservado para una utilización futura. |
| QUERY_COST_ESTIMATE | BIGINT | Reservado para una utilización futura. |
| DIRECT_READS | BIGINT | Reservado para una utilización futura. |
| DIRECT_WRITES | BIGINT | Reservado para una utilización futura. |

Función de tabla WLM_GET_WORKLOAD_STATS_V97 - Devolver estadísticas de carga de trabajo

La función WLM_GET_WORKLOAD_STATS_V97 devuelve una fila de estadísticas de carga de trabajo correspondientes a cada combinación de nombre de carga de trabajo y número de partición de base de datos.

Sintaxis

►►—WLM_GET_WORKLOAD_STATS_V97—(—nombre_cargatrabajo—,—númparticiónbd—)————►►

El esquema es SYSPROC.

Parámetros de la función de tabla

nombre_cargatrabajo

Argumento de entrada de tipo VARCHAR(128) que especifica una carga de

trabajo para la que se tienen que devolver estadísticas. Si el argumento es NULL o una serie vacía, se devuelven estadísticas correspondientes a todas las cargas de trabajo.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica el número de partición en la misma instancia que la base de datos conectada actualmente. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos. Si se especifica un valor nulo, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE sobre la función WLM_GET_WORKLOAD_STATS_V97.

Ejemplo

La consulta siguiente muestra estadísticas para las cargas de trabajo:

```
SELECT SUBSTR(WORKLOAD_NAME,1,18) AS WL_DEF_NAME,
       SUBSTR(CHAR(DBPARTITIONNUM),1,4) AS PART,
       COORD_ACT_LIFETIME_TOP,
       COORD_ACT_LIFETIME_AVG,
       COORD_ACT_LIFETIME_STDDEV
FROM TABLE(WLM_GET_WORKLOAD_STATS_V97(CAST(NULL AS VARCHAR(128)), -2)) AS WLSTATS
ORDER BY WL_DEF_NAME, PART
```

Un ejemplo de resultado de la consulta es el siguiente:

```
WL_DEF_NAME      PART COORD_ACT_LIFETIME_TOP ...
-----
SYSDEFAULTADWORKL 0          -1 ...
SYSDEFAULTUSERWORK 0          -1 ...
WL1                0           2 ...
... COORD_ACT_LIFETIME_AVG COORD_ACT_LIFETIME_STDDEV
...
... -1.0000000000000000E+000 -1.0000000000000000E+000
... -1.0000000000000000E+000 -1.0000000000000000E+000
... +2.5600000000000000E+000 +6.0000000000000001E-002
```

Nota sobre uso

La función no realiza ninguna agregación de datos entre cargas de trabajo, particiones o clases de servicio. Sin embargo, puede utilizar consultas de SQL para agregar datos.

Información devuelta

Tabla 209. Información devuelta por WLM_GET_WORKLOAD_STATS_V97

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| WORKLOAD_NAME | VARCHAR(128) | Nombre de la carga de trabajo de la que se ha recopilado este registro. |
| DBPARTITIONNUM | SMALLINT | Número de la partición de la que se ha recopilado este registro. |

Tabla 209. Información devuelta por WLM_GET_WORKLOAD_STATS_V97 (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|---------------------------|---------------|---|
| LAST_RESET | TIMESTAMP | <p>Hora en que se restauraron las estadísticas por última vez. Se pueden producir cuatro sucesos que causarán un restablecimiento de estadísticas:</p> <ul style="list-style-type: none"> • Se invoca el procedimiento WLM_COLLECT_STATS. • El parámetro de configuración wlm_collect_int genera una recopilación y un restablecimiento. • Se reactiva la base de datos. • Se modifica la carga de trabajo para la que se notifican estadísticas y se confirma el cambio. <p>La indicación de fecha y hora de LAST_RESET está en el horario local.</p> |
| CONCURRENT_WLO_TOP | INTEGER | Número más alto de apariciones simultáneas de una determinada carga de trabajo de esta partición desde el último restablecimiento. |
| CONCURRENT_WLO_ACT_TOP | INTEGER | Número más alto de actividades simultáneas (tanto las de coordinación como las anidadas) en estado de ejecución (lo que incluye las desocupadas y las en espera) o bien en cola que se ha alcanzado en cualquier aparición de esta carga de trabajo desde el último restablecimiento. El valor de la columna lo actualiza cada aparición de carga de trabajo al final de su unidad de trabajo. |
| COORD_ACT_COMPLETED_TOTAL | BIGINT | Número total de actividades de coordinación a cualquier nivel de anidamiento asignadas a cualquier aparición de esta carga de trabajo que se han completado desde el último restablecimiento. El valor de esta columna lo actualiza cada aparición de carga de trabajo al final de su unidad de trabajo. |
| COORD_ACT_ABORTED_TOTAL | BIGINT | Número total de actividades de coordinación a cualquier nivel de anidamiento asignadas a cualquier aparición de esta carga de trabajo que han terminado anormalmente antes de finalizar desde el último restablecimiento. El valor de esta columna lo actualiza cada aparición de carga de trabajo al final de su unidad de trabajo. |

Tabla 209. Información devuelta por WLM_GET_WORKLOAD_STATS_V97 (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|---------------------------|---------------|--|
| COORD_ACT_REJECTED_TOTAL | BIGINT | <p>Número total de actividades de coordinación a cualquier nivel de anidamiento asignadas a cualquier aparición de esta carga de trabajo que se han rechazado antes de la ejecución desde el último restablecimiento. El valor de esta columna lo actualiza cada aparición de carga de trabajo al final de su unidad de trabajo.</p> <p>Una actividad se cuenta como rechazada cuando se impide su ejecución mediante una acción que impide el trabajo de ejecución o mediante un umbral de predicción. A diferencia de la columna del mismo nombre de la función WLM_GET_SERVICE_SUBCLASS_STATS_V97, esta columna WLM_GET_WORKLOAD_STATS_V97 también incluye la cantidad de rechazos producidos antes de que una actividad se pueda asignar a una clase de servicio. Por ejemplo, este tipo de rechazo se produce cuando una actividad viola el umbral ConcurrentWorkloadOccurrences.</p> |
| WLO_COMPLETED_TOTAL | BIGINT | Número de apariciones de carga de trabajo que hay que completar desde el último restablecimiento. |
| COORD_ACT_LIFETIME_TOP | BIGINT | Indicador de nivel alto correspondiente a la duración de actividad de coordinación, recopilado en todos los niveles de anidamiento. Las unidades son milisegundos. Si el parámetro COLLECT AGGREGATE ACTIVITY DATA de la clase de servicio está establecido en NONE, el valor de la columna es nulo. |
| COORD_ACT_LIFETIME_AVG | DOUBLE | Media aritmética de duración para actividades de coordinación completadas o abortadas a nivel de anidamiento 0 asociadas a esta carga de trabajo. Las unidades son milisegundos. Si se ha desbordado el promedio del que se hace un seguimiento interno, se devuelve el valor -2. Si el parámetro COLLECT AGGREGATE ACTIVITY DATA de la carga de trabajo está establecido en NONE, el valor de la columna es nulo. |
| COORD_ACT_LIFETIME_STDDEV | DOUBLE | Desviación estándar de duración para actividades de coordinación completadas o abortadas a nivel de anidamiento 0 asociadas a esta carga de trabajo. Las unidades son milisegundos. Si el parámetro COLLECT AGGREGATE ACTIVITY DATA de la carga de trabajo está establecido en NONE, el valor de la columna es nulo. Esta desviación estándar se calcula a partir del histograma de duración de las actividades de coordinación y puede resultar imprecisa si el histograma no se ha dimensionado correctamente para dar cabida a los datos. Si alguno de los valores cae dentro de la bandeja del último histograma, se devuelve el valor -1. |

Tabla 209. Información devuelta por WLM_GET_WORKLOAD_STATS_V97 (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-----------------------------|---------------|---|
| COORD_ACT_EXEC_TIME_AVG | DOUBLE | Media aritmética de los tiempos de ejecución para actividades de coordinación completadas o abortadas a nivel de anidamiento 0 asociadas a esta carga de trabajo. Las unidades son milisegundos. Si se ha desbordado el promedio del que se hace un seguimiento interno, se devuelve el valor -2. Si el parámetro COLLECT AGGREGATE ACTIVITY DATA de la carga de trabajo está establecido en NONE, el valor de la columna es nulo. |
| COORD_ACT_EXEC_TIME_STDDEV | DOUBLE | Desviación estándar de los tiempos de ejecución para actividades de coordinación completadas o abortadas a nivel de anidamiento 0 asociadas a esta carga de trabajo. Las unidades son milisegundos. Esta desviación estándar se calcula a partir del histograma de tiempo de ejecución de las actividades de coordinación y puede resultar imprecisa si el histograma no se ha dimensionado correctamente para dar cabida a los datos. Si alguno de los valores cae dentro de la bandeja del último histograma, se devuelve el valor -1. Si el parámetro COLLECT AGGREGATE ACTIVITY DATA de la carga de trabajo está establecido en NONE, el valor de la columna es nulo. |
| COORD_ACT_QUEUE_TIME_AVG | DOUBLE | Media aritmética del tiempo de cola para actividades de coordinación completadas o abortadas a nivel de anidamiento 0 asociadas a esta carga de trabajo. Las unidades son milisegundos. Si se ha desbordado el promedio del que se hace un seguimiento interno, se devuelve el valor -2. Si el parámetro COLLECT AGGREGATE ACTIVITY DATA de la clase de servicio está establecido en NONE, el valor de la columna es nulo. |
| COORD_ACT_QUEUE_TIME_STDDEV | DOUBLE | Desviación estándar del tiempo de cola para actividades de coordinación completadas o abortadas a nivel de anidamiento 0 asociadas a esta carga de trabajo. Las unidades son milisegundos. Si el parámetro COLLECT AGGREGATE ACTIVITY DATA de la carga de trabajo está establecido en NONE, el valor de la columna es nulo. Esta desviación estándar se calcula a partir del histograma de tiempo de cola de las actividades de coordinación y puede resultar imprecisa si el histograma no se ha dimensionado correctamente para dar cabida a los datos. Si alguno de los valores cae dentro de la bandeja del último histograma, se devuelve el valor -1. |
| UOW_TOTAL_TIME_TOP | BIGINT | Reservado para una utilización futura. |

Procedimiento WLM_SET_CLIENT_INFO - Establecer información de cliente

El procedimiento WLM_SET_CLIENT_INFO establece información de cliente asociada con la conexión actual en el servidor DB2.

Mediante la utilización de este procedimiento, puede establecer el ID de usuario del cliente, el nombre de aplicación, el nombre de estación de trabajo, la información de contabilidad o la información de carga de trabajo en el servidor DB2. La invocación de este procedimiento cambia los valores almacenados para esta conexión de los campos de información de cliente de supervisor de procesador de transacciones (TP) pertinentes y los valores de registro especial.

Los campos de información de cliente se utilizan en el servidor DB2 para determinar la identidad de la aplicación o del usuario que utiliza actualmente la conexión. Los campos de información de cliente para una conexión se tienen en cuenta durante la evaluación de carga de trabajo de DB2 y también aparecen en cualquier instantánea de aplicación o registros de comprobación de DB2 que se genere para esta conexión.

A diferencia de la API `sqleaseti`, este procedimiento no establece información de cliente en el cliente sino que establece los atributos de cliente correspondientes en el servidor DB2. Por consiguiente, no se puede utilizar la API `sqleqry` para consultar la información de cliente que se establece en el servidor DB2 utilizando este procedimiento.

Los valores de datos proporcionados con el procedimiento se convierten en la página de códigos de base de datos apropiada antes de almacenarse en los campos de Supervisor de TP relacionados o los registros especiales. Los valores de datos que sobrepasan el tamaño máximo soportado después de la conversión a la página de códigos de la base de datos se truncarán antes de almacenarse en el servidor. Los campos de Supervisor de TP y los registros especiales devolverán los valores truncados cuando se consulten esos valores almacenados.

El procedimiento `WLM_SET_CLIENT_INFO` no está bajo el control de la transacción y los cambios en la información de cliente efectuados por el procedimiento son independientes de la acción de confirmar o retrotraer unidades de trabajo. Sin embargo, debido a que la reevaluación de la carga de trabajo se produce al principio de la siguiente unidad de trabajo para cada aplicación, debe emitir una sentencia `COMMIT` o bien `ROLLBACK` para que los cambios en la información de cliente sean efectivos.

Si las aplicaciones deben resultar portátiles entre servidores de datos de IBM, utilice una de las interfaces de programación de aplicaciones de base de datos para establecer información de cliente en el servidor en vez de utilizar el procedimiento `WLM_SET_CLIENT_INFO`. El procedimiento sólo está disponible para DB2 Database para Linux, UNIX y Windows.

Sintaxis

```
►►—WLM_SET_CLIENT_INFO—(—id_usuario_cliente—,—nombre_estación_trabajo_cliente—,—  
►—nombre_apl_cliente—,—serie_cont_cliente—,—carga_trabajo_cliente—)—►►
```

El esquema es `SYSPROC`.

Parámetros del procedimiento

id_usuario_cliente

Argumento de entrada de tipo `VARCHAR(255)` que especifica el ID de usuario para el cliente. Si se especifica `NULL`, el valor se queda como está. Si se

especifica una serie vacía, que es el valor por omisión, el ID de usuario del cliente se restablece en el valor por omisión, que es un espacio en blanco.

nombre_estación_trabajo_cliente

Argumento de entrada de tipo VARCHAR(255) que especifica el nombre de estación de trabajo para el cliente. Si se especifica NULL, el valor se queda como está. Si se especifica una serie vacía, que es el valor por omisión, el nombre de estación de trabajo del cliente se restablece en el valor por omisión, que es un espacio en blanco.

nombre_apl_cliente

Argumento de entrada de tipo VARCHAR(255) que especifica el nombre de aplicación del cliente. Si se especifica NULL, el valor se queda como está. Si se especifica una serie vacía, que es el valor por omisión, el nombre de aplicación del cliente se restablece en el valor por omisión, que es un espacio en blanco.

serie_cont_cliente

Argumento de entrada de tipo VARCHAR(255) que especifica la serie de contabilidad para el cliente. Si se especifica NULL, el valor se queda como está. Si se especifica una serie vacía, que es el valor por omisión, la serie de contabilidad del cliente se restablece en el valor por omisión, que es un espacio en blanco.

carga_trabajo_cliente

Argumento de entrada de tipo VARCHAR(255) que especifica la modalidad de asignación de carga de trabajo para el cliente. Si se especifica NULL, el valor se queda como está. Los valores son los siguientes:

SYSDEFAULTADMWORKLOAD

Especifica que la conexión de base de datos se asignará a SYSDEFAULTADMWORKLOAD, lo que permitirá que los usuarios con autorización ACCESSCTRL, DATAACCESS, DBADM, SECADM o WLMADM ignoren la evaluación de carga de trabajo normal.

AUTOMATIC

Especifica que la base de datos se asignará a una carga de trabajo escogida mediante la evaluación de la carga de trabajo que el servidor efectúa automáticamente.

Nota: El argumento *carga_trabajo_cliente* es sensible a las mayúsculas y minúsculas.

Autorización

Privilegio EXECUTE en el procedimiento WLM_SET_CLIENT_INFO.

Ejemplos

La siguiente invocación de procedimiento establece el ID de usuario, el nombre de estación de trabajo, el nombre de aplicación, la serie de contabilidad y la modalidad de asignación de carga de trabajo para el cliente:

```
CALL SYSPROC.WLM_SET_CLIENT_INFO('db2user', 'machine.torolab.ibm.com',  
    'auditor', 'Departamento contabilidad', 'AUTOMATIC')
```

La siguiente invocación de procedimiento establece el ID de usuario en db2user2 para cliente sin establecer los demás atributos de cliente:

```
CALL SYSPROC.WLM_SET_CLIENT_INFO('db2user2', NULL, NULL, NULL, NULL)
```

La siguiente invocación de procedimiento restablece el ID de usuario para el cliente en un espacio en blanco sin modificar los valores de los demás atributos de cliente:

```
CALL SYSPROC.WLM_SET_CLIENT_INFO('', NULL, NULL, NULL, NULL)
```

Capítulo 20. Rutinas y vistas varias

Procedimiento ADMIN_COPY_SCHEMA - Copiar un esquema específico y sus objetos

El procedimiento ADMIN_COPY_SCHEMA sirve para copiar un esquema específico y todos los objetos contenidos en el mismo. Los objetos del nuevo esquema de destino se crearán con los mismos nombres de objeto que los objetos del esquema de origen, pero con el calificador del esquema de destino. El procedimiento ADMIN_COPY_SCHEMA se puede utilizar para copiar tablas con o sin los datos de las tablas originales.

Sintaxis

```
►► ADMIN_COPY_SCHEMA (—esquemaorigen—, —esquemadestino—, —————►  
► modalidadcopia—, —propietarioobjeto—, —esptablaorigen—, —————►  
► esptabladestino—, —esquemataberrores—, —taberrores—) —————►►
```

El esquema es SYSPROC.

Parámetros del procedimiento

esquemaorigen

Argumento de entrada de tipo VARCHAR(128) que especifica el nombre del esquema cuyos objetos se van a copiar. El nombre es sensible a mayúsculas y minúsculas.

esquemadestino

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de esquema exclusivo en el que se van a crear los objetos copiados. El nombre es sensible a mayúsculas y minúsculas. Si el nombre del esquema ya existe, la llamada al procedimiento falla y devuelve un mensaje que indica que el esquema se debe eliminar antes de invocar el procedimiento.

modalidadcopia

Argumento de entrada de tipo VARCHAR(128) que especifica la modalidad de la operación de copia. Las opciones válidas son:

- 'DDL': crear copias vacías para todos los objetos soportados del esquema de origen.
- 'COPY': crear copias vacías de todos los objetos del esquema de origen y luego cargar cada tabla de esquema de destino con datos. La carga se realiza en modalidad 'NONRECOVERABLE'. Se debe realizar una copia de seguridad después de llamar a ADMIN_COPY_SCHEMA; de lo contrario, no se podrá acceder a las tablas copiadas tras la recuperación.
- 'COPYNO': crear copias vacías de todos los objetos del esquema de origen y luego cargar cada tabla de esquema de destino con datos. La carga se realiza en modalidad 'COPYNO'.

Nota: Si *modalidadcopia* es 'COPY' o 'COPYNO', se puede especificar un nombre de archivo totalmente calificado, como por ejemplo 'COPYNO

/home/mckeough/loadoutput', junto con el valor del parámetro *modalidadcopia*. Cuando se pasa una vía de acceso, se anotarán cronológicamente los mensajes de carga en el archivo indicado. Es necesario que el ID de usuario utilizado para las invocaciones de rutina delimitada en la instancia pueda grabar en el archivo. Si no se especifica ninguna vía de acceso, los mensajes de carga se descartarán (comportamiento por omisión).

propietarioobjeto

Argumento de entrada de tipo VARCHAR(128) que especifica el ID de autorización que se debe utilizar como el propietario de los objetos copiados. Si es nulo, el propietario será el ID de autorización del usuario que realice la operación de copia.

establaorigen

Argumento de entrada de tipo CLOB(2 M) que especifica una lista de espacios de tabla de origen para la copia, separados por comas. Se da soporte a los nombres de espacios de tabla delimitados. Para cada tabla que se cree, cualquier espacio de tabla que se encuentre en esta lista, y la definición de tabla, se convertirá en la entrada número n de la lista *esptabladestino*. Si se especifica nulo para este parámetro, los objetos nuevos se crearán utilizando los mismos espacios de tabla que utilizan los objetos de origen.

esptabladestino

Argumento de entrada de tipo CLOB(2 M) que especifica una lista de espacios de tabla de destino para la copia, separados por comas. Se da soporte a los nombres de espacios de tabla delimitados. Se debe especificar un espacio de tabla para cada entrada de la lista *esptablaorigen* de espacios de tabla. El espacio de tabla número n de la lista *esptablaorigen* se correlacionará con el espacio de tabla número n de la lista *esptabladestino* durante la reproducción DDL. Se puede especificar 'SYS_ANY' como espacio de tabla final (un nombre de espacio de tabla adicional que no corresponde con ningún nombre de la lista de origen). Cuando se encuentre 'SYS_ANY', se utilizará el algoritmo de selección de espacio de tabla por omisión cuando se creen objetos (consulte la opción IN *nombre-espacio-tablas1* de la documentación de la sentencia CREATE TABLE para obtener más información sobre el algoritmo de selección). Si se especifica nulo para este parámetro, los objetos nuevos se crearán utilizando los mismos espacios de tabla que utilizan los objetos de origen.

esquemataberrores

Argumento de entrada y de salida de tipo VARCHAR(128) que especifica el nombre de esquema de una tabla que contiene información sobre errores correspondiente a los objetos que no se han podido copiar. Esta tabla la crea para el usuario el procedimiento ADMIN_COPY_SCHEMA en el espacio de tabla SYSTOOLSPACE. Si no se ha producido ningún error, este parámetro es nulo en la salida.

taberrores

Argumento de entrada y de salida de tipo VARCHAR(128) que especifica el nombre de una tabla que contiene información sobre errores correspondiente a los objetos que no se han podido copiar. Esta tabla la crea para el usuario el procedimiento ADMIN_COPY_SCHEMA en el espacio de tabla SYSTOOLSPACE. Esta tabla es propiedad del ID de usuario que ha invocado el procedimiento. Si no se ha producido ningún error, este parámetro es nulo en la salida. Si la tabla no se puede crear o ya existe, la operación del procedimiento falla y se devuelve un mensaje de error. El usuario debe limpiar la tabla después de una llamada al procedimiento ADMIN_COPY_SCHEMA; es decir, la tabla se debe eliminar para poder reclamar el espacio que consume en SYSTOOLSPACE.

Tabla 210. Formato de la tabla de errores de ADMIN_COPY_SCHEMA

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| OBJECT_SCHEMA | VARCHAR(128) | Nombre de esquema del objeto para el que ha fallado el mandato de copia. |
| OBJECT_NAME | VARCHAR(128) | Nombre del objeto para el que ha fallado el mandato de copia. |
| OBJECT_TYPE | VARCHAR(30) | Tipo de objeto. |
| SQLCODE | INTEGER | SQLCODE del error. |
| SQLSTATE | CHAR(5) | SQLSTATE del error. |
| ERROR_TIMESTAMP | TIMESTAMP | Hora del error correspondiente a la operación que ha fallado. |
| STATEMENT | CLOB(2 M) | DDL correspondiente al objeto que ha fallado. Si el error se ha producido mientras se estaban cargando datos en una tabla de destino, este campo contiene el texto correspondiente al mandato de carga que ha fallado. |
| DIAGTEXT | CLOB(2 K) | Texto del mensaje de error correspondiente a la operación que ha fallado. |

Autorización

Para que la operación de copia de esquema sea satisfactoria, el ID de usuario que llama a este procedimiento debe tener las autorizaciones de creación de objetos adecuadas, incluidas la autorización para seleccionar en las tablas de origen y la autorización para realizar una carga. Si una tabla del esquema de origen está protegido por un control de acceso basado en etiqueta (LBAC), el ID de usuario debe tener credenciales de LBAC que le permitan crear la misma protección en la tabla de destino. Si la copia se realiza con datos, el ID de usuario debe tener también credenciales de LBAC que le permitan leer la tabla de origen y escribir dichos datos en la tabla de destino.

También se necesita el privilegio EXECUTE sobre el procedimiento ADMIN_COPY_SCHEMA.

Ejemplo

```
CALL SYSPROC.ADMIN_COPY_SCHEMA('SOURCE_SCHEMA', 'TARGET_SCHEMA',
    'COPY', NULL, 'SOURCETS1', SOURCETS2', 'TARGETTS1', TARGETTS2,
    SYS_ANY', 'ERRORSCHEMA', 'ERRORNAME')
```

Restricciones

- Para bases de datos HADR, sólo se da soporte a la *modalidad copia* DDL.
- No se da soporte a XML con COPY o COPY NO.
- Utilizar el procedimiento ADMIN_COPY_SCHEMA con la opción COPYNO sitúa los espacios de tabla en los que reside el objeto de la base de datos de destino en estado pendiente de copia de seguridad. Cuando ha finalizado la

operación de carga, las tablas de esquema de destino se establecen en estado pendiente de integridad establecida y el procedimiento ADMIN_COPY_SCHEMA emite una sentencia SET INTEGRITY para que las tablas salgan de este estado. Puesto que los espacios de tablas ya están en estado pendiente de copia de seguridad, la sentencia SET INTEGRITY falla. Para obtener información sobre cómo resolver este problema, consulte “Copiar un esquema”.

Notas de uso

- Las referencias a objetos calificados por completo dentro de los objetos que se copian no se modificarán. El procedimiento ADMIN_COPY_SCHEMA sólo modifica el esquema calificador del objeto que se crea, no los nombres de esquema que aparecen en las expresiones SQL para dichos objetos. Aquí se incluyen los objetos del tipo columnas generadas y cuerpos de activador.
- Este procedimiento no da soporte a la copia de los siguientes objetos:
 - extensiones de índice
 - apodos
 - paquetes
 - tipo, tablas
 - tipos de matriz
 - tipos estructurados definidos por el usuario (y sus funciones de transformación)
 - vistas con tipo
 - jars (archivadores de rutinas Java™)
 - tablas de etapas
- Si el esquema que se va a copiar contiene alguno de los objetos anteriores, el objeto no se copia, sino que se añade una entrada a la tabla de errores que indica que el objeto no se ha copiado.
- Cuando se copia una tabla duplicada, la nueva copia de la tabla no tiene suscripciones habilitadas. La tabla se vuelve a crear únicamente como una tabla básica.
- Para poder realizar este procedimiento, debe existir el espacio de tabla SYSTOOLSPACE. Este espacio de tabla se utiliza para albergar los metadatos que utiliza el procedimiento ADMIN_COPY_SCHEMA, así como las tablas de errores que devuelve este procedimiento. Si el espacio de tabla no existe, se devuelve un error.
- Las estadísticas correspondientes a los objetos del esquema de destino adoptan los valores por omisión.
- Si una tabla tiene una columna de identidad generada y *modalidadcopia* es 'COPY' o 'COPYNO', los valores de los datos de la tabla de origen se conservan durante la carga.
- Se crea una nueva entrada de catálogo para cada rutina externa, que hace referencia al binario de la rutina de origen original.
- Si una tabla está en estado de establecimiento de integridad pendiente al principio de la operación de copia, los datos no se cargan en la tabla de destino y se registra una entrada en *taberrores* que indica que no se han cargado los datos correspondientes a dicha tabla.
- Si una operación de carga o DDL falla, se registra una entrada en *taberrores* correspondiente a cualquier objeto que no se haya creado. Todos los objetos que se crean satisfactoriamente se conservan. Para realizar la recuperación, se puede iniciar una carga manual o se puede eliminar el nuevo esquema mediante el

procedimiento ADMIN_DROP_SCHEMA y se puede volver a llamar al procedimiento ADMIN_COPY_SCHEMA.

- Durante la reproducción DDL, el esquema por omisión se sustituye por el esquema de destino si coincide con el esquema de origen.
- La vía de acceso de función utilizada para compilar un activador, vista o función SQL es la vía de acceso utilizada para crear el objeto de origen, con la siguiente excepción: si la vía de acceso de la función del objeto contiene el nombre del esquema de origen, esta entrada de la vía de acceso se sustituye por el nombre del esquema de destino durante la reproducción DDL.
- Ejecutar múltiples procedimientos ADMIN_COPY_SCHEMA dará como resultado puntos muertos. Sólo se debe emitir una llamada de procedimiento ADMIN_COPY_SCHEMA a la vez. Los cambios realizados en tablas del esquema fuente durante el proceso de copia podrían significar que los datos del esquema de destino no son idénticos siguiendo una operación de copia.
- Debe proceder con especial precaución al copiar un esquema con tablas de un espacio de tabla de un grupo de particiones de base de datos de partición única en un espacio de tabla de un grupo de particiones de base de datos de varias particiones. A menos que se prefiera la selección de la clave de distribución automática, la clave de distribución debe definirse en las tablas antes de efectuar la operación de copia del esquema. La clave de distribución sólo se puede modificar para una tabla cuyo espacio de tabla esté asociado con un grupo de particiones de base de datos de partición única.

Consideraciones sobre la transacción

- Si se fuerza una retrotracción del procedimiento ADMIN_COPY_SCHEMA debido a un punto muerto o a un tiempo de espera excedido de bloqueo durante su proceso, el trabajo realizado en la unidad de trabajo que ha llamado al procedimiento ADMIN_COPY_SCHEMA también se retrotrae.
- Si se produce un error durante la fase DDL de la copia, todos los cambios efectuados en el esquema de destino se retrotraen a un punto de salvaguarda.
- Si *modalidadcopia* tiene el valor 'COPY' o 'COPYNO', el procedimiento ADMIN_COPY_SCHEMA se confirma cuando finaliza la fase DDL de la copia, confirmando también el trabajo realizado en la unidad de trabajo que ha llamado al procedimiento.

Procedimiento ADMIN_DROP_SCHEMA - Descartar un esquema específico y sus objetos

El procedimiento ADMIN_DROP_SCHEMA sirve para eliminar un esquema específico y todos los objetos contenidos en el mismo.

Sintaxis

```
▶▶—ADMIN_DROP_SCHEMA—(—esquema—, —modalidadeliminación—, —————▶  
▶—esquemataberrores—, —taberrores—)—————▶▶
```

El esquema es SYSPROC.

Parámetros del procedimiento

esquema

Argumento de entrada de tipo VARCHAR(128) que especifica el nombre del esquema que se va a eliminar. El nombre debe ser especificado en mayúsculas.

modalidadeliminación

Reservado para su utilización en el futuro; debe tener el valor nulo.

esquemataberrores

Argumento de entrada y de salida de tipo VARCHAR(128) que especifica el nombre de esquema de una tabla que contiene información sobre errores correspondiente a los objetos que no se han podido eliminar. El nombre es sensible a mayúsculas y minúsculas. Esta tabla la crea para el usuario el procedimiento ADMIN_DROP_SCHEMA en el espacio de tablas SYSTOOLSPACE. Si no se ha producido ningún error, este parámetro es nulo en la salida.

taberrores

Argumento de entrada y de salida de tipo VARCHAR(128) que especifica el nombre de una tabla que contiene información sobre errores correspondiente a los objetos que no se han podido eliminar. El nombre es sensible a mayúsculas y minúsculas. Esta tabla la crea para el usuario el procedimiento ADMIN_DROP_SCHEMA en el espacio de tablas SYSTOOLSPACE. Esta tabla es propiedad del ID de usuario que ha invocado el procedimiento. Si no se ha producido ningún error, este parámetro es nulo en la salida. Si la tabla no se puede crear o ya existe, la operación del procedimiento falla y se devuelve un mensaje de error. El usuario debe limpiar la tabla después de una llamada a ADMIN_DROP_SCHEMA; es decir, la tabla se debe eliminar para poder reclamar el espacio que consume en SYSTOOLSPACE.

Tabla 211. Formato de la tabla de errores de ADMIN_DROP_SCHEMA

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| OBJECT_SCHEMA | VARCHAR(128) | Nombre de esquema del objeto para el que ha fallado el mandato de eliminación. |
| OBJECT_NAME | VARCHAR(128) | Nombre del objeto para el que ha fallado el mandato de eliminación. |
| OBJECT_TYPE | VARCHAR(30) | Tipo de objeto. |
| SQLCODE | INTEGER | SQLCODE del error. |
| SQLSTATE | CHAR(5) | SQLSTATE del error. |
| ERROR_TIMESTAMP | TIMESTAMP | Hora a la que ha fallado el mandato de eliminación. |
| STATEMENT | CLOB(2 M) | DDL correspondiente al objeto que ha fallado. |
| DIAGTEXT | CLOB(2 K) | Texto del mensaje de error correspondiente al mandato de eliminación que ha fallado. |

Autorización

Se necesita autorización de eliminación sobre todos los objetos que se van a eliminar para el usuario que llama a este procedimiento.

También se necesita el privilegio EXECUTE sobre el procedimiento ADMIN_DROP_SCHEMA.

Ejemplo

```
CALL SYSPROC.ADMIN_DROP_SCHEMA('SCHNAME', NULL, 'ERRORSCHEMA', 'ERRORTABLE')
```

El ejemplo siguiente muestra los datos de salida del procedimiento.

Valor parámetros de salida

Nombre de parámetro : ERRORTABSCHEMA

Valor de parámetro : ERRORSCHEMA <-- error!

Nombre de parámetro : ERRORTAB

Valor de parámetro : ERRORTABLE <-- error!

Estado de devolución = 0

El estado de devolución no es cero sólo cuando se ha detectado un error interno (por ejemplo, si SYSTOOLSPACE no existe).

Se pueden comprobar los errores consultando la tabla de errores:

```
SELECT * FROM ERRORSCHEMA.ERRORTABLE
```

Notas de uso

- Si hay objetos en otro esquema que dependen de un objeto que se va a eliminar, se aplica la semántica de la sentencia DROP por omisión.
- Este procedimiento no da soporte a la eliminación de los siguientes objetos:
 - extensiones de índice
 - apodos
 - paquetes
 - tipo, tablas
 - tipos de matriz
 - tipos estructurados definidos por el usuario (y sus funciones de transformación)
 - vistas con tipo
 - jars (archivadores de rutinas Java)
 - tablas de etapas
- Si el esquema que se va a eliminar contiene alguno de los objetos anteriores, ni el objeto ni el esquema se eliminan y se añade una entrada a la tabla de errores que indica que el objeto no se ha eliminado.
- Para poder realizar este procedimiento, debe existir el espacio de tabla SYSTOOLSPACE. Este espacio de tablas se utiliza para albergar los metadatos que utiliza el procedimiento ADMIN_DROP_SCHEMA, así como las tablas de errores que devuelve este procedimiento. Si el espacio de tabla no existe, se devuelve un error.

Procedimiento ADMIN_MOVE_TABLE - trasladar una tabla en línea

El procedimiento almacenado ADMIN_MOVE_TABLE mueve los datos de una tabla activa a un objeto de tabla nuevo que tiene el mismo nombre, y durante el proceso los datos continúan en línea y se puede acceder a ellos. Este procedimiento almacenado crea una tabla de protocolo compuesta por filas que contienen información de estado y opciones de configuración relativas a la tabla que se va a

trasladar. El conjunto devuelto por este procedimiento está compuesto por las filas de la tabla de protocolo relacionada con la tabla que se va a mover.

Este procedimiento almacenado emplea la terminología siguiente:

Tabla fuente

Nombre de tabla original que se proporciona como parámetro al procedimiento almacenado. Se trata de la tabla que se va a mover.

Tabla de destino

Tabla creada por el procedimiento almacenado utilizando la definición de tabla proporcionada a través del procedimiento almacenado. Todos los datos de la tabla fuente se copian en esta tabla, que a continuación se renombra con el mismo nombre que la tabla fuente.

Tabla de etapas

Tabla creada por el procedimiento almacenado. La tabla de etapas almacena todos los cambios de actualización, supresión o inserción que se producen en la tabla fuente durante la ejecución del traslado de tabla. Esta tabla se descarta una vez terminado el traslado.

Sintaxis

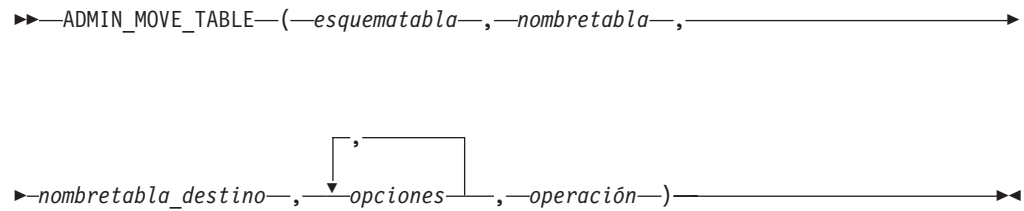
Existen dos métodos igualmente válidos para invocar ADMIN_MOVE_TABLE. El primer método permite modificar únicamente algunas partes determinadas de la definición de tabla para la tabla de destino. Por ejemplo, si tiene una definición de tabla con un tamaño considerable (varios KB) y lo único que quiere hacer es modificar el espacio de tablas para la tabla, puede hacerlo sin tener que determinar toda la sentencia CREATE TABLE necesaria para recrear la tabla fuente. Solo tendrá que cumplimentar los parámetros `espaciotablas_datos`, `espaciotablas_índice` y `espaciotablas_lob` y dejar el resto de parámetros opcionales en blanco.

El segundo método le ofrece más control y flexibilidad ya que le permite crear la tabla de destino con antelación, en lugar de utilizar el procedimiento almacenado para crear la tabla de destino. De esta forma, puede crear una tabla de destino que no sería posible obtener mediante el primer método.

Método 1:

```
►► ADMIN_MOVE_TABLE (—esquematabla—, —nombratabla—, —————►
  
► —espaciotablas_datos—, —espaciotablas_índice—, —espaciotablas_lob—, —————►
  
► —cols_mdc—, —cols_clavepart—, —part_datos—, —defcol—, —opciones—, —————►
  
► —operación—) —————►
```

Método 2:



El esquema de los dos métodos es SYSPROC.

Parámetros del procedimiento

esquematabla

Este parámetro de entrada especifica el nombre del esquema que contiene la tabla que se va a mover. Este parámetro es sensible a mayúsculas y minúsculas y tiene un tipo de datos VARCHAR(128).

nombratabla

Este parámetro de entrada especifica el nombre de la tabla que se va a mover. Este parámetro es sensible a mayúsculas y minúsculas y tiene un tipo de datos VARCHAR(128).

espaciotablas_datos

Este parámetro de entrada especifica el nuevo espacio de tablas de datos para la tabla de destino. Si se proporciona un valor, es necesario utilizar los parámetros *espaciotablas_índice* y *espaciotablas_lob*. Si no se proporciona un valor, se emplea el espacio de tablas de datos de la tabla fuente. Este parámetro es sensible a mayúsculas y minúsculas y tiene un tipo de datos VARCHAR(128). Este parámetro puede tener valor NULL o la serie vacía.

espaciotablas_índice

Este parámetro de entrada especifica el nuevo espacio de tablas de índice para la tabla de destino. Si se proporciona un valor, es necesario utilizar los parámetros *espaciotablas_datos* y *espaciotablas_lob*. Si no se proporciona un valor, se emplea el espacio de tablas de índice de la tabla fuente. Este parámetro es sensible a mayúsculas y minúsculas y tiene un tipo de datos VARCHAR(128). Este parámetro puede tener valor NULL o la serie vacía.

espaciotablas_lob

Este parámetro de entrada especifica el nuevo espacio de tablas LOB para la tabla de destino. Si se proporciona un valor, es necesario utilizar los parámetros *espaciotablas_datos* y *espaciotablas_índice*. Si no se proporciona un valor, se emplea el espacio de tablas LOB de la tabla fuente. Este parámetro es sensible a mayúsculas y minúsculas y tiene un tipo de datos VARCHAR(128). Este parámetro puede tener valor NULL o la serie vacía.

cols_mdc

Este parámetro de entrada proporciona la especificación de columna multidimensional (MDC) para la tabla de destino. Los valores se especifican como una lista separada por comas de las columnas utilizadas para agrupar en clústeres los datos de la tabla de destino en múltiples dimensiones. Si se proporciona un valor NULL o "-", no se utiliza la cláusula ORGANIZE BY DIMENSIONS. Si se proporciona una serie vacía o un solo espacio en blanco, el procedimiento comprueba si existe una especificación de MDC en la tabla fuente y, si la encuentra, utiliza dicha especificación. Este parámetro tiene un tipo de datos VARCHAR(32672) y tiene el mismo formato que la cláusula ORGANIZE BY DIMENSIONS de la sentencia CREATE TABLE. Este parámetro puede tener valor NULL, serie vacía o un solo espacio en blanco.

Ejemplo: 'C1, C4, (C3,C1), C2'

cols_clavepart

Este parámetro de entrada proporciona la especificación de columnas de clave de particionamiento para la tabla de destino. Los valores se especifican como una lista separada por comas de las columnas de clave que especifican cómo se distribuyen los datos en múltiples particiones de base de datos. Si se proporciona un valor NULL o "-", no se utiliza la cláusula PARTITIONING KEY. Si se proporciona una serie vacía o un solo espacio en blanco, el procedimiento comprueba si existe una especificación de columnas de clave de particionamiento y, si la encuentra, utiliza dicha especificación. Este parámetro tiene un tipo de datos VARCHAR(32672) y tiene el mismo formato que la cláusula DISTRIBUTE BY HASH de la sentencia CREATE TABLE.

Ejemplo: 'C1, C3'

part_datos

Este parámetro de entrada proporciona la especificación de particionamiento de datos para la tabla de destino. Esta sentencia define cómo se dividen los datos de tabla entre varios objetos de almacenamiento (denominados particiones de datos), de acuerdo con los valores indicados en una o más de las columnas de tabla. Si se proporciona un valor NULL o "-", no se utiliza la cláusula PARTITION BY RANGE. Si se proporciona una serie vacía o un solo espacio en blanco, el procedimiento comprueba si existe un esquema de particionamiento de datos en la tabla de origen y, si lo encuentra, utiliza dicha información, incluido el nombre de la partición. Este parámetro tiene un tipo de datos VARCHAR(32672) y tiene el mismo formato que la cláusula PARTITION BY RANGE de la sentencia CREATE TABLE.

Ejemplo: '(C1) (STARTING FROM (1) EXCLUSIVE ENDING AT (1000) EVERY (100))'

defcol

Este parámetro especifica una definición de columna nueva para la tabla de destino, para que pueda cambiar los tipos de columna siempre y cuando sean compatibles; no obstante, los nombres de columna deben permanecer inalterados.

También permite agregar columnas nuevas y descartar columnas existentes. Al añadir una columna, debe definirse como anulable o disponer de un conjunto de valores por omisión. Además, solo se puede descartar una columna si hay un índice exclusivo o primario en la tabla y si la columna que se va a descartar no forma parte de dicho índice exclusivo o primario. Este parámetro tiene un tipo de datos VARCHAR(32672). Este parámetro puede tener valor NULL o la serie vacía.

Ejemplo: 'C1 INT, C2 INT DEFAULT 0'

nombretabla_destino

Este parámetro de entrada proporciona el nombre de una tabla existente que se utilizará como tabla de destino durante el traslado. Se pueden realizar los cambios siguientes en la tabla de destino que se está proporcionando:

- Se pueden modificar los datos, índices y espacios de tablas LOB
- Se puede añadir o cambiar la especificación de columna multidimensional (MDC)
- Se puede añadir o cambiar la especificación de columna de clave de particionamiento
- Se puede añadir o cambiar la especificación de particionamiento de datos
- Se puede añadir o eliminar la compresión de datos

- Se puede especificar una definición de columna nueva, aunque en este caso se aplican las mismas restricciones utilizadas al determinar el parámetro *defcol*.

Se aplican las siguientes restricciones a la tabla con nombre:

- La tabla debe existir en el mismo esquema que la tabla fuente
- La tabla debe estar vacía
- No se permiten tablas con tipo, tablas de consulta materializada (MQT), tablas de etapas, tablas remotas o tablas de clústeres

Si este parámetro está establecido en NULL o tiene una serie vacía, el procedimiento almacenado emplea la misma definición que la tabla fuente. Este parámetro es sensible a mayúsculas y minúsculas y tiene un tipo de datos VARCHAR(128).

opciones

Este conjunto de parámetros de entrada separados por comas define todas las opciones que utiliza el procedimiento almacenado.

- KEEP: esta opción conserva una copia de la tabla fuente original con un nombre distinto. Si el nombre de la tabla fuente es T1, la tabla se renombrará automáticamente tras el traslado a algo similar a T1AAAAVxo. Puede recuperar el nombre exacto de la tabla fuente en la tabla de protocolo devuelta, bajo la clave ORIGINAL. Puede establecer esta opción en cualquier momento antes de la fase SWAP y durante la misma.
- COPY_USE_LOAD: esta opción utiliza la API db2Load no recuperable para copiar los datos de la tabla fuente en la tabla de destino. Puede establecer esta opción en cualquier momento antes de la fase COPY y durante la misma.
- COPY_WITH_INDEXES: esta opción crea índices antes de copiar la tabla fuente. Sin embargo, el comportamiento por omisión consiste en crear los índices una vez copiada la tabla fuente. Las ventajas de esta opción son dos: la creación de índice tras la copia requiere una exploración de tabla completa por índice y la creación de índices es una transacción que requiere espacio de anotaciones cronológicas activo. Si el parámetro de configuración de base de datos LOGINDEXREBUILD está activado, es necesario disponer de una cantidad de espacio de anotaciones cronológicas considerable para construir los índices en un marco de tiempo corto. Una desventaja de esta opción es que se reduce el rendimiento de copia ya que es necesario conservar los índices en la tabla de destino. Además, los índices resultantes pueden contener claves pseudosuprimidas, y los índices no están tan bien equilibrados como si se hubieran creado tras la copia. Puede establecer la opción COPY_WITH_INDEXES en cualquier momento antes de la fase COPY y durante la misma.
- FORCE: si está establecida la opción force, la fase SWAP no comprueba si la tabla fuente ha modificado su definición de tabla. Además, si se utiliza LOAD y no se ha establecido FORCE, se visualiza un mensaje de error, ya que es obligatorio especificar FORCE cuando se emplea LOAD. Puede establecer esta opción en cualquier momento antes de la fase SWAP y durante la misma.
- NO_STATS: esta opción no inicia RUNSTATS ni ninguna otra copia de estadísticas en la tabla de destino. Si utiliza los parámetros de configuración de base de datos AUTO_RUNSTATS o AUTO_STMT_STATS, DB2 creará automáticamente estadísticas nuevas posteriormente. En lo que respecta a la

compatibilidad con versiones anteriores, `STATS_NO` también está aceptado. Puede establecer la opción `NO_STATS` en cualquier momento antes de la fase `SWAP` y durante la misma.

- `COPY_STATS`: esta opción copia las estadísticas de la tabla fuente en la tabla de destino antes de realizar el intercambio. Lo anterior puede provocar que se generen estadísticas físicas poco precisas, especialmente si ha cambiado el tamaño de página. Sin embargo, si se establece esta opción se ahorra tiempo de cálculo ya que no se llama a `RUNSTATS` para que calcule estadísticas nuevas. Además, el optimizador puede optar por utilizar los mismos planes de acceso, ya que las estadísticas son las mismas. En lo que respecta a la compatibilidad con versiones anteriores, `STATS_COPY` también está aceptado. Puede establecer la opción `STATS_COPY` en cualquier momento antes de la fase `SWAP` y durante la misma.
- `NO_AUTO_REVAL`: esta opción impide que se ejecute la revalidación automática en la tabla, y recrea en su lugar todos los activadores y las vistas. La opción `NO_AUTO_REVAL` solamente se puede establecer en la fase `INIT`.
- `REORG`: esta opción establece que se realice un `REORG` fuera de línea adicional en la tabla de destino antes del intercambio. Si utiliza esta opción para mejorar su diccionario de compresión, recuerde que utilizar el enfoque de muestreo por omisión es un método mucho más apropiado para crear un diccionario de compresión óptimo. No obstante, si necesita un diccionario de compresión XML óptimo, `REORG` es el único método. Puede establecer la opción `REORG` en cualquier momento antes de la fase `SWAP` y durante la misma.

La lista de opciones no es sensible a mayúsculas y minúsculas y tiene un tipo de datos `VARCHAR(128)`. El valor de la lista puede ser `NULL` o la serie vacía.

operación

Este parámetro de entrada especifica qué operación va a ejecutar el procedimiento almacenado. Existen dos formas de llamar al procedimiento almacenado: utilizando el mandato `MOVE` para ejecutar todas las operaciones simultáneamente, o utilizando los mandatos individuales para ejecutar el traslado de tabla paso a paso. La principal ventaja del segundo método es que puede controlar cuándo se producirá realmente la fase `SWAP`, y así determina cuándo quedará brevemente fuera de línea la tabla. De esta forma, podrá realizar el traslado durante un período de baja actividad del sistema. Si utiliza los mandatos individuales, debe llamarlos en el orden siguiente: `INIT`, `COPY`, `REPLAY`, `VERIFY` (opcional) y `SWAP`.

- `MOVE`: realiza el traslado completo de la tabla (operaciones `INIT`, `COPY`, `REPLAY` y `SWAP`) en un único paso.
- `INIT`: comprueba que se puede realizar un traslado de tabla, e inicializa todos los datos necesarios durante el proceso de traslado de tabla (la tabla de destino, la tabla de etapas y los activadores de la tabla fuente).
- `COPY`: copia el contenido de la tabla fuente en la tabla de destino. Se capturan y almacenan en la tabla de etapas todas las actualizaciones, supresiones o inserciones que se producen en la tabla fuente durante este tiempo. Se crean índices nuevos al final de la fase `COPY`, a menos que esté seleccionada la opción `COPY_WITH_INDEXES`. Además, en caso necesario, se crean índices secundarios en las tablas fuente y de destino para mejorar el rendimiento durante la fase `REPLAY`. `COPY` puede utilizarse únicamente cuando se ha completado la fase `INIT`.
- `REPLAY`: copia en la tabla de destino cualquier fila que haya cambiado en la tabla fuente desde que comenzó la fase `COPY`. `REPLAY` puede utilizarse únicamente cuando se ha completado la fase `COPY`.

- **VERIFY:** compruebe opcionalmente si los contenidos de tabla de las tablas fuente y de destino son idénticos. Para este proceso se tiene que obtener un bloqueo compartido en las tablas fuente y de destino, reproducir cualquier cambio producido en la tabla fuente y, a continuación, realizar una comparación. Si la tabla tiene un índice exclusivo, este mandato compara todos los valores entre las columnas que se encuentran en ambas tablas. En caso contrario, este mandato compara todos los valores entre las columnas que se encuentran en ambas tablas, excepto las columnas LONG, LOB o XML. Ésta es una operación costosa y debe decidir con precaución si le resulta útil para su traslado. VERIFY puede utilizarse únicamente cuando se ha completado la fase COPY o la fase REPLAY.
- **SWAP:** ejecuta la fase REPLAY hasta que el número de cambios aplicados durante la última exploración de la tabla de etapas es menor que el valor REPLAY_THRESHOLD almacenado en la tabla de protocolo. A continuación, la tabla fuente queda brevemente fuera de línea para completar el REPLAY final. Luego, este mandato intercambia la tabla fuente con la tabla de destino y vuelve a poner la tabla en línea. SWAP puede utilizarse únicamente cuando se ha completado la fase COPY e, idealmente, una vez llamada la fase REPLAY.
- **CLEANUP:** descarta la tabla de etapas, todos los activadores o índices no exclusivos creados en la tabla fuente por el procedimiento almacenado, y la tabla fuente si no se ha determinado la opción KEEP. Se puede llamar a CLEANUP si ha fallado el mandato durante la fase SWAP.
- **CANCEL:** cancela un traslado de tabla de varios pasos entre fases, o cancela una operación de traslado de tabla fallida. Para ejecutar este mandato es necesario que el estado de la operación no sea COMPLETED o CLEANUP. CANCEL borra todos los datos intermedios (los índices, la tabla de etapas, la tabla de destino y los activadores de la tabla fuente).

Este parámetro no es sensible a mayúsculas y minúsculas y tiene un tipo de datos VARCHAR(128).

Autorización

No es necesario disponer de autorización explícita para invocar el procedimiento almacenado ADMIN_MOVE_TABLE. No obstante, para ejecutar satisfactoriamente el procedimiento almacenado, debe tener las autorizaciones de creación de objetos adecuadas, incluidas la autorización para seleccionar desde las tablas de origen y la autorización para insertar en la operación de copia.

Ejemplos

En este ejemplo se llama al procedimiento almacenado utilizando el primer método, donde la tabla de destino está definida dentro del procedimiento, con el objetivo de mover una tabla denominada T1 que está ubicada en el esquema "SVALENTI".

```
CALL SYSPROC.ADMIN_MOVE_TABLE(
'SVALENTI',
'T1',
'ACCOUNTING',
'ACCOUNT_IDX',
'ACCOUNT_LONG',
'',
'';
```

```

'',
'CUSTOMER VARCHAR(80), REGION CHAR(5), YEAR INTEGER, CONTENTS CLOB',
'',
'MOVE')

```

El ejemplo siguiente muestra los datos de salida de la consulta.

Conjunto de resultados 1

```

-----
KEY                                VALUE
-----
AUTHID                             SVALENTI
CLEANUP_END                         2009-02-13-11.34.07.609575
CLEANUP_START                       2009-02-13-11.34.07.369331
COPY_END                             2009-02-13-11.34.05.148018
COPY_OPTS                           BY_KEY,OVER_INDEX
COPY_START                          2009-02-13-11.34.04.841292
COPY_TOTAL_ROWS                     100
INDEXNAME                           T1_INDEX
INDEXSCHEMA                         SVALENTI
INDEX_CREATION_TOTAL_TIME           0
INIT_END                            2009-02-13-11.34.04.552875
INIT_START                          2009-02-13-11.34.03.013563
PAR_COLDEF                          CUSTOMER VARCHAR(80), REGION CHAR(5),
                                     YEAR INTEGER, CONTENTS CLOB
REPLAY_END                          2009-02-13-11.34.06.198369
REPLAY_START                        2009-02-13-11.34.05.164582
REPLAY_TOTAL_ROWS                   100
REPLAY_TOTAL_TIME                    5
STATUS                               COMPLETE
SWAP_END                             2009-02-12-11.34.07.214447
SWAP_RETRIES                         0
SWAP_START                          2009-02-13-11.34.06.244506
VERSION                             09.07.0000

```

22 registro(s) seleccionado(s).
Estado de devolución = 0

En este ejemplo se llama al procedimiento almacenado utilizando el segundo método, donde la tabla de destino se crea fuera del procedimiento y a continuación recibe su nombre a partir del parámetro *nombretabla_destino*, con el objetivo de mover la misma tabla que en el ejemplo anterior.

El primer paso es crear la tabla manualmente:

```

CREATE TABLE SVALENTI.T1_TARGET (
  CUSTOMER VARCHAR(80),
  REGION CHAR(5),
  YEAR INTEGER,
  CONTENTS CLOB)
IN ACCOUNTING
INDEX IN ACCOUNT_IDX
LONG IN ACCOUNT_LONG'

```

A continuación, se llama al procedimiento almacenado y se proporciona el nombre de la tabla de destino:

```

CALL SYSPROC.ADMIN_MOVE_TABLE(
'SVALENTI',
'T1',
'T1_TARGET',
'',
'MOVE')

```

El ejemplo siguiente muestra los datos de salida de la consulta.

Conjunto de resultados 1

```
-----  
KEY                               VALUE  
-----  
AUTHID                            SVALENTI  
CLEANUP_END                       2009-02-13-11.37.49.283090  
CLEANUP_START                     2009-02-13-11.37.49.125786  
COPY_END                          2009-02-13-11.37.47.806060  
COPY_OPTS                         BY_KEY,OVER_INDEX  
COPY_START                        2009-02-13-11.37.47.446616  
COPY_TOTAL_ROWS                   0  
INDEXNAME                         T1_INDEX  
INDEXSCHEMA                       SVALENTI  
INDEX_CREATION_TOTAL_TIME         1  
INIT_END                          2009-02-13-11.37.47.287703  
INIT_START                        2009-02-13-11.37.46.052952  
PAR_COLDEF                        utilizando una tabla de destino proporcionada  
                                  para que COLDEF pueda ser diferente  
REPLAY_END                        2009-02-13-11.37.48.785503  
REPLAY_START                      2009-02-13-11.37.47.822109  
REPLAY_TOTAL_ROWS                 0  
REPLAY_TOTAL_TIME                 0  
STATUS                            COMPLETE  
SWAP_END                          2009-02-13-11.37.48.977745  
SWAP_RETRIES                      0  
SWAP_START                        2009-02-13-11.37.48.825228  
VERSION                           09.07.0000  
22 registro(s) seleccionado(s).  
Estado de devolución = 0
```

Notas de uso

Sugerencias para mejorar los resultados al utilizar este procedimiento:

- Evite realizar varios traslados al mismo espacio de tablas simultáneamente. Así se impide la fragmentación en el espacio de tablas de destino.
- Ejecute este procedimiento cuando haya poca actividad en la tabla. Procure no realizar cargas o supresiones de datos masivas para que el acceso de lectura paralelo no constituya un problema.
- Utilice una operación de traslado de varios pasos. Se puede llamar a las fases INIT y COPY en cualquier momento. Ejecute la fase REPLAY varias veces en orden para que el tamaño de la tabla de etapas continúe siendo pequeño, y a continuación emita SWAP durante un período de baja actividad en la tabla.
- Compruebe si los métodos fuera de línea son una opción mejor para su traslado de tabla, especialmente si se trata de tablas sin índices exclusivos y de tablas sin índice.

Restricciones

Las siguientes restricciones se aplican al procedimiento almacenado ADMIN_MOVE_TABLE:

- Solo se soportan las tablas simples como tablas fuente. No se permiten tablas de consulta materializada, tablas con tipo, tablas de clústeres, tablas de sistema, vistas, apodos o alias.
- No se puede mover una tabla si está activo un supervisor de sucesos en la tabla.
- No se soportan las claves foráneas (restricciones de referencia), independientemente de si son padre o hija. Para mover una tabla con claves

foráneas, puede capturar las claves foráneas utilizando el mandato db2look y, a continuación, descartar las claves foráneas, realizar la operación de traslado y volver a crear las claves.

- Las tablas sin índice exclusivo están sujetas a una fase de reproducción compleja y potencialmente costosa.
- Es necesario un índice exclusivo si la tabla contiene columnas LOB, XML o LONG.
- Una columna generada no puede formar parte de la especificación de MDC.
- No existe soporte para los índices de búsqueda de texto.
- Tenga en cuenta los requisitos de espacio de disco grande, ya que el procedimiento crea dos copias de la tabla y los índices, más una tabla de etapas y el espacio de anotaciones cronológicas.
- El rendimiento de la copia puede suponer algún problema, ya que la mayoría de los datos se mueven a la tabla nueva mediante "insertar desde selección".
- La operación VERIFY para tablas sin índice exclusivo no funciona en las tablas con LOB.
- La variable de registro DB2_SKIPDELETED de DB2 no debe estar establecida en ON.
- El espacio de tablas SYSTOOLSPACE debe haberse creado y estar accesible a 'PUBLIC'.
- Pueden producirse tiempos de espera excedidos de bloqueo durante la fase COPY debido a las transacciones de ejecución prolongada que se realizan en la tabla fuente.
- Pueden producirse puntos muertos durante la fase SWAP.
- Pueden producirse puntos muertos en una tabla fuente con índices no exclusivos y varios procesos de actualización.
- Si está habilitado el soporte de VARCHAR2, la base de datos considera que los valores de serie vacía y NULL son equivalentes, pero el espacio en blanco único es un valor diferenciado. Si está habilitado el soporte de VARCHAR2, los parámetros *cols_mdc*, *cols_clavepart* y *part_datos* pueden utilizar un solo espacio en blanco como valor diferenciado con respecto a la serie vacía y NULL.

Información devuelta

Tabla 212. Información devuelta por el procedimiento almacenado ADMIN_MOVE_TABLE

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| TABSCHEMA | VARCHAR(128) | Esquema de la tabla que se va a mover. Serie vacía para los valores por omisión de todo el sistema. |
| TABNAME | VARCHAR(128) | Nombre de tabla de la tabla que se va a mover. Serie vacía para los valores por omisión de todo el sistema. |
| KEY | VARCHAR(32) | Nombre del atributo. |
| VALUE | CLOB(10M) | Valor del atributo. |

Los pares de clave y valor que se devuelven en el conjunto de resultados se incluyen en la Tabla 213 en la página 937. Para modificar las claves configurables por el usuario en el conjunto de resultados, utilice el procedimiento almacenado ADMIN_MOVE_TABLE_UTIL.

Tabla 213. Pares de clave y valor devueltos por el procedimiento almacenado ADMIN_MOVE_TABLE

| Clave | Valor de retorno | Configurable por el usuario |
|------------------------|---|-----------------------------|
| VERSION | Muestra la versión del procedimiento almacenado. | No |
| AUTHID | Muestra el ID de autorización del usuario que llamó al procedimiento almacenado. | No |
| LOCK | Muestra la hora de inicio de LOCK si está activo otro procedimiento almacenado de traslado de tabla en línea; en caso contrario, está vacío. | No |
| STATUS | Muestra el estado actual del traslado de tabla en línea: <ul style="list-style-type: none"> • INIT: INIT está en curso. • COPY: COPY está en curso o se puede realizar. • REPLAY: REPLAY está en curso o se pueden realizar REPLAY y SWAP. • CLEANUP: MOVE está completado, pero no ha terminado la limpieza o se puede realizar CLEANUP. • COMPLETE: MOVE y CLEANUP están completados. • COMPLETE_WITH_WARNINGS: MOVE y CLEANUP están completados, pero se han producido avisos (listados en la clave WARNINGS). | No |
| STAGING | Muestra el nombre de la tabla de etapas. | No |
| TARGET | Muestra el nombre de la tabla de destino. | No |
| ORIGINAL | Muestra el nombre de la tabla original tras el intercambio. | No |
| INDEXSCHEMA | Muestra el esquema del índice o la serie vacía si la tabla no dispone de un índice. | No |
| INDEXNAME | Muestra el nombre del índice o la serie vacía si la tabla no dispone de un índice. | No |
| COMMIT_AFTER_N_ROWS | Durante la fase COPY, se ejecuta una confirmación tras la copia del número de filas especificado. 0 significa que no hay confirmaciones durante COPY. El valor por omisión es 10000. | Sí |
| DEEPCOMPRESSION_SAMPLE | Si la tabla fuente tiene habilitada la compresión, este campo especifica la cantidad (en KB) de datos para la que se producen muestras al crear un diccionario para compresión. 0 significa que no se toman muestras. El valor por omisión es 20 MB (20.480 KB). | Sí |
| COPY_ARRAY_SIZE | Especifica el tamaño ARRAY para COPY_ARRAY_INSERT. Un valor de 0 o menor significa que no se usa COPY_ARRAY_INSERT. El valor por omisión es 100. | Sí |
| COPY_OPTS | Opciones de copia utilizadas durante la fase COPY. | No |

Tabla 213. Pares de clave y valor devueltos por el procedimiento almacenado ADMIN_MOVE_TABLE (continuación)

| Clave | Valor de retorno | Configurable por el usuario |
|---------------------------|---|-----------------------------|
| COPY_INDEXSCHEMA | Esquema del índice utilizado para agrupar en clústeres los datos de la tabla de destino durante la fase COPY. Este valor debe especificarse antes de la fase COPY. El esquema por omisión es el nombre de esquema de un índice de clúster en la tabla fuente, si existe; en caso contrario, nombre de esquema del índice exclusivo o primario en la tabla fuente. | Sí |
| COPY_INDEXNAME | Nombre del índice utilizado para agrupar en clústeres los datos de la tabla de destino durante la fase COPY. Este valor debe especificarse antes de la fase COPY. El nombre por omisión es el nombre de un índice de clúster en la tabla fuente, si existe; en caso contrario, nombre del índice exclusivo o primario en la tabla fuente. | Sí |
| INDEX_CREATION_TOTAL_TIME | Muestra el tiempo total necesario para crear índices secundarios. | No |
| INIT_START | Muestra la hora de inicio de la fase INIT. | No |
| INIT_END | Muestra la hora de finalización de la fase INIT. | No |
| COPY_START | Muestra la hora de inicio de la fase COPY. | No |
| COPY_END | Muestra la hora de finalización de la fase COPY. | No |
| COPY_TOTAL_ROWS | Muestra el número total de filas copiadas durante la fase COPY. | No |
| REPLAY_START | Muestra la hora de inicio de la fase REPLAY. | No |
| REPLAY_END | Muestra la hora de finalización de la fase REPLAY. | No |
| REPLAY_TOTAL_ROWS | Muestra el número acumulado de filas reproducidas. | No |
| REPLAY_TOTAL_TIME | Muestra el tiempo acumulado, en segundos, utilizado para la reproducción de filas. | No |
| REPLAY_MAX_ERR_RETRIES | Especifica el número máximo de intentos para los errores (tiempos de espera excedidos de bloqueo o puntos muertos) que se pueden producir durante la fase REPLAY. El valor por omisión es 100. | Sí |
| REPLAY_THRESHOLD | Para una única iteración de la fase REPLAY, si el número de filas aplicadas a la tabla de etapas es menor que este valor, se detiene REPLAY, aunque se hayan realizado entradas nuevas en este tiempo. El valor por omisión es 100. | Sí |
| REORG_USE_TEMPSPACE | Si se llama a la opción REORG, también se puede especificar un espacio de tablas temporal para la cláusula USE del mandato REORG. Si no se especifica un valor aquí, el mandato REORG emplea el mismo espacio de tablas que la tabla que se está reorganizando. | Sí |
| VERIFY_START | Muestra la hora de inicio de la verificación. | No |
| VERIFY_END | Muestra la hora de finalización de la verificación. | No |
| SWAP_START | Muestra la hora de inicio de la fase SWAP. | No |
| SWAP_END | Muestra la hora de finalización de la fase SWAP. | No |

Tabla 213. Pares de clave y valor devueltos por el procedimiento almacenado ADMIN_MOVE_TABLE (continuación)

| Clave | Valor de retorno | Configurable por el usuario |
|------------------|---|-----------------------------|
| SWAP_MAX_RETRIES | Especifica el número máximo de intentos permitidos durante la fase SWAP (si se producen tiempos de espera excedidos de bloqueo o puntos muertos). El valor por omisión es 10. | Sí |
| SWAP_RETRIES | Muestra el número de reintentos realizados durante la fase SWAP. | No |
| CLEANUP_START | Muestra la hora de inicio de la fase CLEANUP. | No |
| CLEANUP_END | Muestra la hora de finalización de la fase CLEANUP. | No |
| WARNINGS | Muestra los avisos que se proporcionan al usuario. Estos avisos incluyen: <ul style="list-style-type: none"> • Revalidación de todos los objetos fallidos • No se ha podido conceder el control a un usuario, grupo o rol • No se ha creado un índice porque ya no existe una columna a la que hace referencia | No |

Procedimiento ADMIN_MOVE_TABLE_UTIL - Modificar el procedimiento de trasladar tablas en línea

El procedimiento ADMIN_MOVE_TABLE_UTIL colabora con el procedimiento almacenado SYSPROC.ADMIN_MOVE_TABLE para mover datos de tablas activas. Este procedimiento almacenado ofrece un mecanismo para modificar los valores definibles por el usuario en la tabla de protocolo ADMIN_MOVE_TABLE, que crea y utiliza el procedimiento ADMIN_MOVE_TABLE.

Este procedimiento modificará únicamente un valor en la tabla de protocolo ADMIN_MOVE_TABLE si ya se está realizando un traslado de tabla para la tabla a la que se hace referencia en los parámetros TABSCHEMA y TABNAME, y si el ID de autorización del emisor de la llamada al procedimiento es el mismo que el del usuario que está ejecutando el traslado de tabla.

Sintaxis

```
►►—ADMIN_MOVE_TABLE_UTIL—(—esquematabla—,—nombretabla—,—acción—,—clave—,—valor—)————►►
```

El esquema para este procedimiento almacenado es SYSPROC.

Parámetros del procedimiento

esquematabla

Este parámetro de entrada especifica el nombre del esquema que contiene la tabla que se va a mover. Este nombre es sensible a las mayúsculas y minúsculas. Su tipo de datos es VARCHAR(128).

nombretabla

Este parámetro de entrada especifica el nombre de la tabla que se va a mover. Este parámetro es sensible a mayúsculas y minúsculas y tiene un tipo de datos VARCHAR(128).

acción

Este parámetro de entrada especifica la acción que realizará el procedimiento.

Los valores válidos son:

- UPSERT: si TABSCHEMA.TABNAME.KEY existe en la tabla de protocolo ADMIN_MOVE_TABLE, actualiza el VALUE correspondiente con el nuevo parámetro de *valor*. En caso contrario, inserta el par KEY y VALUE en la tabla de protocolo ADMIN_MOVE_TABLE.
- DELETE: si existe el TABSCHEMA.TABNAME.KEY especificado en la tabla de protocolo ADMIN_MOVE_TABLE, elimina el par KEY y VALUE especificado de la tabla de protocolo ADMIN_MOVE_TABLE.

El tipo de datos de este parámetro es VARCHAR(128).

clave

Este parámetro de entrada especifica la clave que se someterá a UPSERT o que se borrará de la tabla de protocolo ADMIN_MOVE_TABLE.

Los valores válidos son:

- COMMIT_AFTER_N_ROWS: durante la fase COPY, se ejecuta una confirmación tras la copia del número de filas especificado. Un valor de 0 significa que no se ejecutan confirmaciones durante COPY.
- DEEPCOMPRESSION_SAMPLE: si la tabla fuente tiene habilitada la compresión, este campo especifica la cantidad (en KB) de datos para la que se producen muestras al crear un diccionario para compresión. Un valor de 0 significa que no se toman muestras.
- COPY_ARRAY_SIZE: especifica el tamaño de ARRAY para COPY_ARRAY_INSERT; un valor de 0 o menor significa que no se usa COPY_ARRAY_INSERT.
- COPY_INDEXSCHEMA: esquema del índice utilizado para agrupar en clústeres los datos de la tabla de destino durante la fase COPY.
- COPY_INDEXNAME: nombre del índice utilizado para agrupar en clústeres los datos de la tabla de destino durante la fase COPY.
- REPLAY_MAX_ERR_RETRIES: especifica el número máximo de intentos para los errores (tiempos de espera excedidos de bloqueo o puntos muertos) que se pueden producir durante la fase REPLAY.
- REPLAY_THRESHOLD: para una única iteración de la fase REPLAY, si el número de filas aplicadas a la tabla de etapas es menor que este valor, se detiene REPLAY, aunque se hayan realizado entradas nuevas en este tiempo.
- REORG_USE_TEMPSPACE: si se llama a la opción REORG en el traslado de tablas, también se puede especificar un espacio de tablas temporal para la cláusula USE del mandato REORG. Si no se especifica un valor aquí, el mandato REORG emplea el mismo espacio de tablas que la tabla que se está reorganizando.
- SWAP_MAX_RETRIES: especifica el número máximo de intentos permitidos durante la fase SWAP (si se producen tiempos de espera excedidos de bloqueo o puntos muertos).

Este parámetro tiene un tipo de datos VARCHAR(128).

valor

Este parámetro de entrada especifica el valor que se someterá a UPSERT en la tabla de protocolo ADMIN_MOVE_TABLE. Este parámetro tiene un tipo de datos CLOB(10M). El parámetro puede tener valor NULL o la serie vacía.

Autorización

No es necesario disponer de autorización explícita para invocar el procedimiento almacenado. No obstante, el ID de autorización utilizado debe ser el mismo que el empleado para llamar al procedimiento almacenado ADMIN_MOVE_TABLE.

Ejemplos

Este ejemplo trata una llamada básica al procedimiento almacenado con el objetivo de actualizar el valor de compresión y eliminar la información de índice específica utilizada para copiar la tabla de destino.

En primer lugar, se llama al procedimiento ADMIN_MOVE_TABLE para que comience el proceso de traslado de tablas antes de llamar al procedimiento ADMIN_MOVE_TABLE_UTIL para actualizar o eliminar un valor de la tabla de protocolo ADMIN_MOVE_TABLE:

```
CALL SYSPROC.ADMIN_MOVE_TABLE('SVALENTI','T1','','','','','','','','','INIT')
```

A continuación, se actualiza el valor DEEP_COMPRESSION_SAMPLE a 30.720 KB:

```
CALL SYSPROC.ADMIN_MOVE_TABLE_UTIL('SVALENTI','T1','UPSERT',  
    'DEEPCOMPRESSIÖN_SÄMPLE','30720')
```

Ahora, se eliminan los valores COPY_INDEXSCHEMA y COPY_INDEXNAME:

```
CALL SYSPROC.ADMIN_MOVE_TABLE_UTIL('SVALENTI','T1','DELETE','COPY_INDEXSCHEMA','')  
CALL SYSPROC.ADMIN_MOVE_TABLE_UTIL('SVALENTI','T1','DELETE','COPY_INDEXNAME','')
```

Tras realizar estos cambios, se continúa con el procedimiento ADMIN_MOVE_TABLE utilizando los nuevos valores de la metatabla:

```
CALL SYSPROC.ADMIN_MOVE_TABLE('SVALENTI','T1','','','','','','','','','COPY')  
CALL SYSPROC.ADMIN_MOVE_TABLE('SVALENTI','T1','','','','','','','','','REPLAY')  
CALL SYSPROC.ADMIN_MOVE_TABLE('SVALENTI','T1','','','','','','','','','SWAP')
```

Notas de uso

Para más información relativa a los valores KEY modificables de la tabla de protocolo ADMIN_MOVE_TABLE, consulte la sección de notas de uso del procedimiento ADMIN_MOVE_TABLE.

ALTOBJ

El procedimiento ALTOBJ analiza una sentencia CREATE TABLE de entrada que sirve como lenguaje de definición de datos (DDL) de destino para una tabla existente que se debe modificar. El procedimiento hace copia de seguridad de los datos de la tabla que se está modificando y, a continuación, descarta la tabla original y crea una nueva versión mediante la sentencia DDL; el último paso vuelve a cargar los datos almacenados en la tabla nueva.

Este procedimiento da soporte a las siguientes operaciones de modificación de tabla y mantiene las dependencias recuperables:

- Renombrar una columna
- Aumentar o disminuir el tamaño de una columna
- Modificación de un tipo de columna y transformación de datos existentes mediante las funciones escalares de DB2
- Cambiar la precisión o la escala de los valores decimales

- Cambiar el valor por omisión de una columna
- Cambiar el atributo de capacidad de nulos de una columna por el de anulable
- Descartar una columna

Sintaxis

►►—ALTOBJ—(—*modalidad-ejec*—,—*sentencia-sql*—,—*id-modif*—,—*msj*—)————►◄

El esquema es SYSPROC.

Parámetros del procedimiento

modalidad-ejec

Argumento de entrada de tipo VARCHAR(30) que especifica una de las siguientes modalidades de ejecución:

'GENERATE'

Especifica que se deben generar todos los scripts necesarios para las modalidades VALIDATE, APPLY y UNDO.

'VALIDATE'

Especifica que se debe validar la sintaxis de la sentencia. Esta opción también genera un script que gestiona el proceso de objetos relacionados y relaciones de la tabla que se debe modificar.

'APPLY_CONTINUE_ON_ERROR' o 'APPLY_STOP_ON_ERROR'

Especifica que se debe generar un script que gestione el proceso de objetos relacionados y relaciones de la tabla que se debe modificar. Se exportarán, transformarán y utilizarán datos de la tabla original para llenar la nueva tabla.

'UNDO'

Especifica que se desharán los cambios hechos mediante la operación de modificación de tabla, en el caso de que una operación de retroacción no pueda recuperar los errores que se hayan podido producir. Esta modalidad sólo es posible si la tabla original y los scripts generados no se han suprimido.

'FINISH'

Especifica que se debe descartar la tabla original renombrada.

sentencia-sql

Argumento de entrada de tipo VARCHAR(2048) que especifica una sentencia CREATE TABLE que se utilizará como plantilla para modificar una tabla existente. Cuando *exec-mode* es 'GENERATE', *sql-stmt* no debe ser el valor nulo. Al contrario, la *sentencia-sql* puede ser el valor nulo, pero sólo si el *id-modif* no es -1.

id-modif

Argumento de entrada y salida de tipo INTEGER que identifica todas las sentencias generadas por esta llamada. Si se especifica -1, se generará y devolverá al llamante un nuevo identificador. Cualquier sentencia existente identificada por el entero especificado se sobregrabará.

msj

Argumento de salida de tipo VARCHAR(2048) que contiene una consulta de SQL que se puede ejecutar para visualizar todas las sentencias de SQL generadas para el proceso de modificación de tabla o utilizadas por dicho proceso bajo la modalidad de ejecución especificada.

Autorización

Privilegio EXECUTE para el procedimiento ALTOBJ.

También se necesitan DBADM con autorización LOAD y SETSESSIONUSER.

Ejemplos

Ejemplo 1: Ejecutar el procedimiento ALTOBJ para modificar la columna CL2 de la tabla T1 cambiando el tipo INTEGER por el tipo BIGINT. El lenguaje de definición de datos original para la tabla T1 es:

```
CREATE TABLE T1 (CL1 VARCHAR(5), CL2 INTEGER)
```

La llamada al procedimiento ALTOBJ para modificar el tipo de datos de la columna es:

```
CALL SYSPROC.ALTOBJ('APPLY_CONTINUE_ON_ERROR',  
    'CREATE TABLE T1 (CL1 VARCHAR(5), CL2 BIGINT)', -1, ?)
```

Nota: Si ve el siguiente error, intente aumentar el valor del parámetro APPLHEAPSZ:

SQL0443N La rutina "SYSPROC.ALTOBJ" (nombre específico "ALTOBJ") ha devuelto un error SQLSTATE con el texto de diagnóstico "SQL0954 ".
SQLSTATE=38553

Ejemplo 2: Ejecutar el procedimiento ALTOBJ en la modalidad VALIDATE con la entrada del *id-modif*.

```
CALL SYSPROC.ALTOBJ('VALIDATE', CAST (NULL AS VARCHAR(2048)), 123, ?)
```

Notas de uso

Aunque el procedimiento descarta y vuelve a crear la tabla, el usuario que ha creado la tabla original permanecerá como definidor de la tabla. Sin embargo, un control mostrará que el usuario que está ejecutando el procedimiento ha descartado y vuelto a crear la tabla.

Este procedimiento no da soporte a las operaciones de modificación de tablas siguientes:

- No se da soporte a la modificación de las tablas de consultas materializadas (MQT). Se da soporte a la modificación de una tabla que contenga una MQT.
- No se da soporte a la modificación de tablas escritas.
- No se da soporte a la modificación de una tabla remota utilizando un apodo.
- No se puede reordenar la secuencia de columnas.
- No se da soporte a la adición y eliminación, ni al cambio de nombre y eliminación de columnas en una llamada al procedimiento, pero sí se da soporte a la adición y cambio de nombre de columnas. Ello se debe a que el único modo de indicar cómo se debe modificar la tabla es mediante el uso del DDL de destino, en lugar de la información sobre la correspondencia de columnas. Las normas siguientes van seguidas del procedimiento ALTOBJ cuando se transforman los datos de la tabla existente a la tabla modificada:
 1. Si el número de columnas en la tabla existente es el mismo que en la tabla modificada, se supone que no se están añadiendo ni eliminando columnas. En este caso sólo se pueden renombrar las columnas, y se comparan mediante el índice de columnas.

2. Si el número de columnas de la tabla existente es inferior al número de columnas de la tabla modificada, se supone que se están añadiendo columnas. Se pueden renombrar las columnas, y las nuevas columnas se añaden al final. Las columnas existentes se comparan mediante el índice.
 3. Si el número de columnas de la tabla existente es mayor que el número de columnas de la tabla modificada, se supone que se están eliminando columnas. Las columnas no se pueden renombrar ni comparar por nombre. La columna que se está eliminando puede ser cualquiera de la tabla.
- No se da soporte a los UDT de tipo estructurado ni de tipo de referencia.
 - Las MQT definidas en una tabla base que está modificada no se llenan durante el proceso de modificación de la tabla.

Si se modifica una tabla mediante el procedimiento ALTOBJ y la tabla tiene una MQT definida, se creará la MQT pero no se llenará con datos.

Si se modifica una tabla mediante el procedimiento ALTOBJ y la tabla tiene una MQT definida, se pierden las columnas que no forman parte de la selección resultado de la tabla que se está modificando, porque el contenido de la MQT se vuelve a crear desde la nueva tabla base.

Es posible que la definición de los objetos cambie entre las llamadas al procedimiento ALTOBJ porque no hay bloqueos de objetos que permanezcan tras varias sesiones.

Los perfiles de tablas (como el perfil runstats) que están asociados con la tabla se pierden tras este amplio proceso de modificación.

SYSTOOLSPACE se utiliza para las tablas de funcionamiento de la rutina para almacenar metadatos; es decir, datos utilizados para describir objetos de base de datos y su funcionamiento.

APPLICATION_ID

La función APPLICATION_ID devuelve el ID de aplicación de la conexión actual. El tipo de datos del resultado es VARCHAR(128).

El valor que devuelve la función es exclusivo dentro de un intervalo de 100 años y válido únicamente mientras dure la conexión establecida antes de llamar a la función.

Sintaxis

►► APPLICATION_ID(—)(—)◀◀

El esquema es SYSFUN.

Ejemplo

```
SELECT APPLICATION_ID() AS APPL_ID FROM SYSIBM.SYSDUMMY1
```

Función de tabla COMPILATION_ENV – Recuperar elementos del entorno de compilación

La función de tabla COMPILATION_ENV devuelve los elementos de un entorno de compilación.

Sintaxis

►—COMPILATION_ENV—(—ent-compilación—)—————►

El esquema es SYSPROC.

Parámetro de la función de tabla

ent-compilación

Parámetro de entrada de tipo BLOB(2 M) que contiene un entorno de compilación proporcionado por un supervisor de sucesos de puntos muertos.

La función devuelve una tabla de dos columnas (consulte la Tabla 214 en la página 946): NAME VARCHAR(256) y VALUE VARCHAR(1024). Los posibles valores de los nombres de los elementos del entorno de compilación se describen en la Tabla 215 en la página 946.

El origen de los valores de elementos depende principalmente de si la sentencia de SQL se emite dinámicamente o se vincula como parte de un paquete.

El número y los tipos de entradas de un entorno de compilación pueden cambiar a lo largo del tiempo a medida que se añaden funciones al gestor de bases de datos de DB2. Si el entorno de compilación es de un nivel de gestor de bases de datos de DB2 diferente del nivel donde se ejecuta esta función, sólo se devuelven los elementos reconocidos por el nivel de la función. Las descripciones de los elementos también pueden variar de un release a otro.

Ejemplos

Ejemplo 1: En este ejemplo se solicitan todos los elementos de un entorno de compilación determinado que previamente fue capturado por un supervisor de sucesos de puntos muertos. Un supervisor de sucesos de punto muerto que se ha creado especificando la opción WITH DETAILS HISTORY capturará el entorno de compilación de las sentencias de SQL dinámico. Este entorno capturado es lo que se utiliza como datos de entrada para la función de tabla.

```
SELECT NAME, VALUE
FROM TABLE(SYSPROC.COMPILATION_ENV(:hv1)) AS t
```

Ejemplo 2: En este ejemplo se solicita un elemento determinado (el esquema por omisión) de un entorno de compilación.

```
SELECT NAME, VALUE
FROM TABLE(SYSPROC.COMPILATION_ENV(:hv1)) AS t
WHERE NAME = 'SCHEMA'
```

Información devuelta

Tabla 214. Información devuelta por la función de tabla `COMPILATION_ENV`

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| NAME | VARCHAR(256) | Elemento del entorno de compilación. Consulte la Tabla 215 para obtener más información. |
| VALUE | VARCHAR(1024) | Valor del elemento. |

Tabla 215. Elementos de un entorno de compilación devueltos por la función de tabla `COMPILATION_ENV`

| Nombre de elemento | Descripción |
|--------------------|--|
| ISOLATION | Es el nivel de aislamiento que se pasa al compilador de SQL. El valor se obtiene a partir del registro especial <code>CURRENT ISOLATION</code> o de la opción de vinculación <code>ISOLATION</code> del paquete actual. |
| QUERY_OPTIMIZATION | Es el nivel de optimización de consultas que se pasa al compilador de SQL. El valor se obtiene a partir del registro especial <code>CURRENT QUERY OPTIMIZATION</code> o de la opción de vinculación <code>QUERYOPT</code> del paquete actual. |
| MIN_DEC_DIV_3 | Es la escala de cálculo decimal solicitada que se pasa al compilador de SQL. El valor se obtiene a partir del parámetro de configuración de base de datos <code>min_dec_div_3</code> . |
| DEGREE | Es el nivel solicitado de intra-paralelismo que se pasa al compilador de SQL. El valor se obtiene a partir del registro especial <code>CURRENT DEGREE</code> o de la opción de vinculación <code>DEGREE</code> del paquete actual. |
| SQLRULES | Son los patrones de comportamiento solicitados para las sentencias de SQL que se pasa al compilador de SQL. El valor se obtiene a partir del valor de la opción de vinculación <code>LANGLVL</code> del paquete actual. Los valores posibles son <code>'DB2'</code> o <code>'SQL92'</code> . |
| REFRESH_AGE | Es el valor de latencia de datos permitida que se pasa al compilador de SQL. El valor se obtiene a partir del registro especial <code>CURRENT REFRESH AGE</code> o de la opción de vinculación <code>REFRESHAGE</code> del paquete actual. |
| SCHEMA | Es el esquema por omisión que se pasa al compilador de SQL. El valor se obtiene a partir del registro especial <code>CURRENT SCHEMA</code> o de la opción de vinculación <code>QUALIFIER</code> del paquete actual. |
| PATH | Es la vía de acceso de función que se pasa al compilador de SQL. El valor se obtiene a partir del registro especial <code>CURRENT PATH</code> o de la opción de vinculación <code>FUNC_PATH</code> del paquete actual. |
| TRANSFORM_GROUP | Es la información sobre el grupo de transformación que se pasa al compilador de SQL. El valor se obtiene a partir del registro especial <code>CURRENT DEFAULT TRANSFORM GROUP</code> o de la opción de vinculación de paquetes <code>TRANSFORMGROUP</code> . |

Tabla 215. Elementos de un entorno de compilación devueltos por la función de tabla *COMPILATION_ENV* (continuación)

| Nombre de elemento | Descripción |
|---------------------------|---|
| MAINTAINED_TABLE _TYPE | Es indicador que se pasa al compilador de SQL y que denota qué tipos de tablas se pueden tener en cuenta para la optimización. El valor se obtiene a partir del registro especial CURRENT MAINTAINED TABLE TYPES FOR OPTIMIZATION. |
| RESOLUTION _TIMESTAMP | Es la indicación de fecha y hora que el compilador de SQL debe utilizar para resolver elementos tales como las referencias a funciones y tipos de datos contenidas en una sentencia de SQL. Esta indicación de fecha y hora es la indicación de fecha y hora actual o la indicación de fecha y hora de la última operación de vinculación explícita realizada para el paquete actual. |
| FEDERATED _ASYNCHRONY | El grado solicitado de paralelismo federado de asincronía que se pasa al compilador de SQL. El valor se obtiene a partir del registro especial CURRENT FEDERATED ASYNCHRONY o de la opción de vinculación FEDERATED_ASYNCHRONY del paquete actual. |

Vista administrativa CONTACTGROUPS - Recuperar la lista de grupos de contacto

La vista administrativa CONTACTGROUPS devuelve la lista de grupos de contactos, que se pueden definir localmente en el sistema o en una lista global. El valor del parámetro de configuración CONTACT_HOST del Servidor administrativo de bases de datos (DAS) determina si la lista es local o global.

El esquema es SYSIBMADM.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa CONTACTGROUPS
- Privilegio CONTROL en la vista administrativa CONTACTGROUPS
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE para la función de tabla ADMIN_GET_CONTACTGROUPS
- Autorización DATAACCESS

Ejemplo

Recuperar todas las listas de grupos de contactos.

```
SELECT * FROM SYSIBMADM.CONTACTGROUPS
```

El siguiente ejemplo muestra los datos de salida de la consulta.

| NAME | DESCRIPTION | MEMBERNAME | MEMBERTYPE |
|--------|-------------------------|------------|------------|
| group1 | DBA Group1 Contact List | name1 | CONTACT |
| group1 | DBA Group1 Contact List | name9 | CONTACT |
| group2 | DBA Group2 List | name2 | CONTACT |

| | | | |
|--------|------------|--------|---------|
| group3 | | group2 | GROUP |
| group5 | DBA Group5 | group2 | GROUP |
| group6 | DBA Group6 | group3 | GROUP |
| group7 | | name1 | CONTACT |

7 registro(s) seleccionado(s).

Nota sobre uso

El servidor DAS debe haberse creado y estar en ejecución.

Información devuelta

Tabla 216. Información que devuelve la vista administrativa CONTACTGROUPS

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| NAME | VARCHAR(128) | Nombre del grupo de contactos. |
| DESCRIPTION | VARCHAR(128) | Descripción del grupo de contactos. |
| MEMBERNAME | VARCHAR(128) | Nombre del miembro del grupo de contactos. Este nombre puede hacer referencia a un contacto o a otro grupo de contactos. |
| MEMBERTYPE | VARCHAR(7) | Tipo de miembro del grupo de contactos. El tipo puede ser CONTACT o GROUP. |

Vista administrativa CONTACTS - Recuperar lista de contactos

La vista administrativa CONTACTS devuelve la lista de contactos definidos en el servidor de bases de datos. El valor del parámetro de configuración CONTACT_HOST del Servidor administrativo de bases de datos (DAS) determina si la lista es local o global.

El esquema es SYSIBMADM.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa CONTACTS
- Privilegio CONTROL en la vista administrativa CONTACTS
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE para la función de tabla ADMIN_GET_CONTACTS
- Autorización DATAACCESS

Ejemplo

Recuperar todos los contactos.

```
SELECT * FROM SYSIBMADM.CONTACTS
```


El siguiente ejemplo muestra los datos de salida de la consulta.

```

NAME      TYPE  ADDRESS                MAX_PAGE_LENGTH  DESCRIPTION
-----
user1     EMAIL user3@ca.ibm.com      - DBA Extraordinaire
user2     EMAIL user2@ca.ibm.com      - DBA on Email
user3     PAGE  user3@ca.ibm.com      128 DBA on Page
user5     EMAIL user2@ca.ibm.com      - DBA Extraordinaire

```

4 registro(s) seleccionado(s).

Nota sobre uso

El servidor DAS debe haberse creado y estar en ejecución.

Información devuelta

Tabla 217. Información devuelta por la vista administrativa CONTACTS

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| NAME | VARCHAR(128) | Nombre del contacto. |
| TYPE | VARCHAR(5) | Tipo de contacto: <ul style="list-style-type: none"> • 'EMAIL' • 'PAGE' |
| ADDRESS | VARCHAR(128) | Dirección de buzón de correo SMTP del receptor. Por ejemplo, joe@somewhere.org. |
| MAX_PAGE_LENGTH | INTEGER | Longitud máxima del mensaje. Se utiliza, por ejemplo, si el servicio de paginación tiene una restricción con respecto a la longitud de los mensajes. |
| DESCRIPTION | VARCHAR(128) | Descripción del contacto. |

Vista administrativa DB_HISTORY - Recuperar información de archivo histórico

La vista administrativa DB_HISTORY devuelve información procedente de los archivos históricos de todas las particiones de bases de datos.

El esquema es SYSIBMADM.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT sobre la vista administrativa DB_HISTORY
- Privilegio CONTROL sobre la vista administrativa DB_HISTORY
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE sobre la función de tabla ADMIN_LIST_HIST
- Autorización DATAACCESS

Ejemplo

Seleccionar el número de partición de base de datos, ID de entrada, operación, hora de inicio e información de estado de los archivos históricos de base de datos para todas las particiones de base de datos a las que el cliente está actualmente conectado.

```
SELECT DBPARTITIONNUM, EID, OPERATION, START_TIME, ENTRY_STATUS
FROM SYSIBMADM.DB_HISTORY
```

El siguiente ejemplo muestra los datos de salida de la consulta.

```
DBPARTITIONNUM EID OPERATION START_TIME ENTRY_STATUS
-----
0 1 A 20051109185510 A
```

1 registro(s) seleccionado(s).

Información devuelta

Tabla 218. Información devuelta por la vista administrativa DB_HISTORY

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| DBPARTITIONNUM | SMALLINT | Número de partición de base de datos. |
| EID | BIGINT | Número que identifica de forma exclusiva una entrada del archivo histórico. |
| START_TIME | VARCHAR(14) | Indicación de fecha y hora que marca el inicio de un suceso anotado cronológicamente. |
| SEQNUM | SMALLINT | Número de secuencia. |
| END_TIME | VARCHAR(14) | Indicación de fecha y hora que marca el fin de un suceso anotado cronológicamente. |
| FIRSTLOG | VARCHAR(254) | Nombre de la primera anotación cronológica de transacción asociada a un suceso. |
| LASTLOG | VARCHAR(254) | Nombre de la última anotación cronológica de transacción asociada a un suceso. |
| BACKUP_ID | VARCHAR(24) | Identificador de copia de seguridad o identificador exclusivo de tabla. |
| TABSCHEMA | VARCHAR(128) | Esquema de tabla. |
| TABNAME | VARCHAR(128) | Nombre de tabla. |
| COMMENT | VARCHAR(254) | Texto de comentario generado por el sistema asociado a un suceso anotado cronológicamente. |
| CMD_TEXT | CLOB(2 M) | Idioma de definición de datos asociado a un suceso anotado cronológicamente. |

Tabla 218. Información devuelta por la vista administrativa DB_HISTORY (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| NUM_TBSPS | INTEGER | Número de espacios de tablas asociados a un suceso anotado cronológicamente. |
| TBSPNAMES | CLOB(5 M) | Nombres de los espacios de tablas asociados a un suceso anotado cronológicamente. |
| OPERATION | CHAR(1) | Identificador de la operación. Consulte la Tabla 219 en la página 953 para ver los valores posibles. |
| OPERATIONTYPE | CHAR(1) | Identificador de acción de una operación. Consulte la Tabla 219 en la página 953 para ver los valores posibles. |
| OBJECTTYPE | CHAR(1) | Identificador del objeto de destino de una operación. Los valores posibles son: D para base de datos completa, P para espacio de tablas y T para tabla. |
| LOCATION | VARCHAR(255) | Nombre completo de vía de acceso para archivos, como imágenes de copia de seguridad o archivo de entrada de carga, asociados a sucesos anotados cronológicamente. |
| DEVICETYPE | CHAR(1) | Identificador del tipo de dispositivo asociado a un suceso anotado cronológicamente. Este campo determina cómo se interpreta el campo LOCATION. Los valores posibles son: A para TSM, C para cliente, D para disco, F para copia de seguridad de instantáneas, K para disquete, L para entorno local, N (generado internamente por DB2), O para otro (para otro soporte de dispositivos de proveedores), P para conducto, Q para cursor, R para datos de captación remota, S para servidor, T para cinta, U para salida de usuario y X para interfaz X/Open XBSA. |

Tabla 218. Información devuelta por la vista administrativa DB_HISTORY (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| ENTRY_STATUS | CHAR(1) | Identificador del estado de una entrada del archivo histórico. Los valores posibles son: A para activo, D para suprimido (uso futuro), E para caducado, I para inactivo, N para aún no confirmado, Y para confirmado o activo. |
| SQLCAID | VARCHAR(8) | Un "observador" de vuelcos de almacenamiento que contienen 'SQLCA', tal como aparece en el campo SQLCAID del área de comunicaciones SQL (SQLCA). |
| SQLCABC | INTEGER | Longitud del SQLCA, tal como aparece en el campo SQLCABC del SQLCA. |
| SQLCODE | INTEGER | Código de retorno de SQL, tal como aparece en el campo SQLCODE del SQLCA. |
| SQLERRML | SMALLINT | Indicador de longitud correspondiente a SQLERRMC, tal como aparece en el campo SQLERRML del SQLCA. |
| SQLERRMC | VARCHAR(70) | Contiene uno o más símbolos, separados por X'FF', tal como aparecen en el campo SQLERRMC del SQLCA. Estos símbolos se sustituyen por variables en las descripciones de condiciones de error. |
| SQLERRP | VARCHAR(8) | Identificador de tres letras que indica el producto, seguido de cinco caracteres alfanuméricos que indican la versión, el release y el nivel de modificación del producto, tal como aparecen en el campo SQLERRP del SQLCA. |
| SQLERRD1 | INTEGER | Consulte SQLCA (área de comunicaciones SQL). |
| SQLERRD2 | INTEGER | Consulte SQLCA (área de comunicaciones SQL). |
| SQLERRD3 | INTEGER | Consulte SQLCA (área de comunicaciones SQL). |
| SQLERRD4 | INTEGER | Consulte SQLCA (área de comunicaciones SQL). |

Tabla 218. Información devuelta por la vista administrativa DB_HISTORY (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| SQLERRD5 | INTEGER | Consulte SQLCA (área de comunicaciones SQL). |
| SQLERRD6 | INTEGER | Consulte SQLCA (área de comunicaciones SQL). |
| SQLWARN | VARCHAR(11) | Conjunto de indicadores de aviso, cada uno de los cuales contiene un blanco o 'W'. Consulte SQLCA (área de comunicaciones SQL). |
| SQLSTATE | VARCHAR(5) | Código de retorno que indica el resultado de la última sentencia SQL ejecutada, tal como aparece en el campo SQLSTATE del SQLCA. |

Tabla 219. Valores de OPERATION y OPERATIONTYPE

| Valor de la operación | Descripción del valor de la operación | Tipo de operación |
|-----------------------|---------------------------------------|--|
| A | Añadir espacio de tablas | Ninguna |
| B | Hacer copia de seguridad | Los tipos de operación son: <ul style="list-style-type: none"> • D = fuera de línea delta • E = en línea delta • F = fuera de línea • I = fuera de línea incremental • N = en línea • O = en línea incremental |
| C | Cargar copia | Ninguna |
| D | Tabla descartada | Ninguna |
| F | Realizar recuperación en avance | Los tipos de operación son: <ul style="list-style-type: none"> • E = fin de anotaciones cronológicas • P = punto en el tiempo |
| G | Reorganizar tabla | Los tipos de operación son: <ul style="list-style-type: none"> • F = fuera de línea • N = en línea |
| L | Cargar | Los tipos de operación son: <ul style="list-style-type: none"> • I = insertar • R = sustituir |
| N | Cambiar nombre de espacio de tablas | Ninguna |
| O | Descartar espacio de tablas | Ninguna |

Tabla 219. Valores de OPERATION y OPERATIONTYPE (continuación)

| Valor de la operación | Descripción del valor de la operación | Tipo de operación |
|-----------------------|---------------------------------------|---|
| Q | Inmovilizar | Los tipos de operación son: <ul style="list-style-type: none"> • S = inmovilizar compartición • U = inmovilizar actualización • X = inmovilizar exclusividad • Z = inmovilizar restablecimiento |
| R | Restaurar | Los tipos de operación son: <ul style="list-style-type: none"> • F = fuera de línea • I = fuera de línea incremental • N = en línea • O = en línea incremental • R = volver a crear |
| T | Modificar espacio de tablas | Los tipos de operación son: <ul style="list-style-type: none"> • C = añadir contenedores • R = volver a equilibrar |
| U | Descargar | Ninguna |
| X | Archivar anotaciones cronológicas | Los tipos de operación son: <ul style="list-style-type: none"> • F = vía de acceso de archivado fallido • M = vía de acceso de anotaciones cronológicas de duplicación • N = truncamiento forzado mediante mandato ARCHIVE LOG • P = vía de acceso de anotaciones cronológicas primarias • 1 = primer método de archivado de anotaciones cronológicas • 2 = segundo método de archivado de anotaciones cronológicas |

Vista administrativa DBPATHS – Recuperar vías de acceso de bases de datos

La vista administrativa DBPATHS devuelve los valores de las vías de acceso de bases de datos necesarias para tareas como dividir copias de seguridad duplicadas.

El esquema es SYSIBMADM.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa DBPATHS

- Privilegio CONTROL en la vista administrativa DBPATHS
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE en la función de tabla ADMIN_LIST_DB_PATHS
- Autorización DATAACCESS

Ejemplo

Recuperar todas las vías de acceso de bases de datos.

```
SELECT * FROM SYSIBMADM.DBPATHS
```

El siguiente ejemplo muestra los datos de salida de la consulta.

```
DBPARTITIONNUM TYPE ...
-----
0 LOGPATH ...
0 MIRRORLOGPATH ...
0 DB_STORAGE_PATH ...
0 DB_STORAGE_PATH ...
0 TBSP_CONTAINER ...
0 TBSP_CONTAINER ...
0 TBSP_CONTAINER ...
0 TBSP_DIRECTORY ...
0 TBSP_DIRECTORY ...
0 LOCAL_DB_DIRECTORY ...
0 DBPATH ...
```

11 registro(s) seleccionados.

Datos de salida de esta consulta (continuación).

```
... PATH
... -----
... S:\dbfiles\INST5\NODE0000\SQL00001\SQLLOGDIR\
... S:\mirrorlogs\NODE0000\
... S:\dbfiles\
... S:\dbfile2\
... S:\dbfiles\INST5\NODE0000\SQL00001\TS3
... S:\dbfiles\INST5\NODE0000\SQL00001\long3
... S:\dbfiles\INST5\NODE0000\SQL00001\regular05
... S:\dbfiles\INST5\NODE0000\SQL00001\usertemp3\
... S:\dbfiles\INST5\NODE0000\SQL00001\systemp3\
... S:\dbfiles\INST5\NODE0000\SQLDBDIR\
... S:\dbfiles\INST5\NODE0000\SQL00001\
```

Información devuelta

Tabla 220. Información devuelta por la vista administrativa DBPATHS

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---------------------------------------|
| DBPARTITIONNUM | SMALLINT | Número de partición de base de datos. |

Tabla 220. Información devuelta por la vista administrativa DBPATHS (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| TYPE | VARCHAR(64) | Describe el tipo de objeto de base de datos al que pertenece la vía de acceso. Por ejemplo, la vía de acceso al directorio de anotaciones cronológicas que indica el parámetro de configuración de base de datos LOGPATH se mostraría en esta columna como LOGPATH. Consulte la Tabla 221 para ver una lista de los posibles valores de retorno. |
| PATH | VARCHAR(5000) | Vía de acceso a la ubicación en la que el gestor de bases de datos tiene un archivo o directorio ubicado. Si la vía de acceso termina con el delimitador del sistema de archivos ('/' en entornos UNIX, '\' en entornos Windows), la vía de acceso apunta a un directorio. |

Tabla 221. Valores de la columna TYPE

| Valor de tipo | Descripción |
|--------------------|---|
| TBSP_DEVICE | Dispositivo en bruto correspondiente a un espacio de tablas del espacio gestionado por la base de datos (DMS) |
| TBSP_CONTAINER | Contenedor de archivos correspondiente a un espacio de tablas DMS. |
| TBSP_DIRECTORY | Directorio correspondiente a un espacio de tablas del espacio gestionado por el sistema (SMS). |
| LOGPATH | Vía de acceso primaria de las anotaciones cronológicas. |
| LOGPATH_DEVICE | Dispositivo en bruto correspondiente a la vía de acceso primaria de las anotaciones cronológicas. |
| MIRRORLOGPATH | Vía de acceso de anotaciones cronológicas de duplicación de la configuración de la base de datos. |
| DB_STORAGE_PATH | Vía de acceso del almacenamiento automático. |
| DBPATH | Vía de acceso del directorio de la base de datos. |
| LOCAL_DB_DIRECTORY | Vía de acceso al directorio de bases de datos locales. |

- Para espacios de tablas que utilizan almacenamiento automático, se devuelve la vía de acceso del almacenamiento utilizado y del no utilizado. Las vías de acceso del almacenamiento automático no utilizado se necesita en el caso de que

se restaure la copia de seguridad de duplicación dividida. Tenga en cuenta el siguiente ejemplo: una copia de seguridad de duplicación dividida se coloca en un sistema de producción. Una vez finalizada la copia de seguridad, las vías de acceso del almacenamiento automático que no se utilizaban antes de la copia de seguridad ahora se utilizan en producción. Supongamos que ahora hay que restaurar la copia de seguridad de duplicación dividida. En este momento, hay que realizar una recuperación en avance de las anotaciones cronológicas de la base de datos de producción. Para poder realizar la recuperación en avance de las anotaciones cronológicas, se necesitan todas las vías de acceso del almacenamiento automático, puesto que ahora todas ellas se utilizan.

- Los contenedores de espacios de tablas gestionados por el almacenamiento automático no se devuelven de forma individual; quedan reflejados en la columna de vía de acceso del almacenamiento automático.
- Las vías de acceso del almacenamiento automático se devuelven una vez por cada partición de base de datos.
- Los valores devueltos correspondientes a LOGPATH y MIRRORLOGPATH son los valores almacenados en memoria. Los valores modificados almacenados en disco, que sólo se aplican después de volver a iniciar una base de datos, no se devuelven.
- Si la salida desde `SELECT * FROM SYSIBMADM.DBPATHS` se está utilizando para crear un archivo de configuración `db2relocatedb` (un archivo que contiene la información necesaria sobre la configuración para reubicar una base de datos), la salida `DBPATH` debe modificarse como corresponda antes de que pueda utilizarse en el archivo de configuración.

Por ejemplo, la siguiente salida `DBPATH`:

```
/storage/svtdbm3/svtdbm3/NODE0000/SQL00001/
```

se puede utilizar para especificar el parámetro `DB_PATH` en un archivo de configuración `db2relocatedb` del modo siguiente:

```
DB_PATH=/storage/svtdbm3,/storage_copy2/svtdbm3
```

- La vía de acceso `LOCAL_DB_DIRECTORY` puede contener información correspondiente a varias bases de datos. Debido a que `sqlldbid` no está separado para varias bases de datos creadas en el mismo directorio, asegúrese de que el sistema destino en el que se copiarán los archivos no existen ya bases de datos en dicha vía de acceso.
- Si dos o más bases de datos comparten como mínimo una vía de acceso de almacenamiento automático, la operación de división de la duplicación afectará a más de una base de datos, provocando problemas de E/S para las bases de datos que no debían dividirse.

Restricción

La vista administrativa no se puede abrir cuando la base de datos está en modalidad `WRITE SUSPEND`. El administrador de bases de datos debe asegurarse de que el diseño físico de la base de datos no cambia entre la invocación de la vista y la activación de la modalidad `WRITE SUSPEND`, que se necesita para realizar la operación de duplicación dividida. Es posible que la imagen de copia de seguridad de duplicación dividida no se pueda restaurar correctamente si, por ejemplo, el diseño de espacio de tablas ha cambiado en ese tiempo.

EXPLAIN_FORMAT_STATS

La nueva función escalar se utiliza para mostrar información estadística formateada analizada y extraída de una instantánea de explicación capturada para una consulta determinada. El tipo de datos del resultado es CLOB(50M).

Sintaxis

►►—EXPLAIN_FORMAT_STATS—(—*instantánea*—)—————►►

El esquema es SYSPROC.

Parámetros de la función

instantánea

Argumento de entrada de tipo BLOB(10M) que incluye la instantánea de explicación capturada para una consulta determinada. Se almacena como columna instantánea de la tabla de explicación *EXPLAIN_STATEMENT*

Autorización

Privilegio EXECUTE en la función EXPLAIN_FORMAT_STATS.

Ejemplo

```
SELECT EXPLAIN_FORMAT_STATS(SNAPSHOT)
FROM EXPLAIN_STATEMENT
WHERE EXPLAIN_REQUESTER = 'DB2USER1' AND
      EXPLAIN_TIME = timestamp('2006-05-12-14.38.11.109432') AND
      SOURCE_NAME = 'SQLC2F0A' AND
      SOURCE_SCHEMA = 'NULLID' AND
      SOURCE_VERSION = '' AND
      EXPLAIN_LEVEL = '0' AND
      STMTNO = 1 AND
      SECTNO = 201
```

A continuación presentamos una salida de ejemplo de esta función:

Contexto espacio de tablas:

| | |
|---|--------------------------|
| Nombre: | USERSPACE1 |
| Actividad general: | 7.500000 |
| Velocidad de transferencia: | 0.060000 |
| Tamaño de captación previa: | 32 |
| Tamaño de extensión: | 32 |
| Tipo: | Base de datos gestionada |
| Nombre del grupo de particiones: | NULLP |
| Identificador agrup. de almac. intermedios: | 0 |

Estadísticas de tablas básicas:

| | |
|--|----------------------------|
| Nombre : | T1 |
| Esquema: | DB2USER2 |
| Número de columnas: | 3 |
| Número de páginas con filas: | 1 |
| Número de páginas: | 1 |
| Número de filas: | 5 |
| Recuento registros desbordamiento tabla: | 0 |
| Anchura de las filas: | 26 |
| Fecha y hora de creación: | 2006-06-16-11.46.53.041085 |
| Última actualización de estadísticas: | 2006-06-26-12.23.44.814201 |
| Tipo de estadísticas: | Fabricación |

Espacio de tablas primario: USERSPACE1
 Espacio de tablas para índices: USERSPACE1
 Espacio de tablas para datos largos: NULLP
 Número de columnas referenciadas: 2
 Número de índices: 1
 Tabla volátil: No
 Bloqueos activos de tabla: 1
 Número de grupos de columna: 0
 Número de particiones de datos: 1
 Proporción media de compresión por fila: -9.000000
 Porcentaje de filas comprimidas: -9.000000
 Tamaño medio de filas comprimidas: -9
 Tipo de estadísticas: U

Información de la columna:

Número: 1
 Nombre: C1
 Estadísticas disponibles: Sí

Estadísticas de columna:

Nombre del esquema del tipo de columna: SYSIBM
 Nombre del tipo de columna: INTEGER
 Longitud máxima de columna: 4
 Escala para la columna decimal: 0
 Número de valores de columna distintos: 4
 Longitud media de la columna: 5
 Número de valores más frecuentes: 1
 Número de cuantiles: 5
 Segundo valor de datos más alto: 3
 Segundo valor de datos más bajo: 2
 Secuencia de columnas en clave de partición: 0
 Promedio de subelementos: -1
 Longitud media de delimitadores: -1

Estadísticas de distribución de columna:

Estadísticas de frecuencia:

| Valcount | Value |
|----------|-------|
| 2 | 1 |

Estadísticas de cuantiles:

| Valcount | Distcount | Value |
|----------|-----------|-------|
| 0 | 1 | 1 |
| 2 | 1 | 1 |
| 3 | 2 | 2 |
| 4 | 3 | 3 |
| 5 | 4 | 4 |

Información de la columna:

Número: 2
 Nombre: C2
 Estadísticas disponibles: Sí

Estadísticas de columna:

Nombre del esquema del tipo de columna: SYSIBM
 Nombre del tipo de columna: INTEGER
 Longitud máxima de columna: 4
 Escala para la columna decimal: 0
 Número de valores de columna distintos: 4
 Longitud media de la columna: 5
 Número de valores más frecuentes: 1

Número de cuantiles: 5
 Segundo valor de datos más alto: 3
 Segundo valor de datos más bajo: 2
 Secuencia de columnas en clave de partición: 0
 Promedio de subelementos: -1
 Longitud media de delimitadores: -1

Estadísticas de distribución de columna:

 Estadísticas de frecuencia:
 Valcount Value

 2 1

Estadísticas de cuantiles:
 Valcount Distcount Value

 0 0 1
 2 0 1
 3 0 2
 4 0 4
 5 0 4

Índices definidos en la tabla:

 Nombre: IDX_T1C1C2
 Esquema: DB2USER2
 Norma exclusiva: Índice duplicado
 Utilizado en el operador: Sí
 Pares de captación de páginas: No disponible
 Número de columnas: 2
 Páginas hojas de índice: 1
 Niveles de árbol de índice: 1
 Cardinalidad de primera clave de índice: 4
 Cardinalidad de clave completa de índice: 4
 Proporción de clúster de índice: 100
 Factor de clúster de índice: -1.000000
 Fecha y hora de creación: 2006-06-16-11.46.53.596717
 Última actualización de estadísticas: 2006-06-26-12.23.44.814201
 Páginas secuenciales de índice: 0
 Cardinalidad de 2 primeras claves de índice: 4
 Cardinalidad de 3 primeras claves de índice: -1
 Cardinalidad de 4 primeras claves de índice: -1
 Intervalo medio de índice entre secuencias: 0.000000
 Intervalo medio de captación entre secuencias: -1.000000
 Promedio de páginas secuenciales del índice: 0.000000
 Promedio de páginas secuenciales de captación: -1.000000
 Promedio de páginas aleatorias del índice: 1.000000
 Promedio de páginas aleatorias de captación: -1.000000
 Recuento RID de índice: 5
 Recuento RID borrado de índice: 0
 Páginas de hojas vacías de índice: 0
 Proporción media de clúster de partición: -1
 Factor de clúster medio de partición: -1.000000
 Factor de clúster de partición de datos: 1.000000
 Pares de captación de pág. de partición de datos: No disponible

Estadísticas de tablas básicas:

 Nombre : T2
 Esquema: DB2USER2
 Número de columnas: 3
 Número de páginas con filas: 1
 Número de páginas: 1
 Número de filas: 2
 Recuento registros desbordamiento tabla: 0
 Anchura de las filas: 26

```

Fecha y hora de creación:          2006-06-16-11.46.53.398092
Última actualización de estadísticas: 2006-06-26-12.23.45.157028
Tipo de estadísticas:              Síncrona
Espacio de tablas primario:        USERSPACE1
Espacio de tablas para índices:    USERSPACE1
Espacio de tablas para datos largos: NULLP
Número de columnas referenciadas:  2
Número de índices:                 1
Tabla volátil:                     No
Bloqueos activos de tabla:         -1
Número de grupos de columna:       0
Número de particiones de datos:    1

```

Información de la columna:

```

Número:          1
Nombre:          C1
Estadísticas disponibles: Sí

```

Estadísticas de columna:

```

Nombre del esquema del tipo de columna:  SYSIBM
Nombre del tipo de columna:              INTEGER
Longitud máxima de columna:              4
Escala para la columna decimal:          0
Número de valores de columna distintos:  2
Longitud media de la columna:             5
Número de valores más frecuentes:         -1
Número de cuantiles:                     2
Segundo valor de datos más alto:          2
Segundo valor de datos más bajo:         1
Secuencia de columnas en clave de partición: 0
Promedio de subelementos:                -1
Longitud media de delimitadores:         -1

```

Estadísticas de distribución de columna:

Estadísticas de cuantiles:

| Valcount | Distcount | Value |
|----------|-----------|-------|
| 1 | 1 | 1 |
| 2 | 2 | 2 |

Información de la columna:

```

Número:          2
Nombre:          C2
Estadísticas disponibles: Sí

```

Estadísticas de columna:

```

Nombre del esquema del tipo de columna:  SYSIBM
Nombre del tipo de columna:              INTEGER
Longitud máxima de columna:              4
Escala para la columna decimal:          0
Número de valores de columna distintos:  2
Longitud media de la columna:             5
Número de valores más frecuentes:         -1
Número de cuantiles:                     2
Segundo valor de datos más alto:          2
Segundo valor de datos más bajo:         1
Secuencia de columnas en clave de partición: 0
Promedio de subelementos:                -1
Longitud media de delimitadores:         -1

```

Estadísticas de distribución de columna:

| Estadísticas de cuantiles: | | |
|----------------------------|-----------|-------|
| Valcount | Distcount | Value |
| 1 | 0 | 1 |
| 2 | 0 | 2 |

Índices definidos en la tabla:

| | | |
|--|-----------|----------------------------|
| Nombre | :IDX_T2C1 | Índice duplicado |
| Esquema: | DB2USER2 | No |
| Norma exclusiva: | | No disponible |
| Utilizado en el operador: | | 1 |
| Pares de captación de páginas: | | 1 |
| Número de columnas: | | 1 |
| Páginas hojas de índice: | | 1 |
| Niveles de árbol de índice: | | 2 |
| Cardinalidad de primera clave de índice: | | 2 |
| Cardinalidad de clave completa de índice: | | 100 |
| Proporción de clúster de índice: | | -1.000000 |
| Factor de clúster de índice: | | 2006-06-16-11.46.53.857520 |
| Fecha y hora de creación: | | 2006-06-26-12.23.45.157028 |
| Última actualización de estadísticas: | | 0 |
| Páginas secuenciales de índice: | | -1 |
| Cardinalidad de 2 primeras claves de índice: | | -1 |
| Cardinalidad de 3 primeras claves de índice: | | -1 |
| Cardinalidad de 4 primeras claves de índice: | | 0.000000 |
| Intervalo medio de índice entre secuencias: | | -1.000000 |
| Intervalo medio de captación entre secuencias: | | 0.000000 |
| Promedio de páginas secuenciales del índice: | | -1.000000 |
| Promedio de páginas secuenciales de captación: | | 1.000000 |
| Promedio de páginas aleatorias del índice: | | -1.000000 |
| Promedio de páginas aleatorias de captación: | | 2 |
| Recuento RID de índice: | | 0 |
| Recuento RID borrado de índice: | | 0 |
| Páginas de hojas vacías de índice: | | -1 |
| Proporción media de clúster de partición: | | -1.000000 |
| Factor de clúster medio de partición: | | 1.000000 |
| Factor de clúster de partición de datos: | | No disponible |
| Pares de captación de pág. partición de datos: | | |

EXPLAIN_GET_MSGS

La función de tabla EXPLAIN_GET_MSGS consulta las tablas Explain EXPLAIN_DIAGNOSTIC y EXPLAIN_DIAGNOSTIC_DATA, y devuelve los mensajes con formato.

Sintaxis

```

▶▶—EXPLAIN_GET_MSGS—(—petionario-explain—,—hora-explain—,——————▶
▶—nombre-fuente—,—esquema-fuente—,—versión-fuente—,—nivel-explain—,—————▶
▶—núm-sent—,—núm-sec—,—entorno-local—)——————▶▶

```

El esquema es el mismo que el de la tabla Explain.

Parámetros de la función de tabla

Cualquiera de los siguientes argumentos de entrada puede ser nulo. Si un argumento es nulo, no se utiliza para limitar la consulta.

petionario-explain

Un Argumento de entrada de tipo VARCHAR(128) que especifica el ID de autorización del iniciador de esta petición de Explain. Un valor nulo excluye este parámetro de la condición de búsqueda de la consulta.

hora-explain

Argumento de entrada del tipo TIMESTAMP que especifica la hora de inicio de la petición de Explain. Un valor nulo excluye este parámetro de la condición de búsqueda de la consulta.

nombre-fuente

Argumento de entrada de tipo VARCHAR(128) que especifica el nombre del paquete en ejecución cuando se explicó la sentencia dinámica, o bien el nombre del archivo fuente cuando se explicó la sentencia SQL estática. Un valor nulo excluye este parámetro de la condición de búsqueda de la consulta.

esquema-fuente

Argumento de entrada de tipo VARCHAR(128) que especifica el esquema o el calificador de la fuente de la petición de Explain. Un valor nulo excluye este parámetro de la condición de búsqueda de la consulta.

versión-fuente

Argumento de entrada de tipo VARCHAR(64) que especifica la versión de la fuente de la petición de Explain. Un valor nulo excluye este parámetro de la condición de búsqueda de la consulta.

nivel-explain

Argumento de entrada del tipo CHAR(1) que especifica el nivel de información de Explain para el que esta fila es aplicable. Un valor nulo excluye este parámetro de la condición de búsqueda de la consulta.

núm-sent

Argumento de entrada del tipo INTEGER que especifica el número de sentencias del paquete con las que se relaciona esta información de Explain. Un valor nulo excluye este parámetro de la condición de búsqueda de la consulta.

núm-sec

Argumento de entrada de tipo INTEGER que especifica el número de secciones del paquete con las que se relaciona esta información de Explain. Un valor nulo excluye este parámetro de la condición de búsqueda de la consulta.

entorno-local

Argumento de entrada de tipo VARCHAR(33) que especifica el entorno local de los mensajes devueltos. Si el entorno local especificado no está instalado en el servidor DB2, se ignora este valor.

Información devuelta

Tabla 222. Información devuelta por la función de tabla EXPLAIN_GET_MSGS

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| EXPLAIN_REQUESTER | VARCHAR(128) | ID de autorización del iniciador de esta petición de Explain. |
| EXPLAIN_TIME | TIMESTAMP | Hora de inicio de la petición de Explain. |

Tabla 222. Información devuelta por la función de tabla EXPLAIN_GET_MSGS (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| SOURCE_NAME | VARCHAR(128) | Nombre del paquete en ejecución cuando se ha explicado la sentencia dinámica o el nombre del archivo fuente cuando se ha explicado la sentencia SQL estática. |
| SOURCE_SCHEMA | VARCHAR(128) | Esquema o calificador de la fuente de la petición de Explain. |
| SOURCE_VERSION | VARCHAR(64) | Versión de la fuente de la petición de Explain. |
| EXPLAIN_LEVEL | CHAR(1) | Nivel de información de Explain para el que esta fila es aplicable. |
| STMTNO | INTEGER | Número de sentencias del paquete con las que se relaciona esta información de Explain. |
| SECTNO | INTEGER | Número de secciones del paquete con las que se relaciona esta información de Explain. |
| DIAGNOSTIC_ID | INTEGER | ID del diagnóstico para una instancia concreta de una sentencia en la tabla EXPLAIN_STATEMENT. |
| LOCALE | VARCHAR(33) | Entorno local de los mensajes devueltos. Este entorno local no coincidirá con el entorno local especificado si este último no está instalado en el servidor DB2. |
| MSG | VARCHAR(4096) | Texto del mensaje con formato. |

Ejemplos

Solicitar mensajes con formato en inglés de las tablas Explain en el esquema por omisión para el peticionario SIMMEN que se han generado en la última hora. Especificar un nombre fuente de SQLC2E03.

```
SELECT MSG
FROM TABLE(EXPLAIN_GET_MSGS(
'SIMMEN',
CAST(NULL AS TIMESTAMP),
'SQLC2E03',
CAST(NULL AS VARCHAR(128)),
CAST(NULL AS VARCHAR(64)),
CAST(NULL AS CHAR(1)),
CAST(NULL AS INTEGER),
CAST(NULL AS INTEGER),
'en_US'))
AS REGISTRYINFO
WHERE EXPLAIN_TIME >= (CURRENT_TIMESTAMP - 1 HOUR)
ORDER BY DIAGNOSTIC_ID
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
MSG
-----
EXP0012W  Petición de acceso no válida. No se ha podido encontrar el índice "index1".
          Número de línea "554", número de carácter "20".
EXP0012W  Petición de acceso no válida. No se ha podido encontrar el índice "index2".
```


Número de línea "573", número de carácter "20".
EXP0015W Petición de unión no válida. Unión hace referencia a tablas que no están en la misma cláusula FROM. Número de línea "573", número de carácter "20".

GET_DBSIZE_INFO

El procedimiento GET_DBSIZE_INFO calcula el tamaño y capacidad máxima de una base de datos.

Sintaxis

```
►►—GET_DBSIZE_INFO—(—indicaciónfechahora-instantánea—,—tamaño-bd—,——————►  
►—capacidad-bd—,—ventana-renovación—)—————►
```

El esquema es SYSPROC.

Parámetros del procedimiento

indicaciónfechahora-instantánea

Parámetro de salida de tipo TIMESTAMP que devuelve la hora en la que se han calculado el *tamaño-bd* y la *capacidad-bd*. Esta indicación de fecha y hora, junto con el valor de *ventana-renovación*, se utiliza para determinar cuándo se tienen que renovar los valores de antememoria de la tabla SYSTOOLS.STMG_DBSIZE_INFO.

tamaño-bd

Parámetro de salida de tipo BIGINT que devuelve el tamaño de la base de datos (en bytes). El tamaño de la base de datos se calcula de este modo:
 $tamaño-bd = \text{sum}(\text{páginas_utilizadas} * \text{tamaño_página})$ para cada espacio de tablas (SMS & DMS).

capacidad-bd

Parámetro de salida de tipo BIGINT que devuelve la capacidad de la base de datos (en bytes). Este valor no está disponible en los sistemas de bases de datos particionadas. La capacidad de la base de datos se calcula de este modo:
 $capacidad-bd = \text{SUM}(\text{páginas_utilizables DMS} * \text{tamaño página}) + \text{SUM}(\text{tamaño del contenedor SMS} + \text{tamaño libre del sistema de archivos por contenedor})$. Si se definen varios contenedores SMS en el mismo sistema de archivos, el tamaño libre del sistema de archivos se incluye sólo una vez en el cálculo de la capacidad.

ventana-renovación

Argumento de entrada de tipo INTEGER que especifica el número de minutos transcurridos hasta que es necesario renovar los valores de antememoria para el tamaño y capacidad de la base de datos. Especifique -1 para la ventana de renovación por omisión de 30 minutos. Una ventana de renovación de 0 impone la renovación inmediata de los valores de antememoria.

Autorización

- Autorización SYSMON
- Privilegio EXECUTE para el procedimiento GET_DBSIZE_INFO

Ejemplos

Ejemplo 1: Obtener el tamaño y capacidad de la base de datos utilizando una ventana de renovación por omisión de 30 minutos. El tamaño y la capacidad de la base de datos se volverán a calcular cuando los datos de antememoria tengan más de 30 minutos.

```
CALL GET_DBSIZE_INFO(?, ?, ?, -1)
```

El procedimiento devuelve:

Valor parámetros de salida

Nombre de parámetro : SNAPSHOTTIMESTAMP

Valor de parámetro : 2004-02-29-18.31.55.178000

Nombre de parámetro : DATABASESIZE

Valor de parámetro : 22302720

Nombre de parámetro : DATABASECAPACITY

Valor de parámetro : 4684793856

Estado de devolución = 0

Ejemplo 2: Obtener el tamaño y capacidad de la base de datos utilizando una ventana de renovación por omisión de 0 minutos. El tamaño y la capacidad de la base de datos se volverán a calcular inmediatamente.

```
CALL GET_DBSIZE_INFO(?, ?, ?, 0)
```

El procedimiento devuelve:

Valor parámetros de salida

Nombre de parámetro : SNAPSHOTTIMESTAMP

Valor de parámetro : 2004-02-29-18.33.34.561000

Nombre de parámetro : DATABASESIZE

Valor de parámetro : 22302720

Nombre de parámetro : DATABASECAPACITY

Valor de parámetro : 4684859392

Estado de devolución = 0

Ejemplo 3: Obtener el tamaño y capacidad de la base de datos utilizando una ventana de renovación de 24 horas. El tamaño y la capacidad de la base de datos se volverán a calcular cuando los datos de antememoria tengan más de 1440 minutos.

```
CALL GET_DBSIZE_INFO(?, ?, ?, 1440)
```

El procedimiento devuelve:

Valor parámetros de salida

Nombre de parámetro : SNAPSHOTTIMESTAMP

Valor de parámetro : 2004-02-29-18.33.34.561000

Nombre de parámetro : DATABASESIZE

Valor de parámetro : 22302720

Nombre de parámetro : DATABASECAPACITY

Valor de parámetro : 4684859392

Estado de devolución = 0

Notas de uso

Los valores calculados se devuelven como parámetros de salida del procedimiento y se colocan en antememoria en la tabla SYSTOOLS.STMG_DBSIZE_INFO. El procedimiento coloca estos valores en antememoria porque los cálculos son costosos. La tabla SYSTOOLS.STMG_DBSIZE_INFO se crea automáticamente la primera vez que se ejecuta el procedimiento. Si hay valores en antememoria en la tabla SYSTOOLS.STMG_DBSIZE_INFO y están lo suficientemente actualizados, como determinan los valores de *indicaciónfechahora-instantánea* y *ventana-renovación*, se devuelven dichos valores colocados en antememoria. Si los valores colocados en antememoria no están lo suficientemente actualizados, se calculan nuevos valores de antememoria, se insertan en la tabla SYSTOOLS.STMG_DBSIZE_INFO y se devuelven, y se actualiza el valor de *indicaciónfechahora-instantánea*.

Se debe activar la base de datos para asegurarse de que todas las particiones devuelven los datos para una instantánea de espacio de tablas global.

SYSTOOLSPACE se utiliza para las tablas de funcionamiento de la rutina para almacenar metadatos; es decir, datos utilizados para describir objetos de base de datos y su funcionamiento.

Vista administrativa NOTIFICATIONLIST - Recuperar lista de contactos para notificación de salud

La vista administrativa NOTIFICATIONLIST devuelve la lista de contactos y grupos de contactos a los que se notifica sobre la salud de una instancia.

El esquema es SYSIBMADM.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa NOTIFICATIONLIST
- Privilegio CONTROL en la vista administrativa NOTIFICATIONLIST
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE en la función de tabla HEALTH_GET_NOTIFICATION_LIST
- Autorización DATAACCESS

Ejemplo

Recuperar todos los contactos que recibirán alertas de salud.

```
SELECT * FROM SYSIBMADM.NOTIFICATIONLIST
```

El siguiente ejemplo muestra los datos de salida de la consulta.

| NAME | TYPE |
|--------|---------|
| ----- | ----- |
| group3 | GROUP |
| user4 | CONTACT |
| group3 | GROUP |

3 registros(s) seleccionado(s).

Información devuelta

Tabla 223. Información devuelta por la vista administrativa NOTIFICATIONLIST

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| NAME | VARCHAR(128) | Nombre del contacto. |
| TYPE | VARCHAR(7) | Tipo de contacto: <ul style="list-style-type: none">• 'CONTACT'• 'GROUP' |

PD_GET_DIAG_HIST - Devolver registros de un determinado recurso

La función de tabla PD_GET_DIAG_HIST devuelve registros cronológicos, registros de sucesos y registros de notificación de un determinado recurso. También se da soporte a opciones que permiten filtrar según el tipo de registro, el valor de impacto en cliente del registro y las indicaciones de fecha y hora desde-hasta.

Sintaxis

►► PD_GET_DIAG_HIST (—recurso—, —tiporeg—, —impacto—, —hora_inicio—, —hora_final—) ◀◀

El esquema es SYSPROC.

Parámetros de la función de tabla

recurso

Argumento de entrada opcional de tipo VARCHAR(20) que especifica el recurso desde el que se devolverán registros. Un recurso es una agrupación lógica con la que están relacionados los registros. Los valores posibles son:

- ALL: Devuelve registros de todos los recursos
- MAIN: Devuelve registros de los registros cronológicos de diagnósticos generales de DB2 Actualmente, esto significa los archivos de anotaciones cronológicas db2diag, el registro de notificaciones de administración y las anotaciones cronológicas rotatorias de sucesos.
- OPTSTATS: Devolver registros relacionados con estadísticas del optimizador

Si este parámetro es nulo o una serie vacía (''), 'ALL' es el valor por omisión.

tiporeg

Argumento de entrada opcional de tipo VARCHAR(30) que especifica qué tipo de registro se va a devolver. Se da soporte a una combinación de tipos separados por '+', por ejemplo: 'D + EI'. Los valores posibles son:

- 'ALL': Devolver todos los tipos de registro.
- 'D': Devolver todos los registros de diagnóstico.
- 'E': Devolver todos los registros de sucesos.
- 'DI': Registros de diagnóstico interno. Se trata de registros de diagnóstico no traducidos que utiliza el equipo de soporte de IBM en una situación de diagnóstico.
- 'DX': Registros de diagnóstico externo. Se trata de diagnósticos traducidos que utiliza el usuario. Estos registros son registros de notificación.
- 'EI': Registro de sucesos internos. Se trata de registros de sucesos que utiliza el equipo de soporte de IBM en una situación de diagnóstico.

- 'EX': Registro de sucesos externos. Se trata de registros de diagnóstico que utiliza el usuario.

Si este parámetro es nulo o una serie vacía (''), se devuelven todos los registros.

impacto

Argumento de entrada opcional de tipo VARCHAR (18) que especifica el nivel mínimo de impacto en el cliente del registro devuelto. Los valores posibles son:

- 'NONE'
- 'UNLIKELY'
- 'POTENTIAL'
- 'IMMEDIATE'
- 'CRITICAL'

Si este parámetro es nulo o una serie vacía (''), se devuelven todos los registros.

hora_inicio

Es un argumento de entrada opcional de tipo TIMESTAMP que especifica una indicación de fecha y hora válida. Se devuelven entradas si su indicación de fecha y hora es más reciente que este valor. Si este parámetro es nulo o una serie vacía (''), se devuelven registros independientemente de su antigüedad.

hora_final

Es un argumento de entrada opcional de tipo TIMESTAMP que especifica una indicación de fecha y hora válida. Se devuelven entradas si su indicación de fecha y hora es anterior a este valor. Si este parámetro es nulo o una serie vacía (''), se devuelven registros independientemente de su antigüedad.

Autorización

Privilegio EXECUTE sobre la función de tabla PD_GET_DIAG_HIST.

Ejemplo

```
SELECT FACILITY, RECTYPE, TIMESTAMP, IMPACT, SUBSTR(MSG,1, 50) AS MSG
FROM TABLE (PD_GET_DIAG_HIST( 'MAIN', 'E', '', CAST (NULL AS TIMESTAMP),
CAST (NULL AS TIMESTAMP) ) ) AS T
WHERE T.PROCESS_NAME = 'db2star2' OR T.PROCESS_NAME = 'db2stop2'
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| FACILITY | RECTYPE | TIMESTAMP | ... |
|----------|---------|----------------------------|-----|
| ----- | ----- | ----- | ... |
| MAIN | EX | 2007-06-25-11.34.05.756171 | ... |
| MAIN | EX | 2007-06-25-11.34.25.946646 | ... |

2 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

| ... IMPACT | MSG |
|------------|--|
| ... | ----- |
| ... - | ADM7514W Se ha detenido el gestor de bases de datos. |
| ... - | ADM7513W Se ha iniciado el gestor de bases de datos. |

Información devuelta

Tabla 224. Información devuelta por la función de tabla PD_GET_DIAG_HIST

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| FACILITY | VARCHAR(20) | Un recurso es una agrupación lógica con la que están relacionados los registros. Los valores posibles son: <ul style="list-style-type: none"> • ALL: Devuelve registros de todos los recursos • MAIN: Devuelve registros de los registros cronológicos de diagnósticos generales de DB2. Actualmente, esto significa los archivos de anotaciones cronológicas db2diag, el registro de notificaciones de administración y las anotaciones cronológicas rotatorias de sucesos. • OPTSTATS: Devolver registros relacionados con estadísticas del optimizador |
| RECTYPE | VARCHAR(3) | El tipo de registro. Los valores posibles son: <ul style="list-style-type: none"> • 'DI': Registro de diagnóstico interno • 'DX': Registro de diagnóstico externo • 'EI': Registro de sucesos internos • 'EX': Registro de sucesos externos |
| TIMESTAMP | TIMESTAMP | La hora a la que se ha creado el mensaje. |
| TIMEZONE | INTEGER | Diferencia horaria (en minutos) con respecto a la Hora Coordinada Universal (UCT). Por ejemplo, -300 es EST. |
| INSTANCENAME | VARCHAR(128) | Nombre de la instancia en la que se ha creado el mensaje. |
| DBPARTITIONNUM | SMALLINT | Número de partición en la que se ha creado el mensaje. Para una base de datos no particionada, se devuelve 0. |
| LEVEL | CHAR(1) | Nivel de gravedad del registro. Los valores posibles son: <ul style="list-style-type: none"> • 'C': Crítico • 'E': Error • 'I': Informativo • 'S': Grave • 'W': Aviso |
| IMPACT | VARCHAR(18) | Califica el impacto de este mensaje desde la perspectiva de un usuario. Esto clarifica el impacto del mensaje en el proceso empresarial del que forma parte DB2. Los valores posibles son: <ul style="list-style-type: none"> • 'CRITICAL' • 'IMMEDIATE' • 'NONE' • 'POTENTIAL' • 'UNLIKELY' |
| DBNAME | VARCHAR(128) | Nombre de la base de datos a la que se accede mientras se crea este mensaje. |
| EDU_ID | BIGINT | Identificador de Engine Dispatched Unit que ha creado este mensaje. |
| EDUNAME | VARCHAR(64) | Nombre de Engine Dispatched Unit que ha creado este mensaje. |
| PID | BIGINT | Identificador de proceso del sistema operativo que ha creado este mensaje. |

Tabla 224. Información devuelta por la función de tabla PD_GET_DIAG_HIST (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| PROCESS_NAME | VARCHAR(255) | Nombre del proceso del sistema operativo que ha creado este mensaje. |
| TID | BIGINT | Identificador numérico de la hebra que ha creado este mensaje. |
| APPLNAME | VARCHAR(255) | Nombre de la aplicación cliente que ha iniciado la conexión, si está disponible. |
| APPL_ID | VARCHAR(64) | Identificador de aplicación que ha iniciado la conexión, si está disponible. Por ejemplo: 'G91A3955.F33A.02DD18143340' |
| APPLHANDLE | VARCHAR(9) | Identificador exclusivo en todo el sistema correspondiente a la aplicación que ha iniciado la conexión, cuando está disponible. Es sinónimo de ID de agente. El identificador consiste en el número de partición de coordinación y un recuento de 16 bits separado por un '-'. El formato es el siguiente: 'nnn-xxxxx' |
| AUTH_ID | VARCHAR(30) | Identificador de autorización del sistema del proceso. |
| PRODUCT | VARCHAR(50) | Nombre del producto que ha creado el mensaje. Por ejemplo, 'DB2 Common'. |
| COMPONENT | VARCHAR(255) | Nombre del componente que ha creado el mensaje. |
| FUNCTION | VARCHAR(255) | Nombre de la función que ha generado el mensaje. |
| PROBE | INTEGER | Número de punto de prueba utilizado para identificar dónde se ha generado el mensaje en la función. |
| CALLEDPRODUCT | VARCHAR(50) | Nombre del producto en el origen del error. Se utiliza cuando el origen de un error no está en el mismo punto en el que se ha creado el mensaje. |
| CALLEDCOMPONENT | VARCHAR(255) | Nombre del componente en el origen del error. Se utiliza cuando el origen de un error no está en el mismo punto en el que se ha creado el mensaje. |
| CALLEDFUNCTION | VARCHAR(255) | Nombre de la función en el origen del error. Se utiliza cuando el origen de un error no está en el mismo punto en el que se ha creado el mensaje. |
| OSERR | INTEGER | Número de error del sistema operativo. |
| RETCODE | INTEGER | Código de retorno específico del producto. |
| MSGNUM | INTEGER | Número de mensaje numérico correspondiente al mensaje asociado, si está disponible. Por ejemplo, es la parte numérica de ADM7513W. |
| MSGTYPE | CHAR(3) | Tipo relacionado con el identificador del mensaje, si está disponible. Por ejemplo, ADM se utiliza para mensajes de registro de notificación de administración. |
| MSG | CLOB(16KB) | Texto de descripción breve de este registro. Es el texto del mensaje traducido correspondiente a MSGNUM y MSGTYPE para mensajes traducidos. Para mensajes no traducidos, es la descripción breve. Por ejemplo: 'Bringing down all db2fmp processes as part of db2stop'. |

Tabla 224. Información devuelta por la función de tabla PD_GET_DIAG_HIST (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| OBJTYPE | VARCHAR(64) | Tipo de objeto al que se aplica el suceso, si está disponible. Los valores posibles son: <ul style="list-style-type: none"> • 'APM' • 'CATALOG CACHE ENTRY' • 'CFG' • 'CLI' • 'CLP' • 'CONTAINER' • 'COUNTER' • 'DAS' • 'DB2AGENT' • 'DB PART MAP ID' • 'DB PART NUM' • 'DBA' • 'DBM' • 'DMS' • 'DPS' • 'EDU' • 'EVALUATION' • 'EXTENDER' • 'FCM' • 'HISTOGRAM TEMPLATE' • 'INDEX STATS' • 'INITIAL SAMPLING' • 'REDIST DB PART GROUP' • 'REDIST TABLE' • 'RDS' • 'SAMPLING TEST' • 'SERVICE CLASS' • 'STATS' • 'STATS DAEMON' • 'TABLE' • 'TABLE STATS' • 'TABLE AND INDEX STATS' • 'THRESHOLD' • 'UDF' • 'WORK ACTION SET' • 'WORK CLASS SET' • 'WORKLOAD' |
| OBJNAME | VARCHAR(255) | Nombre del objeto con el que está relacionado el suceso, si está disponible. |
| OBJNAME_QUALIFIER | VARCHAR(255) | Información adicional sobre el objeto, si está disponible. |

Tabla 224. Información devuelta por la función de tabla PD_GET_DIAG_HIST (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| EVENTTYPE | VARCHAR(24) | <p>El tipo de suceso es la acción o verbo asociados a este suceso. Los valores posibles son:</p> <ul style="list-style-type: none"> • 'ACCEPT' • 'ACCESS' • 'ADD' • 'ALTER' • 'ASSOCIATE' • 'AVAILABLE' • 'BRINGDOWN' • 'CHANGE' • 'CHANGECFG' • 'CLOSE' • 'COLLECT' • 'CONNECT' • 'CREATE' • 'DEPENDENCY' • 'DESTROY' • 'DISASSOCIATE' • 'DISCONNECT' • 'DISPATCH' • 'DROP' • 'FINI' • 'FREE' • 'GET' • 'INIT' • 'INTERRUPT' • 'OPEN', 'READ' • 'RECV' • 'REPLY' • 'REPORT' • 'REQUEST' • 'RESET' • 'SEND' • 'START' • 'STARTUP' • 'STOP' • 'SWITCH' • 'TERMINATE' • 'TRANSFER' • 'WAIT' • 'WORK' • 'WRITE' |
| EVENTDESC | VARCHAR(256) | Descripción breve de los campos clave de este suceso. |

Tabla 224. Información devuelta por la función de tabla PD_GET_DIAG_HIST (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|---------------------------|---------------|--|
| FIRST_EVENTQUALIFIERTYPE | VARCHAR(64) | Tipo del primer calificador de suceso. Los calificadores de suceso se utilizan para describir lo que se ha visto afectado por el suceso. Los valores posibles son: <ul style="list-style-type: none"> • 'AT' • 'BY' • 'CONTEXT' • 'DUE TO' • 'FOR' • 'FROM' • 'ON' • 'TO' Si <i>recurso</i> es OPTSTATS, el único valor posible es 'AT'. |
| FIRST_EVENTQUALIFIER | CLOB(16K) | Primer calificador del suceso. Si <i>recurso</i> es OPTSTATS, será la indicación de fecha y hora que indica cuándo se ha producido la recopilación de estadísticas. |
| SECOND_EVENTQUALIFIERTYPE | VARCHAR(64) | Tipo del segundo calificador de suceso. Si <i>recurso</i> es OPTSTATS, el valor es 'BY'. |
| SECOND_EVENTQUALIFIER | CLOB(16K) | Segundo calificador del suceso. Si <i>recurso</i> es OPTSTATS, los valores posibles son: <ul style="list-style-type: none"> • Asíncrono • FABRICATE • FABRICATE PARTIAL • SYNCHRONOUS • SYNCHRONOUS SAMPLED • USER |
| THIRD_EVENTQUALIFIERTYPE | VARCHAR(64) | Tipo del calificador de suceso. Si <i>recurso</i> es OPTSTATS, el valor es 'DUE TO'. |
| THIRD_EVENTQUALIFIER | CLOB(16K) | Tercer calificador del suceso. Si <i>recurso</i> es OPTSTATS, los valores posibles son: <ul style="list-style-type: none"> • Conflicto • Error • Objeto no disponible • Error de RUNSTATS • Tiempo de espera excedido |
| EVENTSTATE | VARCHAR(255) | Estado del objeto o acción como resultado del suceso. También puede contener un porcentaje que indica la progresión del suceso. |

Tabla 224. Información devuelta por la función de tabla PD_GET_DIAG_HIST (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| EVENTATTRIBUTE | VARCHAR(255) | Atributos del suceso. Es una lista de atributos asociados al suceso. Cuando se utiliza más de un atributo, la lista se separa mediante caracteres '+'. Por ejemplo 'CACHED + LOGICAL + AUTO'. Los valores posibles son: <ul style="list-style-type: none"> • 'ASYNc' • 'AUTO' • 'CACHED' • 'DIRECT' • 'EXTERNAL' • 'INDIRECT' • 'INTERNAL' • 'LOGICAL' • 'PERMANENT' • 'PHYSICAL' • 'SYNC' • 'TEMPORARY' |
| EVENTSTACK | CLOB(16K) | Pila de sucesos lógicos en el punto en que se anotó el registro, cuando está disponible. |
| CALLSTACK | CLOB(16K) | Vuelco de la pila del sistema operativo para la hebra que ha generado este registro, cuando está disponible. |
| DUMPFILe | CLOB(5000) | Nombre del archivo de vuelco secundario asociado al registro cronológico, cuando está disponible. Es una vía de acceso completa a un archivo o directorio del que se puede recuperar información adicional asociada al mensaje. |
| FULLREC | CLOB(16K) | Versión de texto formateado del registro completo. Esta sección también contiene campos de datos adicionales. |

Vista administrativa PDLOGMSGs_LAST24HOURS y función de tabla PD_GET_LOG_MSGS – Recuperar mensajes de determinación de problemas

La vista administrativa PDLOGMSGs_LAST24HOURS y la tabla de función PD_GET_LOG_MSGS devuelven mensajes de anotación cronológica de determinación de problemas que se han registrado en la anotación cronológica de notificaciones de DB2. La información está destinada a los administradores de bases de datos y del sistema.

Vista administrativa PDLOGMSGs_LAST24HOURS

La vista administrativa PDLOGMSGs_LAST24HOURS devuelve mensajes de anotación cronológica de determinación de problemas que se han registrado en la anotación cronológica de notificaciones de DB2 durante las últimas 24 horas.

El esquema es SYSIBMADM.

Consulte la Tabla 225 en la página 981 para ver una lista completa de la información que se puede devolver.


```

...TID APPL_ID                                COMPONENT                                ...
-----
... 1 9.26.15.148.36942.051124025612 oper system services ...
...
...
...
...
...
...
... 1 9.26.15.148.36942.051124025612 base sys utilities ...
... 1 9.26.15.148.36942.051124025612 relation data serv ...
...
...
...
... 1 9.26.15.148.36942.051124025612 relation data serv ...
...
...

```

Datos de salida de esta consulta (continuación).

```

... FUNCTION          PROBE  MSGNUM    MSGTYPE ...
-----
... sqloSleepInstance    38      504 ADM    ...
...
...
...
...
...
...
... sqlMarkDBad          10      7518 ADM   ...
... sqlrr_dump_ffdc     10        1 ADM    ...
...
...
...
... sqlrr_dump_ffdc     10        1 ADM    ...
...

```

Datos de salida de esta consulta (continuación).

```

... MSGSEVERITY MSG
-----
... C          ADM0504C Se ha producido un error
...           de proceso interno inesperado. ALL
...           SE HAN CONCLUIDO TODOS LOS PROCESOS DE DB2
...           ASOCIADOS A ESTA INSTANCIA.
...           La información de diagnóstico se ha
...           registrado. Póngase en contacto con el soporte de IBM
...           para obtener ayuda adicional.
... C          ADM7518C "CAPTAIN " está marcada como incorrecta.
... C          ADM0001C Se ha producido un error grave.
...           Examine el registro de notificaciones de
...           administración y consulte al Centro de soporte
...           de IBM, si es necesario.
... C          ADM0001C Se ha producido un error grave.
...           Examine el registro de notificaciones de
...           administración y consulte al Centro de soporte de IBM, si es necesario.

```

PD_GET_LOG_MSGS, función de tabla

La función de tabla PD_GET_LOG_MSGS devuelve la misma información que la vista administrativa PDLOGMSGs_LAST24HOURS, pero le permite especificar un periodo de tiempo específico que no está limitado a las últimas 24 horas.

Consulte la Tabla 225 en la página 981 para ver una lista completa de la información que se puede devolver.

Sintaxis

► PD_GET_LOG_MSGS(—indicaciónhora_más_antigua—) ◀

El esquema es SYSPROC.

Parámetro de la función de tabla

indicaciónhora_más_antigua

Es un argumento de entrada de tipo `TIMESTAMP` que especifica una indicación de fecha y hora válida. Las entradas se devuelven comenzando por la indicación de fecha y hora más reciente y terminando por la entrada de la anotación cronológica con la indicación de fecha y hora especificada por este argumento de entrada. Si se especifica un valor nulo, se devuelven todas las entradas de la anotación cronológica.

Autorización

Privilegio `EXECUTE` sobre la función de tabla `PD_GET_LOG_MSGS`.

Ejemplos

Ejemplo 1: Recuperar todos los mensajes de notificación registrados para la base de datos `SAMPLE` en la instancia de `DB2` durante la última semana para todas las particiones de base de datos. Mostrar los mensajes en orden cronológico.

```
SELECT TIMESTAMP, APPL_ID, DBPARTITIONNUM, MSG
FROM TABLE ( PD_GET_LOG_MSGS( CURRENT_TIMESTAMP - 7 DAYS)) AS T
WHERE INSTANCENAME = 'DB2' AND DBNAME = 'SAMPLE'
ORDER BY TIMESTAMP ASC
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| TIMESTAMP | APPL_ID | DBPARTITIONNUM | ... |
|----------------------------|-------------------------|----------------|-----|
| 2005-11-13-12.51.37.772000 | *LOCAL.DB2.050324175005 | 0 | ... |
| 2005-11-13-12.51.37.772001 | *LOCAL.DB2.050324175005 | 0 | ... |
| 2005-11-13-12.51.37.781000 | *LOCAL.DB2.050324175005 | 0 | ... |
| 2005-11-13-12.51.37.781001 | *LOCAL.DB2.050324175005 | 0 | ... |
| 2005-11-17-14.12.39.036001 | *LOCAL.DB2.041117191249 | 0 | ... |
| 2005-11-17-14.12.39.056000 | *LOCAL.DB2.041117191249 | 0 | ... |
| 2005-11-17-14.13.04.450000 | *LOCAL.DB2.041117191307 | 0 | ... |
| 2005-11-17-14.13.04.460000 | *LOCAL.DB2.041117191307 | 0 | ... |
| 2005-11-17-14.18.29.042000 | *LOCAL.DB2.041117190824 | 0 | ... |
| ... | | | |
| ... | | | |
| ... | | | |

Datos de salida de esta consulta (continuación).

```
... MSG
... -----
... ADM5502W El reajuste de "143" bloqueos en la tabla
... "SYSIBM .SYSINDEXAUTH" respecto al intento de bloqueo
... "X" ha sido satisfactorio.
... ADM5502W El reajuste de "144" bloqueos en la tabla
... "SYSIBM .SYSINDEXES" respecto al intento de bloqueo
... "X" ha sido satisfactorio.
```

```

... ADM5502W El reajuste de "416" bloqueos en la tabla
... "SYSIBM .SYSINDEXCOLUSE" respecto al intento de bloqueo "X"
... ha sido satisfactorio.
... ADM5500W DB2 está realizando un reajuste de bloqueos. El número
... total de bloqueos retenidos actualmente es "1129", y el número
... de bloqueos a retener de destino es de "564".
... ADM7506W Se ha solicitado la inmovilización de la base de datos.
... ADM7507W La petición de inmovilización de la base de datos se ha
... completado satisfactoriamente.
... ADM7510W Se ha solicitado la inmovilización de la base de datos.
... ADM7509W La petición de inmovilización de la base de datos se ha
... completado satisfactoriamente.
... ADM4500W Se ha producido una condición de
... desbordamiento de antememoria de paquete. No existe ningún error,
... pero esto indica que la antememoria del paquete ha sobrepasado
... el tamaño máximo configurado. Si esta condición persiste,
... tal vez desee ajustar el parámetro de configuración PCKCACHESZ DB.

```

Ejemplo 2: Recuperar todos los errores críticos registrados en la instancia de DB2 para la partición de base de datos 0 durante el último día, clasificados por el más reciente.

```

SELECT TIMESTAMP, DBNAME, MSG
  FROM TABLE (PD_GET_LOG_MSGS(CURRENT_TIMESTAMP - 1 DAYS)) AS T
 WHERE MSGSEVERITY = 'C' AND INSTANCENAME = 'DB2' AND DBPARTITIONNUM = 0
 ORDER BY TIMESTAMP DESC

```

El ejemplo siguiente muestra los datos de salida de la consulta.

| TIMESTAMP | DBNAME | MSG |
|----------------------------|----------|---|
| 2004-11-04-13.49.17.022000 | TESTSBCS | ADM0503C Se ha producido un error de procesamiento interno. TODOS LOS PROCESOS DB2 RELACIONADOS CON ESTA INSTANCIA SE HAN CONCLUIDO. Se ha registrado información de diagnóstico. Consulte al Centro de Soporte de IBM para obtener asistencia. |
| 2004-11-04-11.32.26.760000 | SAMPLE | ADM0503C Se ha producido un error de procesamiento interno. TODOS LOS PROCESOS DB2 RELACIONADOS CON ESTA INSTANCIA SE HAN CONCLUIDO. Se ha registrado información de diagnóstico. Consulte al Centro de Soporte de IBM para obtener asistencia. |

2 registro(s) seleccionado(s).

Ejemplo 3: Recuperar los mensajes grabados por la aplicación de servicios de procesos de DB2 con el ID de aplicación *LOCAL.DB2.050927195337, durante el último día.

```

SELECT TIMESTAMP, MSG
  FROM TABLE (PD_GET_LOG_MSGS(CURRENT_TIMESTAMP - 1 DAYS)) AS T
 WHERE APPL_ID = '*LOCAL.DB2.050927195337'

```

El ejemplo siguiente muestra los datos de salida de la consulta.

| TIMESTAMP | MSG |
|-----------|-------|
| ----- | ----- |

```

2005-06-27-21.17.12.389000 ADM4500W Se ha producido una condición de
desbordamiento de paquete de antememoria. No existe ningún error
pero esto indica que la antememoria del paquete
ha sobrepasado el tamaño máximo
configurado. Si esta condición persiste,
tal vez desee ajustar el parámetro de configuración
PCKCACHESZ DB.
2005-06-27-18.41.22.248000 ADM4500W Se ha producido una condición de
desbordamiento de paquete de antememoria. No existe ningún error
pero esto indica que la antememoria del paquete
ha sobrepasado el tamaño máximo
configurado. Si esta condición persiste,
tal vez desee ajustar el parámetro de configuración
PCKCACHESZ DB.
2005-06-27-12.51.37.772001 ADM5502W El reajuste de "143" bloqueos
en la tabla "SYSIBM .SYSINDEXAUTH"
respecto al intento de bloqueo "X" ha
sido satisfactorio.
2005-06-27-12.51.37.772000 ADM5502W El reajuste de "144" bloqueos
en la tabla "SYSIBM .SYSINDEXES" respecto
al intento de bloqueo "X" ha sido satisfactorio.
2005-06-27-12.51.37.761001 ADM5502W El reajuste de "416" bloqueos
en la tabla "SYSIBM .SYSINDEXCOLUSE"
respecto al intento de bloqueo "X" ha
sido satisfactorio.
...

```

Ejemplo 4: Encontrar todas las instancias del mensaje ADM0504C en la anotación cronológica de notificaciones. Observe que los mensajes que se tienen en cuenta no están limitados por una indicación de fecha y hora. Podría resultar una operación cara si el archivo de anotación cronológica de notificación es muy grande.

```

SELECT TIMESTAMP, DBPARTITIONNUM, DBNAME, MSG
FROM TABLE (PD_GET_LOG_MSGS(CAST(NULL AS TIMESTAMP))) AS T
WHERE MSGNUM = 504 AND MSGTYPE = 'ADM' AND MSGSEVERITY = 'C'

```

El ejemplo siguiente muestra los datos de salida de la consulta.

```

TIMESTAMP                DBPARTITIONNUM DBNAME      ...
-----
2005-11-23-21.56.41.240066          0 CAPTAIN    ...
...
...
...
...
...
...
...
...
...
...

```

Datos de salida de esta consulta (continuación).

```

... APPL_ID                MSG
... -----
... 9.26.15.148.36942.051124025612 ADM0504C Se ha producido
... un error de procesamiento
... interno. TODOS LOS PROCESOS
... de DB2 QUE ESTÁN RELACIONADOS CON ESTA
... INSTANCIA HAN QUEDADO
... SUSPENDIDOS. Se ha registrado
... información de
... diagnóstico. Consulte al
... Centro de Soporte de IBM
... para obtener asistencia.

```


Información devuelta

Nota: en un entorno de bases de datos particionadas, no está garantizado el orden en el que se devuelven los mensajes de la anotación cronológica. Si el orden de los registros de la anotación cronológica es importante, los resultados se deben clasificar por indicación de fecha y hora.

Tabla 225. Información que devuelve la vista administrativa PDLOGMSGs_LAST24HOURS y la función de tabla PD_GET_LOG_MSGS

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| TIMESTAMP | TIMESTAMP | La hora a la que se ha anotado la entrada. |
| TIMEZONE | INTEGER | Diferencia horaria (en minutos) con respecto a la Hora Coordinada Universal (UCT). Por ejemplo, -300 es EST. |
| INSTANCENAME | VARCHAR(128) | Nombre de la instancia que ha generado el mensaje. |
| DBPARTITIONNUM | SMALLINT | La partición de base de datos que ha generado el mensaje. Para un entorno que no sea de bases de datos particionadas, se devuelve 0. |
| DBNAME | VARCHAR(128) | La base de datos en la que se ha producido el error o el suceso. |
| PID | BIGINT | ID de proceso del proceso que ha generado el mensaje. |
| PROCESSNAME | VARCHAR(255) | Nombre del proceso que ha generado el mensaje. |
| TID | BIGINT | ID de la hebra del proceso que ha generado el mensaje. |
| APPL_ID | VARCHAR(64) | ID de la aplicación para la que está trabajando el proceso. |
| COMPONENT | VARCHAR(255) | El nombre del componente de DB2 que proporciona el mensaje. Para mensajes escritos por aplicaciones que utilizan la API db2AdminMsgWrite, se devuelve "Aplicación de usuario". |
| FUNCTION | VARCHAR(255) | El nombre de la función de DB2 que proporciona el mensaje. Para mensajes escritos por aplicaciones que utilizan la API db2AdminMsgWrite, se devuelve "Función de usuario". |

Tabla 225. Información que devuelve la vista administrativa PDLOGMSGS_LAST24HOURS y la función de tabla PD_GET_LOG_MSGS (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| PROBE | INTEGER | Identificador interno exclusivo que permite al Centro de desarrollo y de soporte al cliente DB2 localizar el punto, en el código fuente de DB2 que ha generado el mensaje. |
| MSGNUM | INTEGER | El número de mensaje correspondiente al error o al suceso. |
| MSGTYPE | CHAR(3) | Indica el tipo de mensaje: ADM (para mensajes escritos en la anotación cronológica de notificaciones de administración) o nulo si no se puede determinar el tipo de mensaje. |
| MSGSEVERITY | CHAR(1) | Gravedad del mensaje: C (grave), E (error), W (aviso), I (informativo) o nulo (si no se puede determinar la gravedad del mensaje). |
| MSG | CLOB(16 KB) | Texto del mensaje de la anotación cronológica de notificaciones. |

Procedimiento REORGCHK_IX_STATS - Recuperar estadísticas de índice para evaluación de la reorganización

El procedimiento REORGCHK_IX_STATS devuelve un conjunto de resultados que contiene estadísticas de índice que indican si existe o no necesidad de reorganización.

Sintaxis

►► REORGCHK_IX_STATS (—*ámbito*—, —*criterios*—) ◀◀

El esquema es SYSPROC.

Parámetros del procedimiento

ámbito

Argumento de entrada de tipo CHAR(1) que especifica el ámbito de las tablas que se deben evaluar, utilizando uno de los valores siguientes:

'T' Tabla

'S' Esquema

criterios

Argumento de entrada de tipo VARCHAR(259). Si *ámbito* tiene el valor 'T',

especifica un nombre de tabla completamente calificado o acepta uno de los valores siguientes: ALL, USER o SYSTEM. Si *ámbito* tiene el valor 'S', especifica un nombre de esquema.

Autorización

- Privilegio SELECT en las tablas de catálogos.
- Privilegio EXECUTE para el procedimiento REORGCHK_IX_STATS.

Ejemplo

```
CALL SYSPROC.REORGCHK_IX_STATS('T','JESCOTT.EMPLOYEE')
```

Nota sobre uso

El procedimiento utiliza el espacio de tablas SYSTOOLSTMPSPACE. Si aún no existe SYSTOOLSTMPSPACE, el procedimiento creará este espacio de tablas.

Información devuelta

Tabla 226. Información devuelta por el procedimiento REORGCHK_IX_STATS

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| TABLE_SCHEMA | VARCHAR(128) | Nombre de esquema. |
| TABLE_NAME | VARCHAR(128) | Nombre de tabla. |
| DATAPARTITIONNAME | VARCHAR(128) | Nombre de la partición de datos. NULL para las tablas no particionadas. |
| INDEX_SCHEMA | VARCHAR(128) | Nombre del esquema de índice. |
| INDEX_NAME | VARCHAR(128) | Nombre del índice. |
| INDCARD | BIGINT | Número de entradas en el índice. Éste puede ser diferente de la cardinalidad de tablas de algunos índices. Por ejemplo, la cardinalidad del índice en columnas XML puede ser superior a la cardinalidad de la tabla. |
| NLEAF | BIGINT | Número total de páginas hojas de índice. |
| NUM_EMPTY_LEAFS | BIGINT | Número de páginas hojas de índice pseudovacías. |
| NLEVELS | INTEGER | Número de niveles de índice. |
| NUMRIDS_DELETED | BIGINT | Número de RID pseudosuprimidos. |
| FULLKEYCARD | BIGINT | Número de entradas de índice exclusivas que no están marcadas como suprimidas. |
| LEAF_RECSIZE | BIGINT | Tamaño de registro de la entrada de índice de una página hoja. Es el tamaño promedio de la entrada de índice excluyendo cualquier actividad general y se calcula a partir del promedio de longitud de columna de todas las columnas que participan en el índice. |

Tabla 226. Información devuelta por el procedimiento REORGCHK_IX_STATS (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-----------------------|---------------|---|
| NONLEAF_RECSIZE | BIGINT | Tamaño de registro de la entrada de índice de una página no hoja. Es el tamaño promedio de la entrada de índice excluyendo cualquier actividad general y se calcula a partir del promedio de longitud de columna de todas las columnas que participan en el índice, excepto las columnas INCLUDE. |
| LEAF_PAGE_OVERHEAD | BIGINT | Espacio reservado en la página hoja de índice para uso interno. |
| NONLEAF_PAGE_OVERHEAD | BIGINT | Espacio reservado en la página no hoja de índice para uso interno use |
| PCT_PAGES_SAVED | SMALLINT | Porcentaje de páginas guardadas utilizando la compresión de índice. Un número distinto de cero indica que el índice está comprimido. |
| F4 | INTEGER | Valor de fórmula F4. |
| F5 | INTEGER | Valor de fórmula F5. |
| F6 | INTEGER | Valor de fórmula F6. |
| F7 | INTEGER | Valor de fórmula F7. |
| F8 | INTEGER | Valor de fórmula F8. |
| REORG | CHAR(5) | Campo con 5 caracteres, cada uno de los cuales se correlaciona con una de las cinco fórmulas: F4, F5, F6, F7 y F8; un guión significa que el valor de la fórmula se encuentra en el rango recomendado; un asterisco significa que el valor de la fórmula está fuera del rango recomendado, e indica que es necesaria la reorganización. |

Procedimiento REORGCHK_TB_STATS - Recuperar estadísticas de tabla para evaluación de la reorganización

El procedimiento REORGCHK_TB_STATS devuelve un conjunto de resultados que contiene estadísticas de tabla que indican si existe o no necesidad de reorganización.

Sintaxis

►►—REORGCHK_TB_STATS—(—*ámbito*—,—*criterios*—)————►►

El esquema es SYSPROC.

Parámetros del procedimiento

ámbito

Argumento de entrada de tipo CHAR(1) que especifica el ámbito de las tablas que se deben evaluar, utilizando uno de los valores siguientes:

'T' Tabla

'S' Esquema

criterios

Argumento de entrada de tipo VARCHAR(259). Si *ámbito* tiene el valor 'T', especifica un nombre de tabla completamente calificado o acepta uno de los valores siguientes: ALL, USER o SYSTEM. Si *ámbito* tiene el valor 'S', especifica un nombre de esquema.

Autorización

- Privilegio SELECT en las tablas de catálogos.
- Privilegio EXECUTE para el procedimiento REORGCHK_TB_STATS.

Ejemplo

```
CALL SYSPROC.REORGCHK_TB_STATS('T','JESCOTT.EMPLOYEE')
```

Nota sobre uso

El procedimiento utiliza el espacio de tablas SYSTOOLSTMPSPACE. Si aún no existe SYSTOOLSTMPSPACE, el procedimiento creará este espacio de tablas.

Información devuelta

Tabla 227. Información devuelta por el procedimiento REORGCHK_TB_STATS

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| TABLE_SCHEMA | VARCHAR(128) | Nombre de esquema. |
| TABLE_NAME | VARCHAR(128) | Nombre de tabla. |
| DATAPARTITIONNAME | VARCHAR(128) | Nombre de la partición de datos. NULL para las tablas no particionadas. |
| CARD | BIGINT | Cardinalidad (número de filas en la tabla). |
| OVERFLOW | BIGINT | Número de filas de desbordamiento. |
| NPAGES | BIGINT | Número total de páginas en las que existen filas de la tabla; -1 para una vista o alias, o si no se recopilan estadísticas; -2 para una subtabla o tabla de jerarquía. |
| FPAGES | BIGINT | Número total de páginas; -1 para una vista o alias, o si no se recopilan estadísticas; -2 para una subtabla o tabla de jerarquía. |

Tabla 227. Información devuelta por el procedimiento REORGCHK_TB_STATS (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| ACTIVE_BLOCKS | BIGINT | Número total de bloques activos de una tabla de clustering de múltiples dimensiones (MDC). Este campo sólo puede aplicarse a las tablas que se definen utilizando la cláusula ORGANIZE BY. Indica el número de bloques de la tabla que contiene datos. |
| TSIZE | BIGINT | Tamaño de la tabla. |
| F1 | INTEGER | Valor de fórmula F1. |
| F2 | INTEGER | Valor de fórmula F2. |
| F3 | INTEGER | Valor de fórmula F3. |
| REORG | CHAR(3) | Campo con 3 caracteres, cada uno de los cuales se correlaciona con una de las tres fórmulas: F1, F2 y F3; un guión significa que el valor de la fórmula se encuentra en el rango recomendado; un asterisco significa que el valor de la fórmula está fuera del rango recomendado, e indica que es necesaria la reorganización |

Funciones escalares SQLERRM – Recuperar información de mensajes de error

Hay dos versiones de la función escalar SQLERRM. La primera ofrece flexibilidad completa en la recuperación de mensajes, incluidos símbolos de mensajes y selección de idioma. La segunda toma únicamente un SQLCODE como parámetro de entrada y devuelve el mensaje abreviado en inglés.

SQLERRM, función escalar

La función escalar SQLERRM toma un identificador de mensaje, un entorno local y una entrada de símbolo y devuelve el mensaje abreviado o largo de tipo VARCHAR(32672) en el entorno local especificado. Si el entorno local de entrada no recibe soporte del servidor, el mensaje aparece en inglés.

Sintaxis

```

▶▶ SQLERRM (—idmsj—, —símbolos—, —delimitador_símbolo—, —entorno-local—, —msjabreviado—)

```

El esquema es SYSPROC.

Parámetros de la función escalar

idmsj

Argumento de entrada de tipo VARCHAR(9) que representa el número de mensaje para el que se debe recuperar información. El número de mensaje es el código de retorno de la aplicación, precedido por 'SQL', 'DBA' o 'CLI'. Por ejemplo, 'SQL551', 'CLI0001'. El número de mensaje también puede ser un SQLSTATE, como por ejemplo '42829'.

símbolos

Argumento de entrada de tipo VARCHAR(70) que representa la lista de símbolos del mensaje de error. Es posible que algunos mensajes no tengan símbolos. Si este parámetro es nulo, significa que no se produce ninguna sustitución de símbolo en el mensaje devuelto. La sustitución de símbolo sólo se produce cuando se devuelven mensajes abreviados por omisión. Si se selecciona la opción de mensaje largo, no se produce ninguna sustitución de símbolo.

delimitador_símbolo

Argumento de entrada de tipo VARCHAR(1) que representa el delimitador del símbolo. Este delimitador debe ser exclusivo y no debe estar contenido en ninguno de los símbolos que se pasan a la función escalar. Si no se proporciona ningún delimitador, el delimitador por omisión utilizado es el signo de punto y coma.

entorno-local

Argumento de entrada de tipo VARCHAR(33) que representa el entorno local que se debe pasar al servidor para que el mensaje de error se recupere en ese idioma. Si no se especifica ningún entorno local, o si el servidor no da soporte al entorno local, el mensaje se devuelve en inglés y se devuelve también un aviso.

msjabreviado

Argumento de entrada de tipo INTEGER que se utiliza para indicar si se debe recuperar el mensaje largo en lugar del mensaje abreviado por omisión. Para devolver mensajes largos, este valor debe estar establecido en 0 o en CAST(NULL as INTEGER).

Autorización

Privilegio EXECUTE sobre la función escalar SQLERRM

Ejemplos

Ejemplo 1: Recuperar el mensaje abreviado en inglés para SQL0551N con los símbolos "AYYANG", "UPDATE" y "SYSCAT.TABLES".

```
VALUES (SYSPROC.SQLERRM
        ('SQL551', 'AYYANG;UPDATE;SYSCAT.TABLES', ';', 'en_US', 1))
```

El ejemplo siguiente muestra los datos de salida.

```
1
-----...--
SQL0551N "AYYANG" no tiene el privilegio necesario para realizar la operación
"UPDATE" en el objeto "SYSCAT.TABLES"
```

Ejemplo 2: Recuperar el mensaje de error en inglés asociado a SQLSTATE 42501.

```
VALUES (SYSPROC.SQLERRM ('42501', '', '', 'en_US', 1))
```

El ejemplo siguiente muestra los datos de salida.

```
1
-----
SQLSTATE 42501: The authorization ID does not have the privilege to
perform the specified operation on the identified object.
```

Ejemplo 3: Recuperar el mensaje de error largo en inglés correspondiente a SQL1001N.

```
VALUES (SYSPROC.SQLERRM ('SQL1001', '', '', 'en_US', 0))
```

El ejemplo siguiente muestra los datos de salida.

```
1
-----
SQL1001N "<name>" no es un nombre de base de datos válido.
```

Explanation:

The syntax of the database name specified in the command is not valid. The database name must contain 1 to 8 characters and all the characters must be from the database manager base character set.

The command cannot be processed.

User Response:

Resubmit the command with the correct database name.

```
sqlcode : -1001
```

```
sqlstate : 2E000
```

SQLERRM, función escalar

Esta función escalar SQLERRM toma un SQLCODE como la única entrada y devuelve el mensaje abreviado de tipo VARCHAR(32672) correspondiente al SQLCODE especificado en inglés.

Sintaxis

```
►► SQLERRM (—sqlcode—) ◀◀
```

El esquema es SYSPROC.

Parámetro de la función escalar

sqlcode

Argumento de entrada de tipo INTEGER que representa un SQLCODE.

Autorización

Privilegio EXECUTE sobre la función escalar SQLERRM

Ejemplo

Recuperar el mensaje abreviado correspondiente al SQLCODE SQL0551N.

```
VALUES (SYSPROC.SQLERRM (551))
```

El ejemplo siguiente muestra los datos de salida.

SQL0551N "" no tiene el privilegio necesario para realizar una operación "" en el objeto "".

SYSINSTALOBJECTS

El procedimiento SYSINSTALOBJECTS crea o descarta los objetos de base de datos requeridos para una herramienta específica.

Sintaxis

```
►—SYSINSTALOBJECTS—(—nombre-herramienta—,—acción—,——————►
►—nombre-espacio-tablas—,—nombre-esquema—)—————►
```

El esquema es SYSPROC.

Parámetros del procedimiento

nombre-herramienta

Argumento de entrada de tipo VARCHAR(128) que especifica el nombre de la herramienta que se debe cargar, utilizando uno de los valores siguientes:

- 'AM' para crear objetos de supervisor de actividad
- 'DB2AC' para el cálculo autónomo (supervisor de salud)
- 'STMG_DBSIZE_INFO' para la gestión del almacenamiento
- 'OPT_PROFILES' para crear la tabla de perfil de optimización
- 'POLICY' para la política (tablas y activadores)
- 'EXPLAIN' para crear tablas de explicación

acción

Argumento de entrada de tipo CHAR(1) que especifica la acción que se debe emprender. Los valores válidos son:

- | | |
|---|--------------------|
| C | Crear objetos. |
| D | Descartar objetos. |
| V | Verificar objetos. |

nombre-espacio-tablas

Argumento de entrada de tipo VARCHAR(128) que especifica el nombre del espacio de tablas en el que se van a crear los objetos. Si no se ha especificado un valor o si éste es una serie vacía o en blanco, se utilizará el espacio por omisión del usuario si el nombre de la herramienta es AM. En caso contrario, se utilizará el espacio de la tabla SYSTOOLSPACE. Si no existe SYSTOOLSPACE, se creará.

nombre-esquema

Excepto para la opción de nombre-herramienta 'EXPLAIN', SYSTOOLS siempre se utiliza como esquema sin tener en cuenta el nombre-esquema que se pasa como parámetro de entrada.

Para la opción de nombre-herramienta 'EXPLAIN', puede pasarse un nombre-esquema de entrada y pueden crearse las tablas bajo el nombre-esquema especificado. Si no se pasa ningún nombre-esquema como parámetro de entrada, se utiliza el esquema SYSTOOLS.

Ejemplo

```
CALL SYSPROC.SYSINSTALLOBJECTS('AM', 'C', CAST (NULL AS VARCHAR(128)),  
    CAST (NULL AS VARCHAR(128)))
```

Capítulo 21. Rutinas administrativas de SQL obsoletas y sus vistas o rutinas de sustitución

Para ofrecer un mayor soporte en DB2 para Linux, UNIX y Windows Versión 9.7 para las rutinas administrativas existentes, algunas de las rutinas de DB2 Versión 9.5 se han sustituido por otras rutinas o vistas nuevas y más completas.

Las aplicaciones que utilizan las funciones de tabla DB2 para Linux, UNIX y Windows Versión 9.7 se deben modificar para que utilicen las funciones o vistas administrativas nuevas. Las funciones nuevas de la tabla tienen los mismos nombres base que las funciones originales pero tienen el sufijo '_Vxx' para indicar la versión del producto en que se añadieron (por ejemplo, _V97). En la mayoría de los casos, las vistas administrativas y las funciones de la tabla nueva devuelven información adicional. Las vistas administrativas siempre se basarán en la versión más actual de las funciones de tablas, y por consiguiente permitirán una mayor portabilidad de aplicaciones. Puesto que las columnas pueden variar de una versión a la siguiente (algunas se añaden y otras se borran), se recomienda que se seleccionen columnas específicas desde las vistas administrativas, o que se describa el conjunto de resultados si una aplicación utiliza una sentencia SELECT *.

Tabla 228. Rutinas administrativas de SQL obsoletas y sus vistas o rutinas de sustitución para DB2 Versión 9.7

| Función obsoleta en DB2 Versión 9.5 | Nueva función o vista en DB2 para Linux, UNIX y Windows Versión 9.7 |
|--|--|
| "Vista ADMIN_TAB_COMPRESS_INFO y ADMIN_GET_TAB_COMPRESS_INFO" en la página 1001 | "Vista administrativa ADMIN_TAB_COMPRESS_INFO y función de tabla ADMIN_GET_TAB_COMPRESS_INFO_V97 - Devolver información de compresión" en la página 246 |
| "Tabla de función ADMIN_GET_TAB_INFO - Recuperar información de tamaño y estado para las tablas" en la página 993 | "Vista administrativa ADMIN_TAB_INFO y función de tabla ADMIN_GET_TAB_INFO_V97 - Recuperar información de tamaño y estado" en la página 254 |
| "Vista administrativa SNAPSTORAGE_PATHS y función de tabla SNAP_GET_STORAGE_PATHS - Recuperar información de vía de acceso del almacenamiento automático" en la página 770 | "Vista administrativa SNAPSTORAGE_PATHS y función de tabla SNAP_GET_STORAGE_PATHS_V97 - Recuperar información de vía de acceso de almacenamiento automático" en la página 618 |
| "Vista administrativa SNAPTbsp_PART y función de tabla SNAP_GET_TBSP_PART_V91 - Recuperar información de instantáneas del grupo de datos lógicos tablespace_nodeinfo" en la página 797 | "Vista administrativa SNAPTbsp_PART y función de tabla SNAP_GET_TBSP_PART_V97 - Recuperar información de instantáneas del grupo de datos lógicos tablespace_nodeinfo" en la página 645 |
| "WLM_GET_ACTIVITY_DETAILS - Devolver información detallada sobre una actividad específica" en la página 1156 | "Función de tabla MON_GET_ACTIVITY_DETAILS - Obtener detalles completos sobre la actividad" en la página 370 |
| "WLM_GET_SERVICE_SUBCLASS_STATS - Devolver estadísticas de subclases de servicio" en la página 1173 | "Función de tabla WLM_GET_SERVICE_SUBCLASS_STATS_V97 - Devolver estadísticas de subclases de servicio" en la página 895 |
| "WLM_GET_WORKLOAD_STATS - Devolver estadísticas de carga de trabajo" en la página 1185 | "Función de tabla WLM_GET_WORKLOAD_STATS_V97 - Devolver estadísticas de carga de trabajo" en la página 913 |
| "WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES - Devolver una lista de actividades" en la página 1180 | "WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES_V97 - Devolver una lista de actividades" en la página 908 |

Tabla 228. Rutinas administrativas de SQL obsoletas y sus vistas o rutinas de sustitución para DB2 Versión 9.7 (continuación)

| Función obsoleta en DB2 Versión 9.5 | Nueva función o vista en DB2 para Linux, UNIX y Windows Versión 9.7 |
|---|--|
| "WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES - Lista de apariciones de carga de trabajo" en la página 1168 | "WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES_V97 - Lista de apariciones de carga de trabajo" en la página 890 |
| "WLM_GET_SERVICE_CLASS_AGENTS - Listar agentes en ejecución en una clase de servicio" en la página 1163 | "Función de tabla WLM_GET_SERVICE_CLASS_AGENTS_V97 - Listar los agentes que se ejecutan en una clase de servicio" en la página 883 |

El Supervisor de salud ha quedado en desuso en DB2 para Linux, UNIX y Windows Versión 9.7. Las interfaces del Supervisor de salud en desuso aún están soportadas en la Versión 9.7. Una nueva suite de herramientas de GUI para gestionar los datos y las aplicaciones orientadas a datos de DB2 para Linux, UNIX y Windows está disponibles y puede utilizarse en lugar de las herramientas del Centro de control. Para obtener más información, consulte Herramientas de desarrollo de aplicaciones y de gestión de bases de datos.

Tabla 229. Rutinas del Supervisor de salud en desuso

| |
|---|
| "EALTH_CONT_HI" en la página 1009 |
| "HEALTH_CONT_HI_HIS" en la página 1010 |
| "HEALTH_CONT_INFO" en la página 1012 |
| "HEALTH_DB_HI" en la página 1014 |
| "HEALTH_DB_HI_HIS" en la página 1017 |
| "HEALTH_DB_HIC" en la página 1021 |
| "HEALTH_DB_HIC_HIS" en la página 1023 |
| "HEALTH_DB_INFO" en la página 1026 |
| "HEALTH_DBM_HI" en la página 1027 |
| "HEALTH_DBM_HI_HIS" en la página 1029 |
| "HEALTH_DBM_INFO" en la página 1031 |
| "HEALTH_GET_ALERT_ACTION_CFG" en la página 1032 |
| "HEALTH_GET_ALERT_CFG" en la página 1036 |
| "HEALTH_GET_IND_DEFINITION" en la página 1039 |
| "RHEALTH_HI_REC" en la página 1041 |
| "HEALTH_TBS_HI" en la página 1043 |
| "HEALTH_TBS_HI_HIS" en la página 1045 |
| "HEALTH_TBS_INFO" en la página 1049 |

En el release anterior, DB2 Versión 9.5, también había funciones nuevas que sustituían a las funciones de DB2 Versión 9.1.

Tabla 230. Rutinas administrativas de SQL obsoletas y sus vistas o rutinas de sustitución para DB2 Versión 9.5

| Función obsoleta en DB2 Versión 9.1 | Función o vista nueva en DB2 Versión 9.5 |
|---|---|
| "Función de tabla SNAP_GET_APPL – Recuperar información de instantáneas del grupo de datos lógico appl" en la página 1051 | "Vista administrativa SNAPAPPL y función de tabla SNAP_GET_APPL_V95 – Recuperar información de instantáneas del grupo de datos lógicos appl" en la página 530 |

Tabla 230. Rutinas administrativas de SQL obsoletas y sus vistas o rutinas de sustitución para DB2 Versión 9.5 (continuación)

| Función obsoleta en DB2 Versión 9.1 | Función o vista nueva en DB2 Versión 9.5 |
|---|--|
| “Función de tabla SNAP_GET_APPL_INFO – Recuperación de información de instantáneas del grupo de datos lógico appl_info” en la página 1059 | “Vista administrativa SNAPAPPL_INFO y función de tabla SNAP_GET_APPL_INFO_V95 – Recuperar información de instantáneas del grupo de datos lógicos appl_info” en la página 523 |
| “Función de tabla SNAP_GET_BP – Recuperación de la información de instantáneas del grupo de datos lógico bufferpool” en la página 1065 | “Vista administrativa SNAPBP y función de tabla SNAP_GET_BP_V95 – Recuperar información de instantáneas del grupo lógico bufferpool” en la página 540 |
| “Función de tabla SNAP_GET_DB_V91 - Recuperar información de instantánea desde el grupo lógico dbase” en la página 1083 | “Vista administrativa SNAPDB y función de tabla SNAP_GET_DB_V95 – Recuperar información de instantánea desde el grupo lógico dbase” en la página 554 |
| “Función de tabla SNAP_GET_DBM – Recuperación de la información de instantáneas del grupo de datos lógicos dbm” en la página 1079 | “Vista administrativa SNAPDBM y función de tabla SNAP_GET_DBM_V95 – Recuperar la información de instantáneas del grupo de datos lógicos dbm” en la página 572 |
| “Función de tabla SNAP_GET_DYN_SQL_V91 - Recuperar información de la instantánea del grupo de datos lógicos dynsql” en la página 1095 | “Vista administrativa SNAPDYN_SQL y función de tabla SNAP_GET_DYN_SQL_V95 – Recuperar información de instantáneas del grupo de datos lógicos dynsql” en la página 583 |

Tabla de función ADMIN_GET_TAB_INFO - Recuperar información de tamaño y estado para las tablas

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la “Vista administrativa ADMINTABINFO y función de tabla ADMIN_GET_TAB_INFO_V97 – Recuperar información de tamaño y estado” en la página 254.

La función de tabla ADMIN_GET_TAB_INFO proporcionan métodos para recuperar la información de tamaño y estado de tabla que actualmente no está disponible en las vistas de catálogo.

En la tabla Metadatos de la función de tabla ADMIN_GET_TAB_INFO encontrará una lista completa de la información que puede devolverse.

Sintaxis

►►—ADMIN_GET_TAB_INFO—(—esquematabla—,—nombretabla—)—◄◄

El esquema es SYSPROC.

Parámetros de la función de tabla

esquematabla

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de esquema.

nombretabla

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de tabla, un nombre de tabla de consultas materializadas o un nombre de tabla de jerarquía.

Autorización

Privilegio EXECUTE para la función de tabla ADMIN_GET_TAB_INFO.

Ejemplo

Ejemplo 1: Recuperar información de tamaño y estado para la tabla DBUSER1.EMPLOYEE.

```
SELECT * FROM TABLE (SYSPROC.ADMIN_GET_TAB_INFO('DBUSER1', 'EMPLOYEE'))  
AS T
```

Ejemplo 2: Imaginemos que existe una tabla no particionada (DBUSER1.EMPLOYEE) cuyos objetos asociados (por ejemplo, los índices y los LOB) se almacenan en un único espacio de tablas. Calcule cuánto espacio físico del espacio de tablas utiliza la tabla:

```
SELECT (data_object_p_size + index_object_p_size + long_object_p_size +  
lob_object_p_size + xml_object_p_size) as total_p_size  
FROM TABLE( SYSPROC.ADMIN_GET_TAB_INFO( 'DBUSER1', 'EMPLOYEE' )) AS T
```

Calcular cuánto espacio se necesitaría si la tabla se moviera a otro espacio de tablas, donde el nuevo espacio de tablas tendría el mismo tamaño de página y tamaño de extensión que el espacio de tablas original:

```
SELECT (data_object_l_size + index_object_l_size + long_object_l_size +  
lob_object_l_size + xml_object_l_size) as total_l_size  
FROM TABLE( SYSPROC.ADMIN_GET_TAB_INFO( 'DBUSER1', 'EMPLOYEE' )) AS T
```

Notas de uso

- Si se especifican *esquematabla* y *nombretabla*, sólo se devuelve información para esa tabla específica.
- Si se especifica el *esquematabla* pero *nombretabla* está vacío (") o es NULL, se devolverá información para todas las tablas del esquema determinado.
- Si *esquematabla* está vacío (") o es NULL y se especifica *nombretabla*, se devolverá un error. Para recuperar información para una tabla determinada, tanto el esquema como el nombre de la tabla deberán identificar a la tabla.
- Si *esquematabla* y *nombretabla* están vacíos (") o son NULL, se devolverá información para todas las tablas.
- Si no existe *esquematabla* o *nombretabla*, o si *nombretabla* no corresponde a un nombre de tabla (tipo T), un nombre de tabla de consultas materializadas (tipo S) o un nombre de tabla de jerarquía (tipo H), se devuelve un conjunto de resultados vacío.
- Cuando la función de tabla ADMIN_GET_TAB_INFO recupere datos para una tabla determinada, necesitará un bloqueo compartido en la fila correspondiente de SYSTABLES para garantizar la coherencia de los datos que se devuelven (por ejemplo, para garantizar que la tabla no se descarte mientras se recupera información para ésta). El bloqueo sólo se mantendrá mientras tiene lugar la recuperación de la información de tamaño y de estado para la tabla, no mientras está activa la llamada de función de tabla.
- El tamaño físico que se informa para las tablas de los espacios de tablas SMS es igual al tamaño lógico.
- Cuando exista una reorganización in situ activa en una tabla, no se calculará el tamaño físico del objeto de datos (DATA_OBJECT_P_SIZE). Sólo se devolverá el tamaño lógico. Para determinar si una reorganización in situ está activa en la tabla, consulte la columna de salida INPLACE_REORG_STATUS.

- El tamaño lógico que se informa para los objetos LOB creados antes de la Versión 8 de DB2 UDB puede que sea superior al tamaño físico si los objetos todavía no se han reorganizado.

Metadatos de la función de tabla ADMIN_GET_TAB_INFO

Tabla 231. Metadatos de la función de tabla ADMIN_GET_TAB_INFO

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| TABSCHEMA | VARCHAR(128) | Nombre de esquema. |
| TABNAME | VARCHAR(128) | Nombre de tabla. |
| TABTYPE | CHAR(1) | Tipo de tabla: <ul style="list-style-type: none"> • 'H' = tabla de jerarquía • 'S' = tabla de consultas materializadas • 'T' = tabla |
| DBPARTITIONNUM | SMALLINT | Número de partición de base de datos. |
| DATA_PARTITION_ID | INTEGER | Número de partición de datos. |
| AVAILABLE | CHAR(1) | Estado de la tabla: <ul style="list-style-type: none"> • 'N' = la tabla no está disponible. Si la tabla no está disponible, todas las demás columnas de salida relacionadas con el tamaño y el estado serán NULL. • 'Y' = la tabla está disponible. <p>Nota: La realización de una recuperación en avance en una carga no recuperable establecerá una tabla en estado de no disponibilidad.</p> |

Tabla 231. Metadatos de la función de tabla ADMIN_GET_TAB_INFO (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|--------------------|---------------|---|
| DATA_OBJECT_L_SIZE | BIGINT | <p>Tamaño lógico del objeto de datos. Cantidad de espacio de disco que se ha asignado lógicamente para la tabla y que se informa en kilobytes. El tamaño lógico es la cantidad de espacio que la tabla reconoce. Puede que sea inferior a la cantidad de espacio que se ha asignado físicamente para la tabla (por ejemplo, en el caso de un truncamiento de tabla lógica). Para las tablas de clúster multidimensional (MDC), este tamaño incluye el tamaño lógico del objeto de correlación de bloques. El tamaño devuelto tiene en cuenta las extensiones completas que se han asignado lógicamente para la tabla y, para los objetos que se han creado en espacios de tablas DMS, incluye una estimación de las extensiones EMP (Extent Map Page). Este tamaño representa el tamaño lógico sólo de la tabla base. En otras columnas se informa del espacio que utilizan los datos LOB, los datos largos, los índices y los objetos XML.</p> |
| DATA_OBJECT_P_SIZE | BIGINT | <p>Tamaño físico del objeto de datos. Cantidad de espacio de disco que se ha asignado físicamente para la tabla y que se informa en kilobytes. Para las tablas MDC, este tamaño incluye el tamaño del objeto de correlación de bloques. El tamaño devuelto tiene en cuenta las extensiones completas que se han asignado para la tabla e incluye las extensiones EMP para los objetos que se han creado en los espacios de tablas DMS. Este tamaño representa el tamaño físico sólo de la tabla base. En otras columnas se informa del espacio que utilizan los datos LOB, los datos largos, los índices y los objetos XML.</p> |

Tabla 231. Metadatos de la función de tabla ADMIN_GET_TAB_INFO (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|---------------------|---------------|---|
| INDEX_OBJECT_L_SIZE | BIGINT | Tamaño lógico del objeto de índice. Cantidad de espacio de disco que se ha asignado lógicamente para los índices definidos en la tabla y que se informa en kilobytes. El tamaño lógico es la cantidad de espacio que la tabla reconoce. Puede que sea inferior a la cantidad de espacio que se ha asignado físicamente para contener los datos de índice para la tabla (por ejemplo, en el caso de un truncamiento de tabla lógica). El tamaño devuelto tiene en cuenta las extensiones completas que se han asignado lógicamente para los índices y, para los índices que se han creado en espacios de tablas DMS, incluye una estimación de las extensiones EMP. Este valor sólo se informa para las tablas no particionadas. Para las tablas particionadas, este valor será 0. |
| INDEX_OBJECT_P_SIZE | BIGINT | Tamaño físico del objeto de índice. Cantidad de espacio de disco que se ha asignado físicamente para los índices definidos en la tabla y que se informa en kilobytes. El tamaño devuelto tiene en cuenta las extensiones completas que se han asignado para los índices e incluye las extensiones EMP para los índices que se han creado en los espacios de tablas DMS. Este valor sólo se informa para las tablas no particionadas. Para las tablas particionadas, este valor será 0. |
| LONG_OBJECT_L_SIZE | BIGINT | Tamaño lógico del objeto largo. Cantidad de espacio de disco que se ha asignado lógicamente para los datos de campo largo en una tabla y que se informa en kilobytes. El tamaño lógico es la cantidad de espacio que la tabla reconoce. Puede que sea inferior a la cantidad de espacio que se ha asignado físicamente para contener los datos de campo largo para la tabla (por ejemplo, en el caso de un truncamiento de tabla lógica). El tamaño devuelto tiene en cuenta las extensiones completas que se han asignado lógicamente para los datos de campo largo y, para los datos de campo largo que se han creado en espacios de tablas DMS, incluye una estimación de las extensiones EMP. |

Tabla 231. Metadatos de la función de tabla ADMIN_GET_TAB_INFO (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|--------------------|---------------|--|
| LONG_OBJECT_P_SIZE | BIGINT | Tamaño físico del objeto largo. Cantidad de espacio de disco que se ha asignado físicamente para los datos de campo largo en una tabla y que se informa en kilobytes. El tamaño devuelto tiene en cuenta las extensiones completas que se han asignado para los datos de campo largo e incluye las extensiones EMP para los datos de campo largo que se han creado en los espacios de tablas DMS. |
| LOB_OBJECT_L_SIZE | BIGINT | Tamaño lógico del objeto LOB. Cantidad de espacio de disco que se ha asignado lógicamente para los datos LOB en una tabla y que se informa en kilobytes. El tamaño lógico es la cantidad de espacio que la tabla reconoce. Puede que sea inferior a la cantidad de espacio que se ha asignado físicamente para contener los datos LOB para la tabla (por ejemplo, en el caso de un truncamiento de tabla lógica). El tamaño incluye el espacio que se ha asignado lógicamente para el objeto de asignación de LOB. El tamaño devuelto tiene en cuenta las extensiones completas que se han asignado lógicamente para los datos LOB y, para los datos LOB que se han creado en espacios de tablas DMS, incluye una estimación de las extensiones EMP. |
| LOB_OBJECT_P_SIZE | BIGINT | Tamaño físico del objeto LOB. Cantidad de espacio de disco que se ha asignado físicamente para los datos LOB en una tabla y que se informa en kilobytes. El tamaño incluye el espacio que se ha asignado para el objeto de asignación de LOB. El tamaño devuelto tiene en cuenta las extensiones completas que se han asignado para los datos LOB e incluye las extensiones EMP para los datos LOB que se han creado en los espacios de tablas DMS. |

Tabla 231. Metadatos de la función de tabla ADMIN_GET_TAB_INFO (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| XML_OBJECT_L_SIZE | BIGINT | <p>Tamaño lógico del objeto XML. Cantidad de espacio de disco que se ha asignado lógicamente para los datos XML en una tabla y que se informa en kilobytes. El tamaño lógico es la cantidad de espacio que la tabla reconoce. Puede que sea inferior a la cantidad de espacio que se ha asignado físicamente para contener los datos XML para la tabla (por ejemplo, en el caso de un truncamiento de tabla lógica). El tamaño devuelto tiene en cuenta las extensiones completas que se han asignado lógicamente para los datos XML y, para los datos XML que se han creado en espacios de tablas DMS, incluye una estimación de las extensiones EMP.</p> |
| XML_OBJECT_P_SIZE | BIGINT | <p>El tamaño físico del objeto XML. Cantidad de espacio de disco que se ha asignado físicamente para los datos XML en una tabla y que se informa en kilobytes. El tamaño devuelto tiene en cuenta las extensiones completas que se han asignado para los datos XML e incluye las extensiones EMP para los datos XML que se han creado en los espacios de tablas DMS.</p> |
| INDEX_TYPE | SMALLINT | <p>Indica el tipo de índices que actualmente está utilizándose para la tabla. Devuelve:</p> <ul style="list-style-type: none"> • 1 si están utilizándose índices de tipo 1. • 2 si están utilizándose índices de tipo 2. |
| REORG_PENDING | CHAR(1) | <p>El valor 'Y' indica que se ha aplicado una modificación de reorganización recomendada a la tabla y que se necesita una reorganización clásica (fuera de línea). De lo contrario, se devuelve 'N'.</p> |

Tabla 231. Metadatos de la función de tabla ADMIN_GET_TAB_INFO (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|----------------------|---------------|--|
| INPLACE_REORG_STATUS | VARCHAR(10) | Estado actual de una reorganización de tabla in situ en la tabla. El valor de estado puede ser uno de los siguientes: <ul style="list-style-type: none"> • ABORTED (en estado PAUSED, pero incapaz de RESUME; se necesita STOP) • EXECUTING • NULL (si no se ha realizado ninguna reorganización in situ en la tabla) • PAUSED |
| LOAD_STATUS | VARCHAR(12) | Estado actual de una operación de carga para la tabla. El valor de estado puede ser uno de los siguientes: <ul style="list-style-type: none"> • IN_PROGRESS • NULL (si no existe ninguna carga en progreso para la tabla y si la tabla no está en estado de carga pendiente) • PENDING |
| READ_ACCESS_ONLY | CHAR(1) | 'Y' si la tabla se encuentra en estado de Acceso de sólo lectura; de lo contrario, 'N'. El valor 'N' no significa que la tabla ofrezca acceso completo. Si existe una carga en proceso o pendiente, el valor 'Y' significa que los datos de tabla están disponibles para el acceso de lectura y el valor 'N' significa que no puede accederse a la tabla. De forma similar, si el estado de la tabla corresponde a un establecimiento de integridad pendiente (vea la columna SYSCAT.TABLES STATUS), el valor 'N' significa que no puede accederse a la tabla. |
| NO_LOAD_RESTART | CHAR(1) | El valor 'Y' indica que la tabla se encuentra en un estado de carga parcial y que no será posible realizar un reinicio de carga. De lo contrario, se devuelve el valor 'N'. |
| NUM_REORG_REC_ALTERS | SMALLINT | Número de operaciones de modificación de reorganización recomendada (por ejemplo, operaciones de modificación después de las cuales se necesita una reorganización) que se han realizado para esta tabla desde la última reorganización. |

Tabla 231. Metadatos de la función de tabla ADMIN_GET_TAB_INFO (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------------|---------------|---|
| INDEXES_REQUIRE_REBUILD | CHAR(1) | 'Y' si debe volver a crearse cualquiera de los índices definidos en la tabla; de lo contrario, 'N'. Si no se ha definido ningún índice en la tabla, también se devolverá 'N', pues no existirá ningún índice que deba volver a crearse. |
| LARGE_RIDS | CHAR(1) | Indica si la tabla utiliza o no los ID de fila grandes (RID) (número de página de 4 bytes, número de ranura de 2 bytes). El valor 'Y' indica que la tabla utiliza RID grandes y el valor 'N' indica que no utiliza RID grandes. Se devolverá el valor 'P' (pendiente) si la tabla da soporte a los RID grandes (es decir, la tabla se encuentra en un espacio de tablas grande) pero, como mínimo, uno de los índices para la tabla no se ha reorganizado o no ha vuelto a crearse todavía, por lo que la tabla todavía utiliza RID de 4 bytes (lo que significa que debe realizarse una acción para convertir la tabla o los índices). |
| LARGE_SLOTS | CHAR(1) | Indica si la tabla utiliza o no ranuras grandes (lo que admite más de 255 filas por página). El valor 'Y' indica que la tabla utiliza ranuras grandes y el valor 'N' indica que no utiliza ranuras grandes. Se devolverá el valor 'P' (pendiente) si la tabla da soporte a las ranuras grandes (es decir, la tabla se encuentra en un espacio de tablas grande), pero todavía no se ha realizado en la tabla ninguna operación de truncamiento de tabla u operación de reorganización de tabla fuera de línea, por lo que todavía utiliza un máximo de 255 filas por página. |
| DICTIONARY_SIZE | BIGINT | Tamaño del diccionario, en bytes, utilizado para la compresión de filas, si existe un diccionario de compresión de filas para la tabla. |

Vista ADMINTABCOMPRESSINFO y ADMIN_GET_TAB_COMPRESS_INFO

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la "Vista administrativa ADMINTABCOMPRESSINFO y función de tabla ADMIN_GET_TAB_COMPRESS_INFO_V97 - Devolver información de compresión" en la página 246.

La vista administrativa ADMINTABCOMPRESSINFO y la tabla de función ADMIN_GET_TAB_COMPRESS_INFO devuelven información de compresión para tablas, tablas de consultas materializadas (MQT) y tablas de jerarquía.

Vista administrativa ADMINTABCOMPRESSINFO

La vista administrativa ADMINTABCOMPRESSINFO devuelve información de compresión para tablas, tablas de consultas materializadas (MQT) y tablas de jerarquía. En la vista de catálogo SYSCAT.TABLES, estos tipos de tablas se informan con T para tabla, S para tablas de consultas materializadas y H para tablas de jerarquía. Para una tabla, la información se devuelve en el nivel de partición de datos y en el nivel de partición de base de datos.

El esquema es SYSIBMADM.

En la tabla Vista administrativa ADMINTABCOMPRESSINFO y los metadatos de la función de tabla ADMIN_GET_TAB_COMPRESS_INFO encontrará una lista completa de la información que puede devolverse.

Autorización

Se necesita una de las autorizaciones siguientes:

- Privilegio SELECT en la vista administrativa ADMINTABCOMPRESSINFO
- Privilegio CONTROL en la vista administrativa ADMINTABCOMPRESSINFO
- Autorización DATAACCESS

Asimismo, también es necesario uno de los privilegios o autorizaciones siguientes:

- Privilegio EXECUTE en la función de tabla ADMIN_GET_TAB_COMPRESS_INFO
- Autorización DATAACCESS

Ejemplos

Ejemplo 1: Recuperar toda la información de compresión para todas las tablas

```
SELECT * FROM SYSIBMADM.ADMINTABCOMPRESSINFO
```

El ejemplo siguiente muestra los datos de salida de la consulta:

| TABSCHEMA | TABNAME | DBPARTITIONNUM | DATA_PARTITION_ID | COMPRESS_ATTR | DICT_BUILDER | DICT_BUILD_TIMESTAMP |
|-----------|------------|----------------|-------------------|---------------|--------------|----------------------------|
| | | | | | | |
| SYSIBM | SYSTABLES | 0 | 0 | N | NOT BUILT | - |
| SYSIBM | SYSCOLUMNS | 0 | 0 | N | NOT BUILT | - |
| ... | | | | | | |
| SIMAP2 | STAFF | 0 | 0 | Y | REORG | 2006-08-27-19.07.36.000000 |
| SIMAP2 | PARTTAB | 0 | 0 | Y | REORG | 2006-08-27-22.07.17.000000 |
| ... | | | | | | |

156 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación):

| COMPRESS_DICT_SIZE | EXPAND_DICT_SIZE | ROWS_SAMPLED | PAGES_SAVED_PERCENT | BYTES_SAVED_PERCENT | AVG_COMPRESS_REC_LENGTH |
|--------------------|------------------|--------------|---------------------|---------------------|-------------------------|
| | | | | | |
| 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 |
| ... | | | | | |
| 13312 | 5312 | 35 | 65 | 84 | 100 |
| 5760 | 4248 | 45 | 76 | 79 | 98 |
| ... | | | | | |

Ejemplo 2: Determinar la acción de creación del diccionario y la hora de la creación del diccionario para todas las tablas.

```
SELECT TABSCHEMA, TABNAME, DBPARTITIONNUM, DATA_PARTITION_ID, DICT_BUILDER, DICT_BUILD_TIMESTAMP
FROM SYSIBMADM.ADMINTABCOMPRESSINFO
```

El ejemplo siguiente muestra los datos de salida de la consulta:

| TABSCHEMA | TABNAME | DBPARTITIONNUM | DATA_PARTITION_ID | DICT_BUILDER | DICT_BUILD_TIMESTAMP |
|-----------|-----------|----------------|-------------------|--------------|----------------------------|
| SYSIBM | SYSTABLES | 0 | 0 | NOT BUILT | - |
| SYSIBM | SYSOLUMNS | 0 | 0 | NOT BUILT | - |
| ... | | | | | |
| SIMAP2 | STAFF | 0 | 0 | REORG | 2006-08-27-19.07.36.000000 |
| SIMAP2 | SALES | 0 | 0 | NOT BUILT | - |
| SIMAP2 | CATALOG | 0 | 0 | NOT BUILT | - |
| ... | | | | | |

156 registro(s) seleccionado(s).

Función de tabla ADMIN_GET_TAB_COMPRESS_INFO

La función de tabla ADMIN_GET_TAB_COMPRESS_INFO devuelve la misma información que la vista administrativa ADMINTABCOMPRESSINFO, pero le permite especificar un esquema, un nombre de tabla y una modalidad de ejecución.

En la tabla Vista administrativa ADMINTABCOMPRESSINFO y los metadatos de la función de tabla ADMIN_GET_TAB_COMPRESS_INFO encontrará una lista completa de la información que puede devolverse.

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la "Vista administrativa ADMINTABCOMPRESSINFO y función de tabla ADMIN_GET_TAB_COMPRESS_INFO_V97 - Devolver información de compresión" en la página 246.

Sintaxis

```
►►—ADMIN_GET_TAB_COMPRESS_INFO—(—esquematabla—,—nombretabla—,—execmode—)————►◄
```

El esquema es SYSPROC.

Parámetros de la función de tabla

esquematabla

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de esquema.

nombretabla

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de tabla, un nombre de tabla de consultas materializadas o un nombre de tabla de jerarquía.

modejec

Argumento de entrada de tipo VARCHAR(30) que especifica la modalidad de ejecución. La modalidad de ejecución puede ser una de las siguientes:

- 'REPORT' -- Notifica la información de compresión de última generación. Es el valor por omisión.
- 'ESTIMATE' -- Genera la nueva información de compresión a partir de la tabla actual.

Autorización

Privilegio EXECUTE en la función ADMIN_GET_TAB_COMPRESS_INFO.

Ejemplos

Ejemplo 1: Recuperar la información de compresión existente para la tabla SIMAP2.STAFF

```
SELECT * FROM TABLE (SYSPROC.ADMIN_GET_TAB_COMPRESS_INFO('SIMAP2', 'STAFF', 'REPORT'))
AS T
```

El siguiente ejemplo muestra los datos de salida de esta consulta:

| TABSCHEMA | TABNAME | DBPARTITIONNUM | DATA_PARTITION_ID | COMPRESS_ATTR | DICT_BUILDER | DICT_BUILD_TIMESTAMP |
|-----------|---------|----------------|-------------------|---------------|--------------|----------------------------|
| SIMAP2 | STAFF | 0 | 0 Y | | REORG | 2006-08-27-19.07.36.000000 |

1 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación):

| COMPRESS_DICT_SIZE | EXPAND_DICT_SIZE | ROWS_SAMPLED | PAGES_SAVED_PERCENT | BYTES_SAVED_PERCENT | AVG_COMPRESS_REC_LENGTH |
|--------------------|------------------|--------------|---------------------|---------------------|-------------------------|
| 13312 | 5312 | 35 | 65 | 84 | 100 |

Ejemplo 2: Recuperar la información de compresión estimada para la tabla SIMAP2.STAFF desde ya.

```
SELECT * FROM TABLE (SYSPROC.ADMIN_GET_TAB_COMPRESS_INFO('SIMAP2', 'STAFF', 'ESTIMATE'))
AS T
```

El siguiente ejemplo muestra los datos de salida de esta consulta:

| TABSCHEMA | TABNAME | DBPARTITIONNUM | DATA_PARTITION_ID | COMPRESS_ATTR | DICT_BUILDER | DICT_BUILD_TIMESTAMP |
|-----------|---------|----------------|-------------------|---------------|----------------|----------------------------|
| SIMAP2 | STAFF | 0 | 0 Y | | TABLE FUNCTION | 2006-08-28-19.18.13.000000 |

1 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación):

| COMPRESS_DICT_SIZE | EXPAND_DICT_SIZE | ROWS_SAMPLED | PAGES_SAVED_PERCENT | BYTES_SAVED_PERCENT | AVG_COMPRESS_REC_LENGTH |
|--------------------|------------------|--------------|---------------------|---------------------|-------------------------|
| 13508 | 6314 | 68 | 72 | 89 | 98 |

Ejemplo 3: Determinar el tamaño total del diccionario para todas las tablas del esquema SIMAP2

```
SELECT TABSCHEMA, TABNAME, DICT_BUILDER,
       (COMPRESS_DICT_SIZE+EXPAND_DICT_SIZE) AS TOTAL_DICT_SIZE,
       DBPARTITIONNUM, DATA_PARTITION_ID
FROM TABLE (SYSPROC.ADMIN_GET_TAB_COMPRESS_INFO('SIMAP2', '', 'REPORT')) AS T
```

Datos de salida de esta consulta:

| TABSCHEMA | TABNAME | DICT_BUILDER | TOTAL_DICT_SIZE | DBPARTITIONNUM | DATA_PARTITION_ID |
|-----------|-----------|--------------|-----------------|----------------|-------------------|
| SIMAP2 | ACT | NOT BUILT | 0 | 0 | 0 |
| SIMAP2 | ADEFUSR | NOT BUILT | 0 | 0 | 0 |
| ... | | | | | |
| SIMAP2 | INVENTORY | NOT BUILT | 0 | 0 | 0 |
| SIMAP2 | ORG | NOT BUILT | 0 | 0 | 0 |
| SIMAP2 | PARTTAB | REORG | 10008 | 0 | 0 |
| SIMAP2 | PARTTAB | REORG | 5464 | 0 | 1 |
| SIMAP2 | PARTTAB | REORG | 8456 | 0 | 2 |
| SIMAP2 | PARTTAB | REORG | 6960 | 0 | 3 |
| SIMAP2 | PARTTAB | REORG | 7136 | 0 | 4 |
| ... | | | | | |
| SIMAP2 | STAFF | REORG | 18624 | 0 | 0 |


```

SIMAP2      SUPPLIERS      NOT BUILT      0      0      0
SIMAP2      TESTTABLE      NOT BUILT      0      0      0

```

28 registro(s) seleccionado(s).

Ejemplo 4: Ver un informe de la información de diccionario de las tablas en el esquema SIMAP2.

```

SELECT * FROM TABLE (SYSPROC.ADMIN_GET_TAB_COMPRESS_INFO('SIMAP2', '', 'REPORT'))
AS T

```

Datos de salida de esta consulta:

| TABSCHEMA | TABNAME | DBPARTITIONNUM | DATA_PARTITION_ID | COMPRESS_ATTR | DICT_BUILDER | DICT_BUILD_TIMESTAMP |
|-----------|---------|----------------|-------------------|---------------|--------------|----------------------------|
| SIMAP2 | T1 | 0 | 0 | Y | NOT BUILT | - |
| SIMAP2 | T2 | 0 | 0 | N | REORG | 2007-02-03-17.35.28.000000 |
| SIMAP2 | T3 | 0 | 0 | Y | INSPECT | 2007-02-03-17.35.44.000000 |
| SIMAP2 | T4 | 0 | 0 | N | NOT BUILT | - |

4 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación):

| COMPRESS_DICT_SIZE | EXPAND_DICT_SIZE | ROWS_SAMPLED | PAGES_SAVED_PERCENT | BYTES_SAVED_PERCENT | AVG_COMPRESS_REC_LENGTH |
|--------------------|------------------|--------------|---------------------|---------------------|-------------------------|
| 0 | 0 | 0 | 0 | 0 | 0 |
| 1280 | 2562 | - | - | - | - |
| 1340 | 2232 | - | - | - | - |
| 0 | 0 | 0 | 0 | 0 | 0 |

Notas de uso

- Si se especifican *esquematabla* y *nombretabla*, sólo se devuelve información para esa tabla específica.
- Si se especifica el *esquematabla* pero *nombretabla* está vacío (") o es NULL, se devolverá información para todas las tablas del esquema determinado.
- Si *esquematabla* está vacío (") o es NULL y se especifica *nombretabla*, se devolverá un error. Para recuperar información para una tabla determinada, tanto el esquema como el nombre de la tabla deberán identificar a la tabla.
- Si *esquematabla* y *nombretabla* están vacíos (") o son NULL, se devolverá información para todas las tablas.
- Si no existe *esquematabla* o *nombretabla*, o si *nombretabla* no corresponde a un nombre de tabla (tipo T), un nombre de tabla de consultas materializadas (tipo S) o un nombre de tabla de jerarquía (tipo H), se devuelve un conjunto de resultados vacío.
- Cuando la función de tabla ADMIN_GET_TAB_COMPRESS_INFO recupere datos para una tabla determinada, necesitará un bloqueo compartido en la fila correspondiente de SYSTABLES para garantizar la coherencia de los datos que se devuelven (por ejemplo, para garantizar que la tabla no se descarte mientras se recupera información para ésta). El bloqueo sólo se mantendrá mientras tiene lugar la recuperación de la información de compresión para la tabla, no mientras está activa la llamada de función de tabla.

Vista administrativa ADMINTABCOMPRESSINFO y metadatos de la función de tabla ADMIN_GET_TAB_COMPRESS_INFO

Tabla 232. Vista administrativa ADMINTABCOMPRESSINFO y metadatos de la función de tabla ADMIN_GET_TAB_COMPRESS_INFO

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|-------------------|
| TABSCHEMA | VARCHAR(128) | Nombre de esquema |
| TABNAME | VARCHAR(128) | Nombre de tabla |

Tabla 232. Vista administrativa ADMINTABCOMPRESSINFO y metadatos de la función de tabla ADMIN_GET_TAB_COMPRESS_INFO (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|----------------------------|---------------|---|
| DBPARTITIONNUM | SMALLINT | Número de partición de base de datos |
| DATA_PARTITION_ID | INTEGER | Número de partición de datos |
| COMPRESS_ATTR | CHAR(1) | El estado del valor COMPRESS en la tabla, que puede ser uno de los siguientes: <ul style="list-style-type: none"> • 'Y' = Compresión de filas establecida en sí • 'N' = Compresión de filas establecida en no |
| DICTIONARY_BUILDER | VARCHAR(30) | Vía de acceso del código tomado para crear el diccionario, que puede ser una de las siguientes: <ul style="list-style-type: none"> • 'INSPECT' = INSPECT ROWCOMPESTIMATE • 'LOAD' = LOAD INSERT/REPLACE • 'NOT BUILT' = ningún diccionario disponible • 'REDISTRIBUTE' = REDISTRIBUTE • 'REORG' = REORG RESETDICTIONARY • 'TABLE GROWTH' = INSERT |
| DICTIONARY_BUILD_TIMESTAMP | TIMESTAMP | Fecha y hora en que se creó el diccionario. La granularidad horaria es de segundos. Si no hay ningún diccionario disponible, entonces la fecha y hora será NULL. |
| COMPRESS_DICT_SIZE | BIGINT | Tamaño del diccionario de compresión medido en bytes. |
| EXPAND_DICT_SIZE | BIGINT | Tamaño del diccionario de expansión medido en bytes. Si existe un diccionario histórico, este valor es la suma de los tamaños de los diccionarios actual e histórico. |
| ROWS_SAMPLED | INTEGER | Número de registros que han contribuido a crear el diccionario. Las tablas migradas con diccionarios de compresión devolverán NULL en esta columna. |
| PAGES_SAVED_PERCENT | SMALLINT | Porcentaje de páginas salvadas de la compresión. Esta información se basa en los datos de registro del almacenamiento intermedio de ejemplo únicamente. Las tablas migradas con diccionarios de compresión devolverán NULL en esta columna. |
| BYTES_SAVED_PERCENT | SMALLINT | Porcentaje de bytes salvadas de la compresión. Esta información se basa en los datos de registro del almacenamiento intermedio de ejemplo únicamente. Las tablas migradas con diccionarios de compresión devolverán NULL en esta columna. |
| AVG_COMPRESS_REC_LENGTH | SMALLINT | La longitud media de registro comprimida de los registros que permiten crear el diccionario. Las tablas migradas con diccionarios de compresión devolverán NULL en esta columna. |

RGET_DB_CONFIG

Nota: Este procedimiento ha quedado obsoleto y se ha sustituido por la "Vista administrativa DBCFG - Recuperar información sobre los parámetros de configuración de la base de datos" en la página 342.

▶▶—GET_DB_CONFIG—(—)————▶▶

El esquema es SYSPROC.

El procedimiento GET_DB_CONFIG devuelve información de configuración de base de datos. El procedimiento no utiliza ningún argumento de entrada.

El procedimiento devuelve un conjunto de resultados individual con dos filas que contienen una columna para cada parámetro. La primera columna se denomina DBCONFIG_TYPE, tal como se muestra a continuación.

Tabla 233. Información devuelta por el procedimiento GET_DB_CONFIG

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| DBCONFIG_TYPE | INTEGER | La fila con el valor 0 en esta columna contiene los valores de los parámetros de configuración de base de datos almacenados en disco. La fila con el valor 1 en esta columna contiene los valores actuales de los parámetros de configuración de base de datos almacenados en la memoria. |

Este procedimiento necesita un espacio de tablas temporal de usuario que se utiliza para crear una tabla temporal global llamada DB_CONFIG donde se almacena el conjunto de resultados.

Ejemplo

Mediante el procesador de línea de mandatos (CLP), cambie el valor de los parámetros de configuración de base de datos *logretain* y *userexit*. Recupere el valor original (en disco) y el valor actualizado (en la memoria) invocando el procedimiento GET_DB_CONFIG y luego ejecutando una consulta para la tabla temporal global resultante (DB_CONFIG).

```
CONNECT TO SAMPLE

CREATE BUFFERPOOL MY8KPOOL SIZE 250 PAGESIZE 8K

CREATE USER TEMPORARY TABLESPACE MYTSP2 PAGESIZE
      8K MANAGED BY SYSTEM USING ( 'TSC2' ) BUFFERPOOL MY8KPOOL

UPDATE DB CFG USING LOGRETAIN RECOVERY USEREXIT ON

CALL SYSPROC.GET_DB_CONFIG()

SELECT DBCONFIG_TYPE, LOGRETAIN, USEREXIT
      FROM SESSION.DB_CONFIG

CONNECT RESET
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
DBCONFIG_TYPE LOGRETAIN  USEREXIT
-----
              0           1           1
              1           0           0
```

2 registro(s) seleccionado(s).

GET_DBM_CONFIG

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la “Vista administrativa DBMCFG - Recuperar información sobre los parámetros de configuración del gestor de bases de datos” en la página 344.

►►—GET_DBM_CONFIG—(—)—————►►

El esquema es SYSFUN.

La función de tabla GET_DBM_CONFIG devuelve información de configuración del gestor de bases de datos. La función no toma ningún argumento.

La función devuelve una tabla con dos filas que contienen una columna para cada parámetro. La primera columna se denomina DBMCONFIG_TYPE, tal como se muestra a continuación.

Tabla 234. Información devuelta por la función de tabla GET_DBM_CONFIG

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| DBMCONFIG_TYPE | INTEGER | La fila con el valor 0 en esta columna contiene los valores de los parámetros de configuración del gestor de bases de datos almacenados en disco. La fila con el valor 1 en esta columna contiene los valores actuales de los parámetros de configuración del gestor de bases de datos almacenados en la memoria. |

Ejemplo

Utilizando el procesador de línea de mandatos (CLP), cambie el valor de los parámetros *numdb* y *diaglevel* del gestor de bases de datos y luego recupere el valor original (contenido en disco) y el valor actualizado (contenido en la memoria).

```
UPDATE DBM CFG USING NUMDB 32 DIAGLEVEL 4

CONNECT TO SAMPLE

SELECT DBMCONFIG_TYPE, NUMDB, DIAGLEVEL
FROM TABLE(SYSFUN.GET_DBM_CONFIG()) AS DBMCFG

CONNECT RESET
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
DBMCONFIG_TYPE NUMDB      DIAGLEVEL
-----
                0         32          4
                1          8          3
```

2 registro(s) seleccionado(s).

Rutinas de instantáneas de salud

HEALTH_CONT_HI

La función de tabla HEALTH_CONT_HI devuelve información del indicador de salud correspondiente a contenedores de espacio de tablas desde una instantánea de los espacios de tablas de una base de datos.

Importante: Esta función de tabla ha quedado en desuso y puede que se elimine en un futuro release ya que el Supervisor de salud ha quedado en desuso en la Versión 9.7. Para obtener más información, consulte el tema “El Supervisor de salud ha quedado en desuso” de la publicación *Novedades en DB2 Versión 9.7*.

Sintaxis

►►—HEALTH_CONT_HI—(—nombre-bd—,—númparticiónbd—)—————►►

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(255) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique el valor nulo para tomar la instantánea desde la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE para la función de tabla HEALTH_CONT_HI.

Ejemplo

```
SELECT * FROM TABLE(HEALTH_CONT_HI('',-1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| SNAPSHOT_TIMESTAMP | CONTAINER_NAME | ... |
|----------------------------|--|-----|
| 2006-02-13-12.30.40.759542 | D:\DB2\NODE0000\SAMPLE\T0000000\C0000000.CAT | ... |
| 2006-02-13-12.30.40.759542 | D:\DB2\NODE0000\SAMPLE\T0000003\C0000000.LRG | ... |
| 2006-02-13-12.30.40.759542 | D:\DB2\NODE0000\SAMPLE\T0000004\C0000000.UTM | ... |
| 2006-02-13-12.30.40.759542 | D:\DB2\NODE0000\SAMPLE\T0000001\C0000000.TMP | ... |
| 2006-02-13-12.30.40.759542 | D:\DB2\NODE0000\SAMPLE\T0000002\C0000000.LRG | ... |

5 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```

... NODE_NUMBER HI_ID                HI_VALUE HI_TIMESTAMP                ...
... -----
... -                3001                1 2006-02-13-12.26.26.158000 ...
... -                3001                1 2006-02-13-12.26.26.158000 ...
... -                3001                1 2006-02-13-12.26.26.158000 ...
... -                3001                1 2006-02-13-12.26.26.158000 ...
... -                3001                1 2006-02-13-12.26.26.158000 ...

```

Datos de salida de esta consulta (continuación).

```

... HI_ALERT_STATE    HI_ALERT_STATE_DETAIL HI_FORMULA    HI_ADDITIONAL_INFO
... -----
...                1 Normal                1                -
...                1 Normal                1                -
...                1 Normal                1                -
...                1 Normal                1                -
...                1 Normal                1                -

```

Información devuelta

Tabla 235. Información devuelta por la función de tabla HEALTH_CONT_HI

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| CONTAINER_NAME | VARCHAR(256) | container_name - Nombre de contenedor |
| NODE_NUMBER | INTEGER | node_number - Número de nodo |
| HI_ID | BIGINT | Número que identifica de forma exclusiva el indicador de salud en la corriente de datos de la instantánea. |
| HI_VALUE | SMALLINT | Valor del indicador de salud. |
| HI_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se generó la alerta. |
| HI_ALERT_STATE | BIGINT | La gravedad de la alerta. |
| HI_ALERT_STATE_DETAIL | VARCHAR(20) | Descripción textual de la columna HI_ALERT_STATE. |
| HI_FORMULA | VARCHAR(2048) | Fórmula utilizada para calcular el indicador de salud. |
| HI_ADDITIONAL_INFO | VARCHAR(4096) | Información adicional acerca del indicador de salud. |

HEALTH_CONT_HI_HIS

Devuelve información histórica de indicador de salud para contenedores a partir de una instantánea de salud de una base de datos.

Importante: Esta función de tabla ha quedado en desuso y puede que se elimine en un futuro release ya que el Supervisor de salud ha quedado en desuso en la Versión 9.7. Para obtener más información, consulte el tema “El Supervisor de salud ha quedado en desuso” de la publicación *Novedades en DB2 Versión 9.7*.

Sintaxis

►►—HEALTH_CONT_HI_HIS—(—nombre-bd—,—númparticiónbd—)————►►

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(255) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique el valor nulo para tomar la instantánea desde la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE para la función de tabla HEALTH_CONT_HI_HIS.

Ejemplo

```
SELECT * FROM TABLE(HEALTH_CONT_HI_HIS('',-1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| SNAPSHOT_TIMESTAMP | CONTAINER_NAME | ... |
|----------------------------|--|-----|
| 2006-02-13-12.30.41.915646 | D:\DB2\NODE0000\SAMPLE\T0000000\C0000000.CAT | ... |
| 2006-02-13-12.30.41.915646 | D:\DB2\NODE0000\SAMPLE\T0000000\C0000000.CAT | ... |
| 2006-02-13-12.30.41.915646 | D:\DB2\NODE0000\SAMPLE\T0000003\C0000000.LRG | ... |
| 2006-02-13-12.30.41.915646 | D:\DB2\NODE0000\SAMPLE\T0000003\C0000000.LRG | ... |
| 2006-02-13-12.30.41.915646 | D:\DB2\NODE0000\SAMPLE\T0000004\C0000000.UTM | ... |
| 2006-02-13-12.30.41.915646 | D:\DB2\NODE0000\SAMPLE\T0000004\C0000000.UTM | ... |
| 2006-02-13-12.30.41.915646 | D:\DB2\NODE0000\SAMPLE\T0000001\C0000000.TMP | ... |
| 2006-02-13-12.30.41.915646 | D:\DB2\NODE0000\SAMPLE\T0000001\C0000000.TMP | ... |
| 2006-02-13-12.30.41.915646 | D:\DB2\NODE0000\SAMPLE\T0000002\C0000000.LRG | ... |
| 2006-02-13-12.30.41.915646 | D:\DB2\NODE0000\SAMPLE\T0000002\C0000000.LRG | ... |

10 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

| ... | NODE_NUMBER | HI_ID | HI_TIMESTAMP | HI_VALUE | HI_ALERT_STATE | ... |
|-----|-------------|-------|----------------------------|----------|----------------|-----|
| ... | - | 3001 | 2006-02-13-12.16.25.911000 | 1 | 1 | ... |
| ... | - | 3001 | 2006-02-13-12.06.26.168000 | 1 | 1 | ... |
| ... | - | 3001 | 2006-02-13-12.16.25.911000 | 1 | 1 | ... |
| ... | - | 3001 | 2006-02-13-12.06.26.168000 | 1 | 1 | ... |
| ... | - | 3001 | 2006-02-13-12.16.25.911000 | 1 | 1 | ... |
| ... | - | 3001 | 2006-02-13-12.06.26.168000 | 1 | 1 | ... |
| ... | - | 3001 | 2006-02-13-12.16.25.911000 | 1 | 1 | ... |

```

...      -      3001 2006-02-13-12.06.26.168000      1      1 ...
...      -      3001 2006-02-13-12.16.25.911000      1      1 ...
...      -      3001 2006-02-13-12.06.26.168000      1      1 ...

```

Datos de salida de esta consulta (continuación).

```

... HI_ALERT_STATE_DETAIL HI_FORMULA      HI_ADDITIONAL_INFO
... -----
... Normal                1                -
... Normal                1                -
... Normal                1                -
... Normal                1                -
... Normal                1                -
... Normal                1                -
... Normal                1                -
... Normal                1                -
... Normal                1                -
... Normal                1                -

```

Información devuelta

Tabla 236. Información devuelta por la función de tabla HEALTH_CONT_HI_HIS

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| CONTAINER_NAME | VARCHAR(256) | container_name - Nombre de contenedor |
| NODE_NUMBER | INTEGER | node_number - Número de nodo |
| HI_ID | BIGINT | Número que identifica de forma exclusiva el indicador de salud en la corriente de datos de la instantánea. |
| HI_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se generó la alerta. |
| HI_VALUE | SMALLINT | Valor del indicador de salud. |
| HI_ALERT_STATE | BIGINT | La gravedad de la alerta. |
| HI_ALERT_STATE_DETAIL | VARCHAR(20) | Descripción textual de la columna HI_ALERT_STATE. |
| HI_FORMULA | VARCHAR(2048) | Fórmula utilizada para calcular el indicador de salud. |
| HI_ADDITIONAL_INFO | VARCHAR(4096) | Información adicional acerca del indicador de salud. |

HEALTH_CONT_INFO

La función de tabla HEALTH_CONT_INFO devuelve información de contenedor desde una instantánea de la salud de una base de datos.

Importante: Esta función de tabla ha quedado en desuso y puede que se elimine en un futuro release ya que el Supervisor de salud ha quedado en desuso en la Versión 9.7. Para obtener más información, consulte el tema “El Supervisor de salud ha quedado en desuso” de la publicación *Novedades en DB2 Versión 9.7*.

Sintaxis

►►—HEALTH_CONT_INFO—(—nombre-bd—,—númparticiónbd—)—————►►

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(255) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique el valor nulo para tomar la instantánea desde la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE para la función de tabla HEALTH_CONT_INFO.

Ejemplo

```
SELECT * FROM TABLE(HEALTH_CONT_INFO('',-1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| SNAPSHOT_TIMESTAMP | CONTAINER_NAME | ... |
|----------------------------|--|-----|
| 2006-02-13-12.30.40.541209 | D:\DB2\NODE0000\SAMPLE\T0000000\C0000000.CAT | ... |
| 2006-02-13-12.30.40.541209 | D:\DB2\NODE0000\SAMPLE\T0000003\C0000000.LRG | ... |
| 2006-02-13-12.30.40.541209 | D:\DB2\NODE0000\SAMPLE\T0000004\C0000000.UTM | ... |
| 2006-02-13-12.30.40.541209 | D:\DB2\NODE0000\SAMPLE\T0000001\C0000000.TMP | ... |
| 2006-02-13-12.30.40.541209 | D:\DB2\NODE0000\SAMPLE\T0000002\C0000000.LRG | ... |

5 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

| TABLESPACE_NAME | NODE_NUMBER | ... |
|------------------|-------------|-----|
| SYSCATSPACE | - | ... |
| SYSTOOLSPACE | - | ... |
| SYSTOOLSTMPSPACE | - | ... |
| TEMPSPACE1 | - | ... |
| USERSPACE1 | - | ... |

Datos de salida de esta consulta (continuación).

| ROLLED_UP_ALERT_STATE | ROLLED_UP_ALERT_STATE_DETAIL |
|-----------------------|------------------------------|
| 1 | Normal |

... 1 Normal
 ... 1 Normal
 ... 1 Normal
 ... 1 Normal

Información devuelta

Tabla 237. Información devuelta por la función de tabla HEALTH_CONT_INFO

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| CONTAINER_NAME | VARCHAR(256) | container_name - Nombre de contenedor |
| TABLESPACE_NAME | VARCHAR(128) | tablespace_name - Nombre de espacio de tablas |
| NODE_NUMBER | INTEGER | node_number - Número de nodo |
| ROLLED_UP_ALERT_STATE | BIGINT | Estado de alerta más grave capturado por esta instantánea. |
| ROLLED_UP_ALERT_STATE_DETAIL | VARCHAR(20) | Descripción textual de la columna ROLLED_UP_ALERT_STATE. |

HEALTH_DB_HI

La función de tabla HEALTH_DB_HI devuelve información del indicador de salud desde una instantánea de la salud de una base de datos.

Importante: Esta función de tabla ha quedado en desuso y puede que se elimine en un futuro release ya que el Supervisor de salud ha quedado en desuso en la Versión 9.7. Para obtener más información, consulte el tema “El Supervisor de salud ha quedado en desuso” de la publicación *Novedades en DB2 Versión 9.7*.

Sintaxis

►►—HEALTH_DB_HI—(—nombre-bd—,—núm-partición-bd—)—►►

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(255) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique el valor nulo para tomar la instantánea desde todas las bases de datos bajo la instancia de la base de datos.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos


```

... ((1170384 / (1170384 + 19229616)) * 100)    ...
...
...
...
...
...
... ((1155116032 / 21138935808) * 100)
...
...
...
...
...
...
... ((5264 / (50 * 4096)) * 100)
... ((0 / 5) * 100)
... ((4587520 / 6160384) * 100)
... -
...
...
...
...
...
... -
...
...
...
...
...
...
... -
...
...
...

```

Datos de salida de esta consulta (continuación).

```

... HI_ADDITIONAL_INFO
... -----
... -
... The high watermark for shared sort
... memory is "57". "99"% of the time
... the sort heap allocation is less
... than or equal to "246". The sort
... heap (sortheap) database
... configuration parameter is set
... to "256". The high watermark for
... private sort memory is "0".
... The sort heap (sortheap) database
... configuration parameter is set to
... "256". The high watermark for
... private sort memory is "57". The
... high watermark for shared sort
... memory is "0"
... The following are the related
... database configuration parameter
... settings: logprimary is "3",
... logsecond is "2", and logfilsiz
... is "1000". The application with
... the oldest transaction is "712".
... The following are the related
... database configuration parameter
... settings: logprimary is "3",
... logsecond is "2", and logfilsiz
... is "1000", blk_log_dsk_ful is
... "NO", userexit is "NO",
... logarchmeth1 is "OFF" and

```

```

... logarchmeth2 is "OFF".
... -
... -
... -
... The scope setting in the reorganization
... policy is "TABSCHEMA NOT LIKE 'SYS%'".
... Automatic reorganization (AUTO_REORG)
... for this database is set to "OFF".
... The longest estimated reorganization
... time is "N/A".
... The last successful backup was taken
... at "N/A". The log space consumed since
... this last backup has been "N/A" 4KB
... pages. Automation for database backup
... is set to "OFF". The last automated
... backup returned with SQLCODE = "N/A".
... The longest estimated backup time
... is "N/A".
... The scope is "N/A". Automatic
... statistics collection (AUTO_RUNSTATS)
... is set to "OFF".

```

Información devuelta

Tabla 238. Información devuelta por la función de tabla HEALTH_DB_HI

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| HI_ID | BIGINT | Número que identifica de forma exclusiva el indicador de salud en la corriente de datos de la instantánea. |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| HI_VALUE | SMALLINT | Valor del indicador de salud. |
| HI_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se generó la alerta. |
| HI_ALERT_STATE | BIGINT | La gravedad de la alerta. |
| HI_ALERT_STATE_DETAIL | VARCHAR(20) | Descripción textual de la columna HI_ALERT_STATE. |
| HI_FORMULA | VARCHAR(2048) | Fórmula utilizada para calcular el indicador de salud. |
| HI_ADDITIONAL_INFO | VARCHAR(4096) | Información adicional acerca del indicador de salud. |

HEALTH_DB_HI_HIS

La función de tabla HEALTH_DB_HI_HIS devuelve información histórica del indicador de salud desde una instantánea de salud de una base de datos.

Importante: Esta función de tabla ha quedado en desuso y puede que se elimine en un futuro release ya que el Supervisor de salud ha quedado en desuso en la Versión 9.7. Para obtener más información, consulte el tema “El Supervisor de salud ha quedado en desuso” de la publicación *Novedades en DB2 Versión 9.7*.

Sintaxis

►►—HEALTH_DB_HI_HIS—(—*nombre-bd*—,—*núm-partición-bd*—)—————►►

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(255) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique el valor nulo para tomar la instantánea desde todas las bases de datos bajo la instancia de la base de datos.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE para la función de tabla HEALTH_DB_HI_HIS.

Ejemplo

```
SELECT * FROM TABLE(HEALTH_DB_HI_HIS('',-1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| SNAPSHOT_TIMESTAMP | HI_ID | DB_NAME | HI_VALUE | ... |
|----------------------------|-------|---------|----------|-----|
| 2006-02-13-12.30.26.325627 | 1001 | SAMPLE | 0 | ... |
| ... | ... | ... | ... | ... |
| 2006-02-13-12.30.26.325627 | 1002 | SAMPLE | 0 | ... |
| ... | ... | ... | ... | ... |
| 2006-02-13-12.30.26.325627 | 1003 | SAMPLE | 0 | ... |
| ... | ... | ... | ... | ... |
| 2006-02-13-12.30.26.325627 | 1005 | SAMPLE | 3 | ... |
| ... | ... | ... | ... | ... |
| 2006-02-13-12.30.26.325627 | 1008 | SAMPLE | 2 | ... |
| ... | ... | ... | ... | ... |
| 2006-02-13-12.30.26.325627 | 1010 | SAMPLE | 0 | ... |
| ... | ... | ... | ... | ... |
| 2006-02-13-12.30.26.325627 | 1014 | SAMPLE | 73 | ... |
| ... | ... | ... | ... | ... |
| 2006-02-13-12.30.26.325627 | 1015 | SAMPLE | 1 | ... |
| ... | ... | ... | ... | ... |
| 2006-02-13-12.30.26.325627 | 1018 | SAMPLE | 1 | ... |
| ... | ... | ... | ... | ... |
| 2006-02-13-12.30.26.325627 | 1022 | SAMPLE | 1 | ... |
| ... | ... | ... | ... | ... |

Datos de salida de esta consulta (continuación).

| HI_TIMESTAMP | HI_ALERT_STATE | HI_ALERT_STATE_DETAIL |
|----------------------------|----------------|-----------------------|
| 2006-02-13-12.21.25.649000 | 1 | Normal |
| 2006-02-13-12.21.25.649000 | 1 | Normal |
| 2006-02-13-12.20.25.182000 | 1 | Normal |
| 2006-02-13-12.16.25.911000 | 1 | Normal |
| 2006-02-13-12.16.25.911000 | 1 | Normal |
| 2006-02-13-12.16.25.911000 | 1 | Normal |
| 2006-02-13-12.21.25.649000 | 1 | Normal |
| 2006-02-13-12.29.55.461000 | 2 | Attention |
| 2006-02-13-12.29.25.281000 | 2 | Attention |
| 2006-02-13-12.27.55.743000 | 2 | Attention |

Datos de salida de esta consulta (continuación).

| HI_FORMULA |
|--|
| 0 |
| $((0 / 5000) * 100)$ |
| $((0 - 0) / ((68 - 0) + 1)) * 100)$ |
| $((698410 / (698410 + 19701590)) * 100)$ |
| $((3920 / (50 * 4096)) * 100)$ |
| $((0 / 4) * 100)$ |
| $((4521984 / 6160384) * 100)$ |
| - |


```

... The scope is "N\A". Automatic
... statistics collection
... (AUTO_RUNSTATS) is set to "OFF".
...

```

Información devuelta

Tabla 239. Información devuelta por la función de tabla HEALTH_DB_HI_HIS

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| HI_ID | BIGINT | Número que identifica de forma exclusiva el indicador de salud en la corriente de datos de la instantánea. |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| HI_VALUE | SMALLINT | Valor del indicador de salud. |
| HI_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se generó la alerta. |
| HI_ALERT_STATE | BIGINT | La gravedad de la alerta. |
| HI_ALERT_STATE_DETAIL | VARCHAR(20) | Descripción textual de la columna HI_ALERT_STATE. |
| HI_FORMULA | VARCHAR(2048) | Fórmula utilizada para calcular el indicador de salud. |
| HI_ADDITIONAL_INFO | VARCHAR(4096) | Información adicional acerca del indicador de salud. |

HEALTH_DB_HIC

La función HEALTH_DB_HIC devuelve información del indicador de salud referente a colecciones desde una instantánea de la salud de una base de datos.

Importante: Esta función de tabla ha quedado en desuso y puede que se elimine en un futuro release ya que el Supervisor de salud ha quedado en desuso en la Versión 9.7. Para obtener más información, consulte el tema “El Supervisor de salud ha quedado en desuso” de la publicación *Novedades en DB2 Versión 9.7*.

Sintaxis

►►—HEALTH_DB_HIC—(—nombre-bd—,—númparticiónbd—)—◀◀

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(255) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto

por el mandato LIST DATABASE DIRECTORY. Especifique el valor nulo para tomar la instantánea desde todas las bases de datos bajo la instancia de la base de datos.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE para la función de tabla HEALTH_DB_HIC.

Ejemplo

```
SELECT * FROM TABLE(HEALTH_DB_HIC('',-1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
SNAPSHOT_TIMESTAMP          HI_ID    DB_NAME    ...
-----
2006-02-13-12.30.33.870959  1015    SAMPLE    ...
2006-02-13-12.30.33.870959  1022    SAMPLE    ...
```

2 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```
... HI_OBJ_NAME                HI_OBJ_DETAIL    ...
... -----
... "JESSICAE"."EMPLOYEE"      REORG TABLE    ...
... "SYSIBM"."SYSDATAPARTITIONEXPRESSION" RUNSTATS        ...
```

Datos de salida de esta consulta (continuación).

```
... HI_OBJ_STATE HI_OBJ_STATE_DETAIL HI_TIMESTAMP
... -----
...           2 Attention                2006-02-13-12.24.27.000000
...           2 Attention                2006-02-13-12.29.26.000000
```

Información devuelta

Tabla 240. Información devuelta por la función de tabla HEALTH_DB_HIC

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| HI_ID | BIGINT | Número que identifica de forma exclusiva el indicador de salud en la corriente de datos de la instantánea. |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| HI_OBJ_NAME | VARCHAR(256) | Un nombre que identifica exclusivamente a un objeto de la colección. |

Tabla 240. Información devuelta por la función de tabla HEALTH_DB_HIC (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------|---------------|---|
| HI_OBJ_DETAIL | VARCHAR(4096) | Texto que describe el motivo por el que el objeto se ha añadido a la colección. |
| HI_OBJ_STATE | BIGINT | El estado del objeto. Los estados válidos (definidos en sqlmon.h) incluyen: <ul style="list-style-type: none"> • NORMAL (1). No se necesita acción alguna en este objeto. • ATTENTION (2). No se ha habilitado la automatización para este indicador de salud; la acción debe ser manual. • AUTOMATED (5). Se ha habilitado la automatización para este indicador de salud; la acción se iniciará automáticamente. • AUTOMATE_FAILED (6). Se ha habilitado la automatización para este indicador de salud; se ha iniciado la acción, pero no se ha podido completar satisfactoriamente. Ahora se necesita la intervención manual. |
| HI_OBJ_STATE_DETAIL | VARCHAR(20) | Versión convertida de la serie del valor de la columna HI_OBJ_STATE. |
| HI_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se generó la alerta. |

HEALTH_DB_HIC_HIS

Devuelve información histórica de indicador de salud de colección a partir de una instantánea de salud de una base de datos.

Importante: Esta función de tabla ha quedado en desuso y puede que se elimine en un futuro release ya que el Supervisor de salud ha quedado en desuso en la Versión 9.7. Para obtener más información, consulte el tema “El Supervisor de salud ha quedado en desuso” de la publicación *Novedades en DB2 Versión 9.7*.

Sintaxis

►►—HEALTH_DB_HIC_HIS—(—nombre-bd—,—núm-partición-bd—)—————►►

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(255) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique el valor nulo para tomar la instantánea desde todas las bases de datos bajo la instancia de la base de datos.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE para la función de tabla HEALTH_DB_HIC_HIS.

Ejemplo

```
SELECT * FROM TABLE(HEALTH_DB_HIC_HIS('',-1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| HI_HIS_ENTRY_NUM | SNAPSHOT_TIMESTAMP | HI_ID | ... |
|------------------|----------------------------|-------|-----|
| 1 | 2006-02-13-12.30.34.496720 | 1015 | ... |
| 2 | 2006-02-13-12.30.34.496720 | 1022 | ... |
| 3 | 2006-02-13-12.30.34.496720 | 1022 | ... |
| 4 | 2006-02-13-12.30.34.496720 | 1022 | ... |
| 5 | 2006-02-13-12.30.34.496720 | 1022 | ... |
| 6 | 2006-02-13-12.30.34.496720 | 1022 | ... |
| 7 | 2006-02-13-12.30.34.496720 | 1022 | ... |
| 8 | 2006-02-13-12.30.34.496720 | 1022 | ... |
| 9 | 2006-02-13-12.30.34.496720 | 1022 | ... |
| 10 | 2006-02-13-12.30.34.496720 | 1022 | ... |

10 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

| ... | DB_NAME | HI_OBJ_NAME | HI_OBJ_STATE | ... |
|-----|---------|---------------------------------------|--------------|-----|
| ... | SAMPLE | "JESSICAE"."EMPLOYEE" | 2 | ... |
| ... | SAMPLE | "SYSIBM"."SYSDATAPARTITIONEXPRESSION" | 2 | ... |
| ... | SAMPLE | "SYSIBM"."SYSDATAPARTITIONEXPRESSION" | 2 | ... |
| ... | SAMPLE | "SYSIBM"."SYSDATAPARTITIONEXPRESSION" | 2 | ... |
| ... | SAMPLE | "SYSIBM"."SYSDATAPARTITIONEXPRESSION" | 1 | ... |
| ... | SAMPLE | "SYSIBM"."SYSDATAPARTITIONEXPRESSION" | 1 | ... |
| ... | SAMPLE | "SYSIBM"."SYSDATAPARTITIONEXPRESSION" | 1 | ... |
| ... | SAMPLE | "SYSIBM"."SYSDATAPARTITIONEXPRESSION" | 1 | ... |
| ... | SAMPLE | "SYSIBM"."SYSDATAPARTITIONEXPRESSION" | 1 | ... |
| ... | SAMPLE | "SYSIBM"."SYSDATAPARTITIONEXPRESSION" | 1 | ... |

Datos de salida de esta consulta (continuación).

| ... | HI_OBJ_STATE_DETAIL | HI_TIMESTAMP |
|-----|---------------------|----------------------------|
| ... | Attention | 2006-02-10-09.04.57.000000 |

```

... Attention      2006-02-13-12.27.56.000000
... Attention      2006-02-13-12.26.27.000000
... Attention      2006-02-13-12.24.56.000000
... Normal         2006-02-13-12.23.28.000000
... Normal         2006-02-13-12.21.56.000000
... Normal         2006-02-13-12.20.26.000000
... Normal         2006-02-13-12.18.57.000000
... Normal         2006-02-13-12.17.27.000000
... Normal         2006-02-13-12.15.56.000000

```

Información devuelta

Tabla 241. Información devuelta por la función de tabla HEALTH_DB_HIC_HIS

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| HI_HIS_ENTRY_NUM | INTEGER | Número que identifica exclusivamente a la entrada del histórico. |
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| HI_ID | BIGINT | Número que identifica de forma exclusiva el indicador de salud en la corriente de datos de la instantánea. |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| HI_OBJ_NAME | VARCHAR(256) | Un nombre que identifica exclusivamente a un objeto de la colección. |
| HI_OBJ_STATE | BIGINT | El estado del objeto. Los estados válidos (definidos en sqlmon.h) incluyen: <ul style="list-style-type: none"> • NORMAL (1). No se necesita acción alguna en este objeto. • ATTENTION (2). No se ha habilitado la automatización para este indicador de salud; la acción debe ser manual. • AUTOMATED (5). Se ha habilitado la automatización para este indicador de salud; la acción se iniciará automáticamente. • AUTOMATE_FAILED (6). Se ha habilitado la automatización para este indicador de salud; se ha iniciado la acción, pero no se ha podido completar satisfactoriamente. Ahora se necesita la intervención manual. |

Tabla 241. Información devuelta por la función de tabla HEALTH_DB_HIC_HIS (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------|---------------|--|
| HI_OBJ_STATE_DETAIL | VARCHAR(20) | Versión convertida de la serie del valor de la columna HI_OBJ_STATE. |
| HI_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se generó la alerta. |

HEALTH_DB_INFO

La función de tabla HEALTH_DB_INFO devuelve información desde una instantánea de la salud de una base de datos.

Importante: Esta función de tabla ha quedado en desuso y puede que se elimine en un futuro release ya que el Supervisor de salud ha quedado en desuso en la Versión 9.7. Para obtener más información, consulte el tema “El Supervisor de salud ha quedado en desuso” de la publicación *Novedades en DB2 Versión 9.7*.

Sintaxis

►►—HEALTH_DB_INFO—(—nombre-bd—,—númparticiónbd—)—————►►

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(255) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique el valor nulo para tomar la instantánea desde todas las bases de datos bajo la instancia de la base de datos.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE para la función de tabla HEALTH_DB_INFO.

Ejemplo

```
SELECT * FROM TABLE(HEALTH_DB_INFO('',-1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```

SNAPSHOT_TIMESTAMP      DB_NAME      INPUT_DB_ALIAS      ...
-----
2006-02-13-12.30.23.340081 SAMPLE      SAMPLE      ...

```

1 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```

... DB_PATH      DB_LOCATION SERVER_PLATFORM ...
-----
... D:\DB2\NODE0000\SQL00003\      1      5 ...

```

Datos de salida de esta consulta (continuación).

```

... ROLLED_UP_ALERT_STATE ROLLED_UP_ALERT_STATE_DETAIL
-----
...      4 Alarm

```

Información devuelta

Tabla 242. Información devuelta por la función de tabla HEALTH_DB_INFO

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| INPUT_DB_ALIAS | VARCHAR(128) | input_db_alias - Alias de base de datos de entrada |
| DB_PATH | VARCHAR(1024) | db_path - Vía de acceso de base de datos |
| DB_LOCATION | INTEGER | db_location - Ubicación de base de datos |
| SERVER_PLATFORM | INTEGER | server_platform - Sistema operativo de servidor |
| ROLLED_UP_ALERT_STATE | BIGINT | Estado de alerta más grave capturado por esta instantánea. |
| ROLLED_UP_ALERT_STATE_DETAIL | VARCHAR(20) | Descripción textual de la columna ROLLED_UP_ALERT_STATE. |

HEALTH_DBM_HI

La función de tabla HEALTH_DBM_HI devuelve información del indicador de salud desde una instantánea de la salud del gestor de bases de datos de DB2.

Importante: Esta función de tabla ha quedado en desuso y puede que se elimine en un futuro release ya que el Supervisor de salud ha quedado en desuso en la Versión 9.7. Para obtener más información, consulte el tema “El Supervisor de salud ha quedado en desuso” de la publicación *Novedades en DB2 Versión 9.7*.

Sintaxis

►►—HEALTH_DBM_HI—(—númparticiónbd—)————►►

El esquema es SYSPROC.

Parámetro de la función de tabla

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE para la función de tabla HEALTH_DBM_HI.

Ejemplo

```
SELECT * FROM TABLE(HEALTH_DBM_HI(-1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
SNAPSHOT_TIMESTAMP      HI_ID      SERVER_INSTANCE_NAME    ...
-----
2006-02-13-12.30.19.773632  1 DB2      ...
2006-02-13-12.30.19.773632  4 DB2      ...
```

2 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```
... HI_VALUE HI_TIMESTAMP      HI_ALERT_STATE HI_ALERT_STATE_DETAIL ...
... -----
...      0 2006-02-13-12.26.26.158000      1 Normal      ...
...      100 2006-02-13-12.26.26.158000      4 Alarm      ...
```

Datos de salida de esta consulta (continuación).

```
... HI_FORMULA      HI_ADDITIONAL_INFO
... -----
... 0      -
... ((327680 / 327680) * 100)      -
```

Tabla 243. Información devuelta por la función de tabla HEALTH_DBM_HI

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|----------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| HI_ID | BIGINT | Número que identifica de forma exclusiva el indicador de salud en la corriente de datos de la instantánea. |
| SERVER_INSTANCE_NAME | VARCHAR(128) | server_instance_name - Nombre de instancia del servidor |
| HI_VALUE | SMALLINT | Valor del indicador de salud. |
| HI_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se generó la alerta. |
| HI_ALERT_STATE | BIGINT | La gravedad de la alerta. |

Tabla 243. Información devuelta por la función de tabla HEALTH_DBM_HI (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------|---------------|--|
| HI_ALERT_STATE_DETAIL | VARCHAR(20) | Descripción textual de la columna HI_ALERT_STATE. |
| HI_FORMULA | VARCHAR(2048) | Fórmula utilizada para calcular el indicador de salud. |
| HI_ADDITIONAL_INFO | VARCHAR(4096) | Información adicional acerca del indicador de salud. |

HEALTH_DBM_HI_HIS

La función de tabla HEALTH_DBM_HI_HIS devuelve información histórica del indicador de salud desde una instantánea de la salud del gestor de bases de datos DB2.

Importante: Esta función de tabla ha quedado en desuso y puede que se elimine en un futuro release ya que el Supervisor de salud ha quedado en desuso en la Versión 9.7. Para obtener más información, consulte el tema “El Supervisor de salud ha quedado en desuso” de la publicación *Novedades en DB2 Versión 9.7*.

Sintaxis

►►—HEALTH_DBM_HI_HIS—(—númparticiónbd—)—————►►

El esquema es SYSPROC.

Parámetro de la función de tabla

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE para la función de tabla HEALTH_DBM_HI_HIS.

Ejemplo

```
SELECT * FROM TABLE(HEALTH_DBM_HI_HIS(-1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
SNAPSHOT_TIMESTAMP      HI_ID      SERVER_INSTANCE_NAME    HI_VALUE ...
-----
2006-02-13-12.30.20.460905    1 DB2      0 ...
2006-02-13-12.30.20.460905    1 DB2      0 ...
2006-02-13-12.30.20.460905    1 DB2      0 ...
2006-02-13-12.30.20.460905    1 DB2      0 ...
2006-02-13-12.30.20.460905    1 DB2      0 ...
2006-02-13-12.30.20.460905    1 DB2      0 ...
```

| | | |
|----------------------------|-------|---------|
| 2006-02-13-12.30.20.460905 | 1 DB2 | 0 ... |
| 2006-02-13-12.30.20.460905 | 1 DB2 | 0 ... |
| 2006-02-13-12.30.20.460905 | 1 DB2 | 0 ... |
| 2006-02-13-12.30.20.460905 | 4 DB2 | 100 ... |
| 2006-02-13-12.30.20.460905 | 4 DB2 | 100 ... |
| 2006-02-13-12.30.20.460905 | 4 DB2 | 100 ... |
| 2006-02-13-12.30.20.460905 | 4 DB2 | 100 ... |
| 2006-02-13-12.30.20.460905 | 4 DB2 | 60 ... |
| 2006-02-13-12.30.20.460905 | 4 DB2 | 60 ... |
| 2006-02-13-12.30.20.460905 | 4 DB2 | 60 ... |
| 2006-02-13-12.30.20.460905 | 4 DB2 | 60 ... |
| 2006-02-13-12.30.20.460905 | 4 DB2 | 60 ... |

18 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

| ... HI_TIMESTAMP | HI_ALERT_STATE | HI_ALERT_STATE_DETAIL | ... |
|--------------------------------|----------------|-----------------------|-----|
| ... | ----- | ----- | ... |
| ... 2006-02-13-12.21.25.649000 | 1 Normal | | ... |
| ... 2006-02-13-12.16.25.911000 | 1 Normal | | ... |
| ... 2006-02-13-12.11.25.377000 | 1 Normal | | ... |
| ... 2006-02-13-12.06.26.168000 | 1 Normal | | ... |
| ... 2006-02-13-12.01.25.165000 | 1 Normal | | ... |
| ... 2006-02-13-11.56.25.927000 | 1 Normal | | ... |
| ... 2006-02-13-11.51.25.452000 | 1 Normal | | ... |
| ... 2006-02-13-11.46.25.211000 | 1 Normal | | ... |
| ... 2006-02-13-11.41.25.972000 | 1 Normal | | ... |
| ... 2006-02-13-12.21.25.649000 | 4 Alarm | | ... |
| ... 2006-02-13-12.16.25.911000 | 4 Alarm | | ... |
| ... 2006-02-13-12.11.25.377000 | 4 Alarm | | ... |
| ... 2006-02-13-12.06.26.168000 | 4 Alarm | | ... |
| ... 2006-02-13-12.01.25.165000 | 1 Normal | | ... |
| ... 2006-02-13-11.56.25.927000 | 1 Normal | | ... |
| ... 2006-02-13-11.51.25.452000 | 1 Normal | | ... |
| ... 2006-02-13-11.46.25.211000 | 1 Normal | | ... |
| ... 2006-02-13-11.41.25.972000 | 1 Normal | | ... |

Datos de salida de esta consulta (continuación).

| ... HI_FORMULA | HI_ADDITIONAL_INFO |
|-------------------------------|--------------------|
| ... | ----- |
| ... 0 | - |
| ... 0 | - |
| ... 0 | - |
| ... 0 | - |
| ... 0 | - |
| ... 0 | - |
| ... 0 | - |
| ... 0 | - |
| ... 0 | - |
| ... 0 | - |
| ... ((327680 / 327680) * 100) | - |
| ... ((327680 / 327680) * 100) | - |
| ... ((327680 / 327680) * 100) | - |
| ... ((327680 / 327680) * 100) | - |
| ... ((196608 / 327680) * 100) | - |
| ... ((196608 / 327680) * 100) | - |
| ... ((196608 / 327680) * 100) | - |
| ... ((196608 / 327680) * 100) | - |
| ... ((196608 / 327680) * 100) | - |

Información devuelta

Tabla 244. Información devuelta por la función de tabla HEALTH_DBM_HI_HIS

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| HI_ID | BIGINT | Número que identifica de forma exclusiva el indicador de salud en la corriente de datos de la instantánea. |
| SERVER_INSTANCE_NAME | VARCHAR(128) | server_instance_name - Nombre de instancia del servidor |
| HI_VALUE | SMALLINT | Valor del indicador de salud. |
| HI_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se generó la alerta. |
| HI_ALERT_STATE | BIGINT | La gravedad de la alerta. |
| HI_ALERT_STATE_DETAIL | VARCHAR(20) | Descripción textual de la columna HI_ALERT_STATE. |
| HI_FORMULA | VARCHAR(2048) | Fórmula utilizada para calcular el indicador de salud. |
| HI_ADDITIONAL_INFO | VARCHAR(4096) | Información adicional acerca del indicador de salud. |

HEALTH_DBM_INFO

La función HEALTH_DBM_INFO devuelve información desde una instantánea de la salud del gestor de bases de datos DB2.

Importante: Esta función de tabla ha quedado en desuso y puede que se elimine en un futuro release ya que el Supervisor de salud ha quedado en desuso en la Versión 9.7. Para obtener más información, consulte el tema “El Supervisor de salud ha quedado en desuso” de la publicación *Novedades en DB2 Versión 9.7*.

Sintaxis

►►—HEALTH_DBM_INFO—(—númparticiónbd—)—————►►

El esquema es SYSPROC.

Parámetro de la función de tabla

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE para la función de tabla HEALTH_DBM_INFO.

Ejemplo

```
SELECT * FROM TABLE(HEALTH_DBM_INFO(-1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
SNAPSHOT_TIMESTAMP          SERVER_INSTANCE_NAME      ROLLED_UP_ALERT_STATE    ...
-----
2006-02-13-12.30.19.663924  DB2                        4                          ...
```

1 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```
... ROLLED_UP_ALERT_STATE_DETAIL DB2START_TIME      ...
... -----
... Alarm                          2006-02-09-10.56.18.126182 ...
```

Datos de salida de esta consulta (continuación).

```
... LAST_RESET          NUM_NODES_IN_DB2_INSTANCE
... -----
... -                      1
```

Información devuelta

Tabla 245. Información devuelta por la función de tabla HEALTH_DBM_INFO

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| SERVER_INSTANCE_NAME | VARCHAR(128) | server_instance_name - Nombre de instancia del servidor |
| ROLLED_UP_ALERT_STATE | BIGINT | Estado de alerta más grave capturado por esta instantánea. |
| ROLLED_UP_ALERT_STATE_DETAIL | VARCHAR(20) | Descripción textual de la columna ROLLED_UP_ALERT_STATE. |
| DB2START_TIME | TIMESTAMP | db2start_time - Indicación de fecha y hora de inicio de gestor de bases de datos |
| LAST_RESET | TIMESTAMP | last_reset - Indicación de fecha y hora de último restablecimiento |
| NUM_NODES_IN_DB2_INSTANCE | INTEGER | num_nodes_in_db2_instance - Número de nodos en la partición de base de datos |

HEALTH_GET_ALERT_ACTION_CFG

Devuelve valores de configuración de acciones de alerta de salud correspondientes a diversos tipos de objetos (gestor de bases de datos, base de datos, espacio de

tablas y contenedor de espacios de tablas) y a diversos niveles de configuración (valor por omisión de instalación, instancia, global y objeto).

Importante: Esta función de tabla ha quedado en desuso y puede que se elimine en un futuro release ya que el Supervisor de salud ha quedado en desuso en la Versión 9.7. Para obtener más información, consulte el tema “El Supervisor de salud ha quedado en desuso” de la publicación *Novedades en DB2 Versión 9.7*.

Sintaxis

```
►►—HEALTH_GET_ALERT_ACTION_CFG—(—tipobjeto—,—nivel_cfg—,—nombre-bd—,—►►  
►—nombreobjeto—)———►►
```

El esquema es SYSPROC.

Parámetros de la función de tabla

tipobjeto

Argumento de entrada de tipo VARCHAR(3) que indica el tipo de objeto. El valor debe ser uno de los siguientes valores sensibles a mayúsculas y minúsculas:

- 'DBM' para gestor de bases de datos
- 'DB' para base de datos
- 'TS' para espacio de tablas
- 'TSC' para contenedor de espacio de tablas

Nota: Los espacios iniciales y finales se pasarán por alto.

nivel_cfg

Argumento de entrada de tipo VARCHAR(1) que indica el nivel de configuración. El valor debe ser uno de los siguientes valores sensibles a mayúsculas y minúsculas:

- Para *tipobjeto* 'DBM': 'D' para valor por omisión de la instalación; 'G' u 'O' para nivel de instancia.
- Para *tipobjeto* que no sea 'DBM': 'D' para valor por omisión de la instalación; 'G' para nivel global; 'O' para nivel de objeto.

nombre-bd

Argumento de entrada de tipo VARCHAR(128) que indica el nombre de la base de datos. El nombre de la base de datos se debe especificar si *tipobjeto* es 'DB', 'TS' o 'TSC' y *nivel_cfg* es 'O'. Para las demás combinaciones de *tipobjeto* y *nivel_cfg*, el parámetro *nombrebd* debe ser nulo (o una serie vacía).

nombreobjeto

Argumento de entrada de tipo VARCHAR(1024) que indica el nombre del objeto, por ejemplo, <nombre espacio tabla> o <nombre espacio tabla>.<nombre contenedor>. El nombre del objeto se debe especificar si *tipobjeto* es 'TS' o 'TSC' y *nivel_cfg* es 'O'. Para las demás combinaciones de *tipobjeto* y *nivel_cfg*, el parámetro *nombreobjeto* debe ser nulo (o una serie vacía).

Autorización

Privilegio EXECUTE sobre la función de tabla
HEALTH_GET_ALERT_ACTION_CFG.

Ejemplos

Ejemplo 1: Recuperar valores de configuración de acciones de alerta de nivel de objeto para la base de datos SAMPLE para el ID de indicador 1004.

```
SELECT OBJECTTYPE, CFG_LEVEL, SUBSTR(DBNAME,1,8) AS DBNAME,
       SUBSTR(OBJECTNAME,1,8) AS OBJECTNAME, ID, IS_DEFAULT,
       SUBSTR(CONDITION,1,10) AS CONDITION, ACTIONTYPE,
       SUBSTR(ACTIONNAME,1,30) AS ACTIONNAME, SUBSTR(USERID,1,8) AS USERID,
       SUBSTR(HOSTNAME,1,10) AS HOSTNAME, SCRIPT_TYPE,
       SUBSTR(WORKING_DIR,1,10) AS WORKING_DIR, TERMINATION_CHAR,
       SUBSTR(PARAMETERS,1,10) AS PARAMETERS
FROM TABLE(HEALTH_GET_ALERT_ACTION_CFG('DB','0','SAMPLE','')) AS ACTION_CFG
WHERE ID = 1004
```

El siguiente ejemplo muestra los datos de salida de la consulta.

| OBJECTTYPE | CFG_LEVEL | DBNAME | OBJECTNAME | ID | IS_DEFAULT | CONDITION |
|------------|-----------|--------|------------|------|------------|-----------|
| DB | 0 | SAMPLE | | 1004 | 1 | ALARM |
| DB | 0 | SAMPLE | | 1004 | 1 | ALARM |

2 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

| ACTIONTYPE | ACTIONNAME | USERID | HOSTNAME |
|------------|-------------------------------|--------|----------|
| S | ~/health_center/script/scrpn6 | uid1 | - |
| T | 00.0005 | uid1 | HOST3 |

Datos de salida de esta consulta (continuación).

| SCRIPT_TYPE | WORKING_DIR | TERMINATION_CHAR | PARAMETERS |
|-------------|-------------|------------------|------------|
| 0 | ~/health_c | - | - |
| - | - | - | - |

Ejemplo 2: Recuperar la condición, el tipo de acción, el nombre de acción, el nombre de sistema principal y el tipo de script para la base de datos SAMPLE para el ID de indicador 1004.

```
SELECT CONDITION, ACTIONTYPE, SUBSTR(ACTIONNAME,1,35) AS ACTIONNAME,
       SUBSTR(USERID,1,8) AS USERID, SUBSTR(HOSTNAME,1,10) AS HOSTNAME, SCRIPT_TYPE
FROM TABLE(HEALTH_GET_ALERT_ACTION_CFG('DB','0','SAMPLE','')) AS ALERT_ACTION_CFG
WHERE ID=1004
```

El siguiente ejemplo muestra los datos de salida de la consulta.

| CONDITION | ACTIONTYPE | ACTIONNAME |
|-----------|------------|-------------------------------|
| ALARM | S | ~/health_center/script/scrpn6 |
| ALARM | T | 00.0005 |

2 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

| USERID | HOSTNAME | SCRIPT_TYPE |
|--------|----------|-------------|
| uid1 | - | 0 |
| uid1 | HOST3 | - |

Notas de uso

La función de tabla HEALTH_GET_IND_DEFINITION se puede utilizar para correlacionar ID de indicadores de salud con nombres de indicadores de salud.

Información devuelta

Tabla 246. Información devuelta por la función de tabla HEALTH_GET_ALERT_ACTION_CFG

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|--|
| OBJECTTYPE | VARCHAR(3) | Tipo de objeto. |
| CFG_LEVEL | CHAR(1) | Nivel de configuración. |
| DBNAME | VARCHAR(128) | Nombre de la base de datos. |
| OBJECTNAME | VARCHAR(512) | Nombre del objeto. |
| ID | BIGINT | ID del indicador de salud. |
| IS_DEFAULT | SMALLINT | Si el valor es el valor por omisión: 1 si es el valor por omisión, 0 si no es el valor por omisión, nulo si no se aplica. |
| CONDITION | VARCHAR(512) | Condición de alerta sobre la que se ha activado la acción. |
| ACTIONTYPE | CHAR(1) | Tipo de acción: 'S' para acción de script o 'T' para acción de tarea. |
| ACTIONNAME | VARCHAR(5000) | Si ACTIONTYPE es 'S', es el nombre de la vía de acceso del script. Si ACTIONTYPE es 'T', es el ID de tarea. |
| USERID | VARCHAR(1024) | Nombre del usuario bajo el que se ejecutará la acción. |
| HOSTNAME | VARCHAR(255) | Nombre del sistema principal. |
| SCRIPT_TYPE | CHAR(1) | Tipo de script: si ACTIONTYPE es 'S', 'O' para script de mandatos del sistema operativo o 'D' para scripts de mandatos de DB2; si ACTIONTYPE es 'T', nulo. |
| WORKING_DIR | VARCHAR(5000) | El directorio de trabajo correspondiente al script si ACTIONTYPE es 'S' o nulo si ACTIONTYPE es 'T'. |
| TERMINATION_CHAR | VARCHAR(4) | El carácter de terminación de la sentencia si se trata de una acción de script de mandatos de DB2; nulo en caso contrario. |
| PARAMETERS | VARCHAR(200) | Los parámetros de línea de mandatos si se trata de una acción de script de mandatos del sistema operativo. |

HEALTH_GET_ALERT_CFG

Devuelve valores de configuración de alerta de salud correspondientes a diversos tipos de objetos (gestor de bases de datos, base de datos, espacio de tablas, contenedor de espacios de tablas) y a diversos niveles de configuración (valor por omisión de instalación, global y objeto).

Importante: Esta función de tabla ha quedado en desuso y puede que se elimine en un futuro release ya que el Supervisor de salud ha quedado en desuso en la Versión 9.7. Para obtener más información, consulte el tema “El Supervisor de salud ha quedado en desuso” de la publicación *Novedades en DB2 Versión 9.7*.

Sintaxis

```
►►—HEALTH_GET_ALERT_CFG—(—tipoobjeto—,—nivel_cfg—,—nombre-bd—,——————►  
►—nombreobjeto—)—————►◄◄
```

El esquema es SYSPROC.

Parámetros de la función de tabla

tipoobjeto

Argumento de entrada de tipo VARCHAR(3) que indica el tipo de objeto. El valor debe ser uno de los siguientes valores sensibles a mayúsculas y minúsculas:

- 'DBM' para gestor de bases de datos
- 'DB' para base de datos
- 'TS' para espacio de tablas
- 'TSC' para contenedor de espacio de tablas

Nota: Los espacios iniciales y finales se pasarán por alto.

nivel_cfg

Argumento de entrada de tipo VARCHAR(1) que indica el nivel de configuración. El valor debe ser uno de los siguientes valores sensibles a mayúsculas y minúsculas:

- Para *tipoobjeto* 'DBM': 'D' para valor por omisión de la instalación; 'G' u 'O' para nivel de instancia.
- Para *tipoobjeto* que no sea 'DBM': 'D' para valor por omisión de la instalación; 'G' para nivel global; 'O' para nivel de objeto.

nombre-bd

Argumento de entrada de tipo VARCHAR(128) que indica el nombre de la base de datos. El nombre de la base de datos se debe especificar si *tipoobjeto* es 'DB', 'TS' o 'TSC' y *nivel_cfg* es 'O'. Para las demás combinaciones de *tipoobjeto* y *nivel_cfg*, el parámetro *nombrebd* debe ser nulo (o una serie vacía).

nombreobjeto

Argumento de entrada de tipo VARCHAR(1024) que indica el nombre del objeto, por ejemplo, <nombre espacio tabla> o <nombre espacio tabla>.<nombre contenedor>. El nombre del objeto se debe especificar si


```

...           85           0           1           0
...           10           0           1           0
...           85           0           1           0
...           10           0           1           0
...           70           0           1           0
...           70           0           0           0

```

Ejemplo 2: Recuperar los umbrales de aviso y de alarma para el indicador de salud con ID '2002' para el espacio de tablas USERSPACE1 de la base de datos SAMPLE.

```

SELECT WARNING_THRESHOLD, ALARM_THRESHOLD
  FROM TABLE(SYSPROC.HEALTH_GET_ALERT_CFG('TS','0','SAMPLE','USERSPACE1'))
  AS T WHERE ID = 2002

```

El siguiente ejemplo muestra los datos de salida de la consulta.

```

WARNING_THRESHOLD  ALARM_THRESHOLD
-----
                        80                        90

```

SQL22004N No se puede encontrar la configuración solicitada para el objeto dado. Se devuelve la configuración por omisión para "espaciostabla".

1 record(s) selected with 1 warning messages printed.

Notas de uso

La función de tabla HEALTH_GET_IND_DEFINITION se puede utilizar para correlacionar ID de indicadores de salud con nombres de indicadores de salud.

Ejemplo: Recuperar los umbrales de aviso y de alerta para el indicador de salud Utilización de espacio de tablas (ts.ts_util) para el espacio de tablas USERSPACE1 de la base de datos SAMPLE.

```

WITH HINAME(ID) AS (SELECT ID FROM TABLE(SYSPROC.HEALTH_GET_IND_DEFINITION('')) AS W
  WHERE NAME = 'ts.ts_util')
SELECT WARNING_THRESHOLD, ALARM_THRESHOLD
  FROM TABLE(SYSPROC.HEALTH_GET_ALERT_CFG('TS','0','SAMPLE','USERSPACE1')) AS T,
  HINAME AS H
  WHERE T.ID = H.ID

```

El siguiente ejemplo muestra los datos de salida de la consulta.

```

WARNING_THRESHOLD  ALARM_THRESHOLD
-----
                        80                        90

```

SQL22004N No se puede encontrar la configuración solicitada para el objeto dado. Se devuelve la configuración por omisión para "espaciostabla".

1 record(s) selected with 1 warning messages printed.

Información devuelta

Tabla 247. Información devuelta por la función de tabla HEALTH_GET_ALERT_CFG

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|-----------------------------|
| OBJECTTYPE | VARCHAR(3) | Tipo de objeto. |
| CFG_LEVEL | VARCHAR(1) | Nivel de configuración. |
| DBNAME | VARCHAR(128) | Nombre de la base de datos. |
| OBJECTNAME | VARCHAR(512) | Nombre del objeto. |
| ID | BIGINT | ID del indicador de salud. |

Tabla 247. Información devuelta por la función de tabla HEALTH_GET_ALERT_CFG (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| IS_DEFAULT | SMALLINT | Si el valor es el valor por omisión: 1 si es el valor por omisión, 0 si no es el valor por omisión o nulo si no se aplica. |
| WARNING_THRESHOLD | BIGINT | Umbral de aviso. Nulo si no se aplica. |
| ALARM_THRESHOLD | BIGINT | Umbral de alarma. Nulo si no se aplica. |
| SENSITIVITY | BIGINT | Sensibilidad del indicador de salud. |
| EVALUATE | SMALLINT | 1 si este indicador de salud se va a evaluar o 0 si no se va a evaluar. |
| ACTION_ENABLED | SMALLINT | 1 si se ha habilitado una acción para que se ejecute tras la aparición de una alerta o 0 si no se ha habilitado ninguna acción para que se ejecute tras la aparición de una alerta. |

HEALTH_GET_IND_DEFINITION

Devuelve las definiciones de indicador de salud.

Importante: Esta función de tabla ha quedado en desuso y puede que se elimine en un futuro release ya que el Supervisor de salud ha quedado en desuso en la Versión 9.7. Para obtener más información, consulte el tema “El Supervisor de salud ha quedado en desuso” de la publicación *Novedades en DB2 Versión 9.7*.

Sintaxis

►►—HEALTH_GET_IND_DEFINITION—(—*entorno-local*—)————►►

El esquema es SYSPROC.

Parámetro de la función de tabla

entorno-local

Argumento de entrada de tipo VARCHAR(33) que indica el entorno local en el que se deben devolver los datos de salida que se pueden traducir. Si el entorno local de entrada no recibe soporte del servidor de bases de datos, se emite un mensaje de aviso de SQL y se utiliza el idioma por omisión (inglés). Si no se especifica el entorno local de entrada, es decir, su valor es nulo (o una serie vacía), se utiliza el idioma por omisión.

Autorización

Privilegio EXECUTE sobre la función de tabla HEALTH_GET_IND_DEFINITION.

Ejemplos

Ejemplo 1: Recuperar el tipo y la descripción abreviada del indicador de salud db.db_op_status en francés.

```
SELECT TYPE, SHORT_DESCRIPTION
      FROM TABLE(SYSPROC.HEALTH_GET_IND_DEFINITION('fr_FR'))
      AS IND_DEFINITION WHERE NAME = 'db.db_op_status'
```

El siguiente ejemplo muestra los datos de salida de la consulta.

```
TYPE          SHORT_DESCRIPTION
-----
STATE         Etat opérationnel de la base de données
```

1 registro(s) seleccionado(s).

Ejemplo 2: Recuperar la descripción abreviada del indicador de salud con ID 1001 en inglés.

```
SELECT SHORT_DESCRIPTION FROM TABLE(SYSPROC.HEALTH_GET_IND_DEFINITION('en_US'))
      AS IND_DEFINITION WHERE ID = 1001
```

El siguiente ejemplo muestra los datos de salida de la consulta.

```
SHORT_DESCRIPTION
-----
Database Operational State
```

Ejemplo 3: Recuperar todos los ID de indicadores de salud y sus nombres.

```
SELECT ID, NAME FROM TABLE(HEALTH_GET_IND_DEFINITION('')) AS T
```

El siguiente ejemplo muestra los datos de salida de la consulta.

```
ID          NAME
-----
1 db2.db2_op_status
2 db2.sort_privmem_util
4 db2.mon_heap_util
1001 db.db_op_status
1002 db.sort_shrmem_util
...
2001 ts.ts_op_status
2002 ts.ts_util
...
3002 tsc.tscont_util
1015 db.tb_reorg_req
...
```

Información devuelta

Tabla 248. Información devuelta por la función de tabla HEALTH_GET_IND_DEFINITION

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|----------------|---|
| ID | BIGINT | ID del indicador de salud. |
| NAME | VARCHAR(128) | Nombre del indicador de salud. |
| SHORT_DESCRIPTION | VARCHAR(1024) | Descripción abreviada del indicador de salud. |
| LONG_DESCRIPTION | VARCHAR(32672) | Descripción larga del indicador de salud. |

Tabla 248. Información devuelta por la función de tabla HEALTH_GET_IND_DEFINITION (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| TYPE | VARCHAR(16) | Tipo de indicador de salud. Los valores posibles son: <ul style="list-style-type: none"> 'THRESHOLD_UPPER': indicadores de salud basados en umbral superior. 'THRESHOLD_LOWER': indicadores de salud basados en umbral inferior. 'STATE': indicadores de salud basados en estado. 'COLLECTION_STATE': indicadores de salud basados en estado de recopilación. |
| UNIT | VARCHAR(1024) | Unidad de los valores de indicadores de salud y de los umbrales o nulo si no se aplica. |
| CATEGORY | VARCHAR(1024) | Categoría del indicador de salud. |
| FORMULA | VARCHAR(512) | Fórmula del indicador de salud. |
| REFRESH_INTERVAL | BIGINT | Intervalo de evaluación del indicador de salud en segundos. |

RHEALTH_HI_REC

Recupera un conjunto de recomendaciones que se ocupan de un indicador de salud en estado de alerta en un objeto DB2 determinado. Las recomendaciones se devuelven en un documento XML que contiene información acerca de acciones que se pueden emprender (por ejemplo, scripts que se pueden ejecutar) para resolver el estado de alerta.

Importante: Este procedimiento ha quedado en desuso y se puede eliminar en un futuro release ya que el Supervisor de salud ha quedado en desuso en la Versión 9.7. Para obtener más información, consulte el tema "El Supervisor de salud ha quedado en desuso" del manual *Novedades en la Versión 9.7*.

Sintaxis

```

▶▶—HEALTH_HI_REC—(—versión-esquema—,—id-indicador—,—nombre-bd—,—
▶—tipo-objeto—,—nombre-objeto—,—núm-partición-bd—,—
▶—entorno-local-cliente—,—doc-recomendaciones—)▶▶▶▶

```

El esquema es SYSPROC.

Los scripts devueltos por este procedimiento se deben invocar desde la instancia donde el indicador de salud ha entrado en el estado de alerta.

Si el indicador de salud que se ha especificado sobre el objeto identificado no se encuentra en estado de alerta, se devuelve un error (SQLSTATE 5U0ZZ).

Parámetros del procedimiento

versión-esquema

Argumento de entrada de tipo INTEGER que especifica el ID de versión del esquema utilizado para representar el documento XML. El documento de recomendaciones sólo contendrá elementos y atributos definidos para esa versión de esquema. Las versiones de esquema válidas están definidas en `db2ApiDf.h`, ubicado en el subdirectorio `include` del directorio `sqllib`.

id-indicador

Argumento de entrada de tipo INTEGER que especifica el identificador numérico del indicador de salud para el que se solicitan recomendaciones. Los ID de indicador de salud válidos están definidos en `sqlmon.h`, ubicado en el subdirectorio `include` del directorio `sqllib`.

nombrebd

Argumento de entrada de tipo VARCHAR(255) que especifica un nombre de alias de la base de datos para la que el indicador de salud ha entrado en estado de alerta y cuando el tipo de objeto es DB2HEALTH_OBJTYPE_TS_CONTAINER, DB2HEALTH_OBJTYPE_TABLESPACE o DB2HEALTH_OBJTYPE_DATABASE. De lo contrario, especifique NULL.

tipo-objeto

Argumento de entrada de tipo INTEGER que especifica el tipo de objeto por el cual el indicador de salud ha entrado en estado de alerta. Los tipos de objetos válidos están definidos en `sqlmon.h`, que se ubica en el subdirectorio `include` del directorio `sqllib`.

nombre-objeto

Argumento de entrada de tipo VARCHAR(255) que especifica el nombre de un espacio de tablas o de contenedor de espacio de tablas cuando el tipo de objeto se establece en DB2HEALTH_OBJTYPE_TABLESPACE o DB2HEALTH_OBJTYPE_TS_CONTAINER. Especifique NULL si el tipo de objeto es DB2HEALTH_OBJTYPE_DATABASE o DB2HEALTH_OBJTYPE_DATABASE_MANAGER. En el caso de un contenedor de espacio de tablas, el nombre de objeto se especifica como `nombre_espacio_tabla.nombre_contenedor`.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica el número de la partición de base de datos en la que el indicador de salud ha entrado en estado de alerta. Los valores válidos son 0 a 999, -1 (que especifica la partición de base de datos conectada actualmente) y -2 (que especifica todas las particiones de base de datos activas). Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

entorno-local-cliente

Argumento de entrada de tipo VARCHAR(33) que especifica el identificador del idioma del cliente. Utilice este parámetro para especificar el idioma en que se deben devolver las recomendaciones. Si no se especifica ningún valor, se utilizará 'En_US' (inglés). Tenga en cuenta que, si los archivos de mensajes del entorno local especificado no están disponibles en el servidor, se utilizará 'En_US' como valor por omisión.

doc-recomendaciones

Argumento de salida de tipo BLOB(2M) que contiene el documento de recomendación (XML), formateado de acuerdo con la definición de esquema de Recomendación de salud de DB2 (consulte el esquema XML DB2RecommendationSchema.xsd, ubicado en el subdirectorio misc del directorio sql1ib). El documento XML está codificado en UTF-8 y el texto del documento tiene el entorno local del llamante, o se encuentra en inglés si los mensajes no están disponibles en el entorno local del llamante en la instancia de destino.

HEALTH_TBS_HI

Devuelve información de indicador de salud para espacios de tablas de una instantánea de salud de los espacios de tablas de una base de datos.

Importante: Esta función de tabla ha quedado en desuso y puede que se elimine en un futuro release ya que el Supervisor de salud ha quedado en desuso en la Versión 9.7. Para obtener más información, consulte el tema “El Supervisor de salud ha quedado en desuso” de la publicación *Novedades en DB2 Versión 9.7*.

Sintaxis

▶▶—HEALTH_TBS_HI—(—*nombrebd*—,—*númparticiónbd*—)————▶▶

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(255) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique el valor nulo para tomar la instantánea desde la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE para la función de tabla HEALTH_TBS_HI.

Ejemplo

```
SELECT * FROM TABLE(HEALTH_TBS_HI('',-1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| SNAPSHOT_TIMESTAMP | TABLESPACE_NAME | HI_ID | HI_VALUE | ... |
|----------------------------|-----------------|-------|----------|-----|
| 2006-02-13-12.30.35.229196 | SYSCATSPACE | 2001 | 0 | ... |
| 2006-02-13-12.30.35.229196 | SYSCATSPACE | 2002 | 99 | ... |
| 2006-02-13-12.30.35.229196 | SYSCATSPACE | 2003 | 0 | ... |

| | | | |
|----------------------------|------------------|------|---------|
| 2006-02-13-12.30.35.229196 | SYSTOOLSPACE | 2001 | 0 ... |
| 2006-02-13-12.30.35.229196 | SYSTOOLSPACE | 2002 | 62 ... |
| 2006-02-13-12.30.35.229196 | SYSTOOLSPACE | 2003 | 0 ... |
| 2006-02-13-12.30.35.229196 | SYSTOOLSTMPSPACE | 2001 | 0 ... |
| 2006-02-13-12.30.35.229196 | TEMPSPACE1 | 2001 | 0 ... |
| 2006-02-13-12.30.35.229196 | USERSPACE1 | 2001 | 0 ... |
| 2006-02-13-12.30.35.229196 | USERSPACE1 | 2002 | 100 ... |
| 2006-02-13-12.30.35.229196 | USERSPACE1 | 2003 | 0 ... |

11 registro(s) seleccionados.

Datos de salida de esta consulta (continuación).

| ... | HI_TIMESTAMP | HI_ALERT_STATE | HI_ALERT_STATE_DETAIL | ... |
|-----|----------------------------|----------------|-----------------------|-----|
| ... | ----- | ----- | ----- | ... |
| ... | 2006-02-13-12.26.26.158000 | 1 | Normal | ... |
| ... | 2006-02-13-12.26.26.158000 | 4 | Alarm | ... |
| ... | 2006-02-13-12.26.26.158000 | 1 | Normal | ... |
| ... | 2006-02-13-12.26.26.158000 | 1 | Normal | ... |
| ... | 2006-02-13-12.26.26.158000 | 1 | Normal | ... |
| ... | 2006-02-13-12.26.26.158000 | 1 | Normal | ... |
| ... | 2006-02-13-12.26.26.158000 | 1 | Normal | ... |
| ... | 2006-02-13-12.26.26.158000 | 1 | Normal | ... |
| ... | 2006-02-13-12.26.26.158000 | 4 | Alarm | ... |
| ... | 2006-02-13-12.26.26.158000 | 1 | Normal | ... |

Datos de salida de esta consulta (continuación).

| ... | HI_FORMULA | HI_ADDITIONAL_INFO |
|-----|-----------------------|---|
| ... | ----- | ----- |
| ... | 0 | - |
| ... | ((9376 / 9468) * 100) | The short term table space growth rate from "02/13/2006 11:26:26.000158" to "02/13/2006 12:26:26.000158" is "N/A" bytes per second and the long term growth rate from "02/12/2006 12:26:26.000158" to "02/13/2006 12:26:26.000158" is "N/A" bytes per second. Time to fullness is projected to be "N/A" and "N/A" respectively. The table space is defined with automatic storage set to "YES" and automatic resize enabled set to "YES". |
| ... | 0 | The table space is defined with automatic storage set to "YES" and automatic resize enabled set to "YES". The following are the automatic resize settings: increase size (bytes) "-1", increase size (percent) "N/A", maximum size (bytes) "-1". Las opciones current table space size (bytes) is "38797312". |
| ... | 0 | - |
| ... | ((156 / 252) * 100) | The short term table space growth rate from "02/13/2006 11:26:26.000158" to "02/13/2006 12:26:26.000158" is "N/A" bytes per second and the long term growth rate from "02/12/2006 12:26:26.000158" to "02/13/2006 12:26:26.000158" is "N/A" bytes per second. Time to fullness is projected to be "N/A" and "N/A" respectively. The table space is defined with automatic storage set to "YES" and automatic resize enabled set to "YES". |
| ... | 0 | The table space is defined with automatic storage set to "YES" and automatic resize enabled set to "YES". The following are the automatic resize settings: increase size (bytes) "-1", increase size (percent) |


```

...                               0           -
...                               0           -
...                               0           -
... ((1504 / 1504) * 100)       The short term table space growth rate from
                                "02/13/2006 11:26:26.000158" to
                                "02/13/2006 12:26:26.000158" is "N/A"
                                bytes per second and the long term growth
                                rate from "02/12/2006 12:26:26.000158" to
                                "02/13/2006 12:26:26.000158" is "N/A" bytes
                                per second. Time to fullness is projected
                                to be "N/A" and "N/A" respectively. Las opciones
                                table space is defined with automatic storage
                                set to "YES" and automatic resize enabled
                                set to "YES".
... 0                             The table space is defined with automatic
                                storage set to "YES" and automatic resize
                                enabled set to "YES". The following are
                                the automatic resize settings: increase
                                size (bytes) "-1", increase size (percent)
                                "N/A", maximum size (bytes) "-1". Las opciones
                                current table space size (bytes) is
                                "6291456".

```

Información devuelta

Tabla 249. Información devuelta por la función de tabla HEALTH_TBS_HI

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| TABLESPACE_NAME | VARCHAR(128) | tablespace_name - Nombre de espacio de tablas |
| HI_ID | BIGINT | Número que identifica de forma exclusiva el indicador de salud en la corriente de datos de la instantánea. |
| HI_VALUE | SMALLINT | Valor del indicador de salud. |
| HI_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se generó la alerta. |
| HI_ALERT_STATE | BIGINT | La gravedad de la alerta. |
| HI_ALERT_STATE_DETAIL | VARCHAR(20) | Descripción textual de la columna HI_ALERT_STATE. |
| HI_FORMULA | VARCHAR(2048) | Fórmula utilizada para calcular el indicador de salud. |
| HI_ADDITIONAL_INFO | VARCHAR(4096) | Información adicional acerca del indicador de salud. |

HEALTH_TBS_HI_HIS

La función de tabla HEALTH_TBS_HI_HIS devuelve información del histórico del indicador de salud correspondiente a espacios de tablas a partir de una instantánea de la salud de una base de datos.

Importante: Esta función de tabla ha quedado en desuso y puede que se elimine en un futuro release ya que el Supervisor de salud ha quedado en desuso en la Versión 9.7. Para obtener más información, consulte el tema “El Supervisor de salud ha quedado en desuso” de la publicación *Novedades en DB2 Versión 9.7*.

Sintaxis

►—HEALTH_TBS_HI_HIS—(—nombre-bd—,—númparticiónbd—)—————►

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(255) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique el valor nulo para tomar la instantánea desde la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE para la función de tabla HEALTH_TBS_HI_HIS.

Ejemplo

```
SELECT * FROM TABLE(HEALTH_TBS_HI_HIS('',-1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| SNAPSHOT_TIMESTAMP | TABLESPACE_NAME | HI_ID | ... |
|----------------------------|------------------|-------|-----|
| 2006-02-13-12.30.37.181478 | SYSCATSPACE | 2001 | ... |
| 2006-02-13-12.30.37.181478 | SYSCATSPACE | 2001 | ... |
| 2006-02-13-12.30.37.181478 | SYSCATSPACE | 2002 | ... |
| 2006-02-13-12.30.37.181478 | SYSCATSPACE | 2002 | ... |
| 2006-02-13-12.30.37.181478 | SYSCATSPACE | 2003 | ... |
| 2006-02-13-12.30.37.181478 | SYSCATSPACE | 2003 | ... |
| 2006-02-13-12.30.37.181478 | SYSTOOLSPACE | 2001 | ... |
| 2006-02-13-12.30.37.181478 | SYSTOOLSPACE | 2001 | ... |
| 2006-02-13-12.30.37.181478 | SYSTOOLSPACE | 2002 | ... |
| 2006-02-13-12.30.37.181478 | SYSTOOLSPACE | 2002 | ... |
| 2006-02-13-12.30.37.181478 | SYSTOOLSPACE | 2003 | ... |
| 2006-02-13-12.30.37.181478 | SYSTOOLSPACE | 2003 | ... |
| 2006-02-13-12.30.37.181478 | SYSTOOLSTMPSPACE | 2001 | ... |
| 2006-02-13-12.30.37.181478 | SYSTOOLSTMPSPACE | 2001 | ... |
| 2006-02-13-12.30.37.181478 | TEMPSPACE1 | 2001 | ... |
| 2006-02-13-12.30.37.181478 | TEMPSPACE1 | 2001 | ... |
| 2006-02-13-12.30.37.181478 | USERSPACE1 | 2001 | ... |
| 2006-02-13-12.30.37.181478 | USERSPACE1 | 2001 | ... |
| 2006-02-13-12.30.37.181478 | USERSPACE1 | 2002 | ... |
| 2006-02-13-12.30.37.181478 | USERSPACE1 | 2002 | ... |

```

2006-02-13-12.30.37.181478 USERSPACE1          2003 ...
2006-02-13-12.30.37.181478 USERSPACE1          2003 ...

```

22 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```

... HI_TIMESTAMP          HI_VALUE HI_ALERT_STATE HI_ALERT_STATE_DETAIL ...
... -----
... 2006-02-13-12.16.25.911000          0          1 Normal
... 2006-02-13-12.06.26.168000          0          1 Normal
... 2006-02-13-12.16.25.911000          99          4 Alarm
... 2006-02-13-12.06.26.168000          99          4 Alarm
... 2006-02-13-12.16.25.911000          0          1 Normal
... 2006-02-13-12.06.26.168000          0          1 Normal
... 2006-02-13-12.16.25.911000          0          1 Normal
... 2006-02-13-12.06.26.168000          0          1 Normal
... 2006-02-13-12.16.25.911000          62          1 Normal
... 2006-02-13-12.06.26.168000          62          1 Normal
... 2006-02-13-12.16.25.911000          0          1 Normal
... 2006-02-13-12.06.26.168000          0          1 Normal
... 2006-02-13-12.16.25.911000          0          1 Normal
... 2006-02-13-12.06.26.168000          0          1 Normal
... 2006-02-13-12.16.25.911000          0          1 Normal
... 2006-02-13-12.06.26.168000          0          1 Normal
... 2006-02-13-12.16.25.911000          0          1 Normal
... 2006-02-13-12.06.26.168000          0          1 Normal
... 2006-02-13-12.16.25.911000          0          1 Normal
... 2006-02-13-12.06.26.168000          0          1 Normal
... 2006-02-13-12.16.25.911000          100         4 Alarm
... 2006-02-13-12.06.26.168000          100         4 Alarm
... 2006-02-13-12.16.25.911000          0          1 Normal
... 2006-02-13-12.06.26.168000          0          1 Normal

```

Datos de salida de esta consulta (continuación).

```

... HI_FORMULA          HI_ADDITIONAL_INFO
... -----
... 0 -
... 0 -
... ((9376 / 9468) * 100) The short term table space growth rate from
                          "02/13/2006 11:16:25.000911" to
                          "02/13/2006 12:16:25.000911" is "N/A" bytes
                          per second and the long term growth rate
                          from "02/12/2006 12:16:25.000911" to
                          "02/13/2006 12:16:25.000911" is "N/A" bytes
                          per second. Time to fullness is projected
                          to be "N/A" and "N/A" respectively. The
                          table space is defined with automatic
                          storage set to "YES" and automatic resize
                          enabled set to "YES".
... ((9376 / 9468) * 100) The short term table space growth rate from
                          "02/13/2006 11:06:26.000168" to
                          "02/13/2006 12:06:26.000168" is "N/A" bytes
                          per second and the long term growth rate
                          from "02/12/2006 12:06:26.000168" to
                          "02/13/2006 12:06:26.000168" is "N/A" bytes
                          per second. Time to fullness is projected
                          to be "N/A" and "N/A" respectively. The
                          table space is defined with automatic
                          storage set to "YES" and automatic resize
                          enabled set to "YES".
... 0 The table space is defined with automatic
      storage set to "YES" and automatic resize
      enabled set to "YES". The following are
      the automatic resize settings: increase
      size (bytes) "-1", increase size (percent)
      "N/A", maximum size (bytes) "-1". Las opciones
      current table space size (bytes) is
      "38797312".

```

```

... 0 The table space is defined with automatic
storage set to "YES" and automatic resize
enabled set to "YES". The following are
the automatic resize settings: increase
size (bytes) "-1", increase size (percent)
"N/A", maximum size (bytes) "-1". Las opciones
current table space size (bytes) is
"38797312".

... 0 -
... 0 -
... ((156 / 252) * 100) The short term table space growth rate from
"02/13/2006 11:16:25.000911" to
"02/13/2006 12:16:25.000911" is "N/A"
bytes per second and the long term growth
rate from "02/12/2006 12:16:25.000911" to
"02/13/2006 12:16:25.000911" is "N/A" bytes
per second. Time to fullness is projected
to be "N/A" and "N/A" respectively. The
table space is defined with automatic
storage set to "YES" and automatic resize
enabled set to "YES".

... ((156 / 252) * 100) The short term table space growth rate from
"02/13/2006 11:06:26.000168" to
"02/13/2006 12:06:26.000168" is "N/A"
bytes per second and the long term growth
rate from "02/12/2006 12:06:26.000168" to
"02/13/2006 12:06:26.000168" is "N/A" bytes
per second. Time to fullness is projected
to be "N/A" and "N/A" respectively. The
table space is defined with automatic
storage set to "YES" and automatic resize
enabled set to "YES".

... 0 The table space is defined with automatic
storage set to "YES" and automatic resize
enabled set to "YES". The following are
the automatic resize settings: increase
size (bytes) "-1", increase size (percent)
"N/A", maximum size (bytes) "-1". Las opciones
current table space size (bytes) is
"1048576".

... 0 The table space is defined with automatic
storage set to "YES" and automatic resize
enabled set to "YES". The following are
the automatic resize settings: increase
size (bytes) "-1", increase size (percent)
"N/A", maximum size (bytes) "-1". Las opciones
current table space size (bytes) is
"1048576".

... 0 -
... 0 -
... 0 -
... 0 -
... 0 -
... 0 -
... ((1504 / 1504) * 100) The short term table space growth rate
from "02/13/2006 11:16:25.000911" to
"02/13/2006 12:16:25.000911" is "N/A"
bytes per second and the long term growth
rate from "02/12/2006 12:16:25.000911"
to "02/13/2006 12:16:25.000911" is "N/A"
bytes per second. Time to fullness is
projected to be "N/A" and "N/A"
respectively. The table space is defined
with automatic storage set to "YES" and
automatic resize enabled set to "YES".

... ((1504 / 1504) * 100) The short term table space growth rate
from "02/13/2006 11:06:26.000168" to

```

```

... 0 "02/13/2006 12:06:26.000168" is "N/A"
bytes per second and the long term growth
rate from "02/12/2006 12:06:26.000168"
to "02/13/2006 12:06:26.000168" is "N/A"
bytes per second. Time to fullness is
projected to be "N/A" and "N/A"
respectively. The table space is defined
with automatic storage set to "YES" and
automatic resize enabled set to "YES".
... 0 The table space is defined with automatic
storage set to "YES" and automatic
resize enabled set to "YES". The
following are the automatic resize
settings: increase size (bytes) "-1",
increase size (percent) "N/A", maximum
size (bytes) "-1". The current table
space size (bytes) is "6291456".
... 0 The table space is defined with automatic
storage set to "YES" and automatic
resize enabled set to "YES". The
following are the automatic resize
settings: increase size (bytes) "-1",
increase size (percent) "N/A", maximum
size (bytes) "-1". The current table
space size (bytes) is "6291456".

```

Información devuelta

Tabla 250. Información devuelta por la función de tabla HEALTH_TBS_HI_HIS

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| TABLESPACE_NAME | VARCHAR(128) | tablespace_name - Nombre de espacio de tablas |
| HI_ID | BIGINT | Número que identifica de forma exclusiva el indicador de salud en la corriente de datos de la instantánea. |
| HI_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se generó la alerta. |
| HI_VALUE | SMALLINT | Valor del indicador de salud. |
| HI_ALERT_STATE | BIGINT | La gravedad de la alerta. |
| HI_ALERT_STATE_DETAIL | VARCHAR(20) | Descripción textual de la columna HI_ALERT_STATE. |
| HI_FORMULA | VARCHAR(2048) | Fórmula utilizada para calcular el indicador de salud. |
| HI_ADDITIONAL_INFO | VARCHAR(4096) | Información adicional acerca del indicador de salud. |

HEALTH_TBS_INFO

Devuelve información de espacio de tablas a partir de una instantánea de salud de una base de datos.

Importante: Esta función de tabla ha quedado en desuso y puede que se elimine en un futuro release ya que el Supervisor de salud ha quedado en desuso en la

Versión 9.7. Para obtener más información, consulte el tema “El Supervisor de salud ha quedado en desuso” de la publicación *Novedades en DB2 Versión 9.7*.

Sintaxis

►►—HEALTH_TBS_INFO—(—nombre-bd—,—númparticiónbd—)—————►►

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(255) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique el valor nulo para tomar la instantánea desde la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE para la función de tabla HEALTH_TBS_INFO.

Ejemplo

```
SELECT * FROM TABLE(HEALTH_TBS_INFO(' ', -1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
SNAPSHOT_TIMESTAMP      TABLESPACE_NAME      ...
-----
2006-02-13-12.30.35.027383 SYSCATSPACE          ...
2006-02-13-12.30.35.027383 SYSTOOLSPACE         ...
2006-02-13-12.30.35.027383 SYSTOOLSTMPSPACE    ...
2006-02-13-12.30.35.027383 TEMPSPACE1          ...
2006-02-13-12.30.35.027383 USERSPACE1          ...
```

5 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```
... ROLLED_UP_ALERT_STATE ROLLED_UP_ALERT_STATE_DETAIL
... -----
...                4 Alarm
...                1 Normal
...                1 Normal
...                1 Normal
...                4 Alarm
```

Información devuelta

Tabla 251. Información devuelta por la función de tabla HEALTH_TBS_INFO

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| TABLESPACE_NAME | VARCHAR(128) | tablespace_name - Nombre de espacio de tablas |
| ROLLED_UP_ALERT_STATE | BIGINT | Estado de alerta más grave capturado por esta instantánea. |
| ROLLED_UP_ALERT_STATE_DETAIL | VARCHAR(20) | Descripción textual de la columna ROLLED_UP_ALERT_STATE. |

Función de tabla SNAP_GET_APPL – Recuperar información de instantáneas del grupo de datos lógico appl

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la “Vista administrativa SNAPAPPL y función de tabla SNAP_GET_APPL_V95 – Recuperar información de instantáneas del grupo de datos lógicos appl” en la página 530.

La SNAP_GET_APPLY devuelve información acerca de las aplicaciones a partir de una instantánea de la aplicación, en concreto del grupo de datos lógico appl.

La función de tabla SNAP_GET_APPL, utilizada con las funciones de tabla SNAP_GET_AGENT, SNAP_GET_AGENT_MEMORY_POOL, SNAP_GET_APPL_INFO, SNAP_GET_STMT y SNAP_GET_SUBSECTION, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR ALL APPLICATIONS, pero recupera datos de todas las particiones de base de datos.

Consulte la Tabla 252 en la página 1052 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
▶▶ SNAP_GET_APPL ( ( nombre_bd [ , numparticiónbd ] ) ) ▶▶
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique una serie vacía para tomar la instantánea desde la base de datos conectada actualmente. Especifique un valor NULL para tomar la instantánea desde todas las bases de datos que se encuentran en la misma instancia que la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombredb* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombredb*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombredb* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_APPL toma una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

- Autorización SYSMON
- Privilegio EXECUTE en la función de tabla SNAP_GET_APPL.

Ejemplo

Recuperar detalles en filas leídas y grabadas para cada aplicación en todas las bases de datos activas.

```
SELECT SUBSTR(DB_NAME,1,8) AS DB_NAME, AGENT_ID, ROWS_READ, ROWS_WRITTEN  
FROM TABLE (SNAP_GET_APPL(CAST(NULL AS VARCHAR(128)),-1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| DB_NAME | AGENT_ID | ROWS_READ | ROWS_WRITTEN |
|---------|----------|-----------|--------------|
| WSDB | 679 | 0 | 0 |
| WSDB | 461 | 3 | 0 |
| WSDB | 460 | 4 | 0 |
| TEST | 680 | 4 | 0 |
| TEST | 455 | 6 | 0 |
| TEST | 454 | 0 | 0 |
| TEST | 453 | 50 | 0 |

Información devuelta

Tabla 252. Información devuelta por la función de tabla SNAP_GET_APPL

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| AGENT_ID | BIGINT | agent_id - Descriptor de contexto de aplicación (ID de agente) |
| UOW_LOG_SPACE_USED | BIGINT | uow_log_space_used - Unidad de espacio de registro de trabajo utilizado |

Tabla 252. Información devuelta por la función de tabla SNAP_GET_APPL (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|--|
| ROWS_READ | BIGINT | rows_read - Filas leídas |
| ROWS_WRITTEN | BIGINT | rows_written - Filas grabadas |
| INACT_STMTHIST_SZ | BIGINT | stmt_history_list_size - Tamaño de lista de historial de sentencias |
| POOL_DATA_L_READS | BIGINT | pool_data_l_reads - Lecturas lógicas de datos de la agrupación de almacenamientos intermedios |
| POOL_DATA_P_READS | BIGINT | pool_data_p_reads - Lecturas físicas de datos de la agrupación de almacenamientos intermedios |
| POOL_DATA_WRITES | BIGINT | pool_data_writes - Grabaciones de datos de la agrupación de almacenamientos intermedios |
| POOL_INDEX_L_READS | BIGINT | pool_index_l_reads - Lecturas lógicas de índices de la agrupación de almacenamientos intermedios |
| POOL_INDEX_P_READS | BIGINT | pool_index_p_reads - Lecturas físicas de índices de la agrupación de almacenamientos intermedios |
| POOL_INDEX_WRITES | BIGINT | pool_index_writes - Grabaciones de índices de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_L_READS | BIGINT | pool_temp_data_l_reads - Lecturas lógicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_P_READS | BIGINT | pool_temp_data_p_reads - Lecturas físicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_L_READS | BIGINT | pool_temp_index_l_reads - Lecturas lógicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_P_READS | BIGINT | pool_temp_index_p_reads - Lecturas físicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_L_READS | BIGINT | pool_temp_xda_l_reads - Lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_P_READS | BIGINT | pool_temp_xda_p_reads - Elemento del supervisor de las lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |

Tabla 252. Información devuelta por la función de tabla SNAP_GET_APPL (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------|---------------|---|
| POOL_XDA_L_READS | BIGINT | pool_xda_l_reads - Lecturas lógicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_XDA_P_READS | BIGINT | pool_xda_p_reads - Lecturas físicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_XDA_WRITES | BIGINT | pool_xda_writes - Grabaciones de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_READ_TIME | BIGINT | pool_read_time - Tiempo total de lectura física de la agrupación de almacenamientos intermedios |
| POOL_WRITE_TIME | BIGINT | pool_write_time - Tiempo total de grabación física de la agrupación de almacenamientos intermedios |
| DIRECT_READS | BIGINT | direct_reads - Lecturas directas de la base de datos |
| DIRECT_WRITES | BIGINT | direct_writes - Grabaciones directas en la base de datos |
| DIRECT_READ_REQS | BIGINT | direct_read_reqs - Peticiones de lectura directa |
| DIRECT_WRITE_REQS | BIGINT | direct_write_reqs - Peticiones de grabación directa |
| DIRECT_READ_TIME | BIGINT | direct_read_time - Tiempo de lectura directa |
| DIRECT_WRITE_TIME | BIGINT | direct_write_time - Tiempo de grabación directa |
| UNREAD_PREFETCH_PAGES | BIGINT | unread_prefetch_pages - Páginas de captación previa no leídas |
| LOCKS_HELD | BIGINT | locks_held - Bloqueos retenidos |
| LOCK_WAITS | BIGINT | lock_waits - Esperas de bloqueo |
| LOCK_WAIT_TIME | BIGINT | lock_wait_time - Tiempo esperado en bloqueos |
| LOCK_ESCALS | BIGINT | lock_escals - Número de escalamientos de bloqueo |
| X_LOCK_ESCALS | BIGINT | x_lock_escals - Escalas de bloqueo exclusivas |
| DEADLOCKS | BIGINT | deadlocks - Puntos muertos detectados |
| TOTAL_SORTS | BIGINT | total_sorts - Clasificaciones totales |
| TOTAL_SORT_TIME | BIGINT | total_sort_time - Tiempo de clasificación total |
| SORT_OVERFLOWS | BIGINT | sort_overflows - Desbordamientos de clasificación |
| COMMIT_SQL_STMTS | BIGINT | commit_sql_stmts - Sentencias de confirmación intentadas |

Tabla 252. Información devuelta por la función de tabla SNAP_GET_APPL (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------|---------------|--|
| ROLLBACK_SQL_STMTS | BIGINT | rollback_sql_stmts - Sentencias de retrotracción intentadas |
| DYNAMIC_SQL_STMTS | BIGINT | dynamic_sql_stmts - Sentencias de SQL dinámico intentadas |
| STATIC_SQL_STMTS | BIGINT | static_sql_stmts - Sentencias de SQL estático intentadas |
| FAILED_SQL_STMTS | BIGINT | failed_sql_stmts - Operaciones de sentencia anómalas |
| SELECT_SQL_STMTS | BIGINT | select_sql_stmts - Sentencias Select de SQL ejecutadas |
| DDL_SQL_STMTS | BIGINT | ddl_sql_stmts - Sentencias de lenguaje de definición de datos (DDL) de SQL |
| UID_SQL_STMTS | BIGINT | uid_sql_stmts - Sentencias UPDATE/INSERT/DELETE de SQL ejecutadas |
| INT_AUTO_REBINDS | BIGINT | int_auto_rebinds - Revinculaciones automáticas internas |
| INT_ROWS_DELETED | BIGINT | int_rows_deleted - Filas internas suprimidas |
| INT_ROWS_UPDATED | BIGINT | int_rows_updated - Filas internas actualizadas |
| INT_COMMITS | BIGINT | int_commits - Confirmaciones internas |
| INT_ROLLBACKS | BIGINT | int_rollback - Retrotracciones internas |
| INT_DEADLOCK_ROLLBACKS | BIGINT | int_deadlock_rollback - Retrotracciones internas debido a punto muerto |
| ROWS_DELETED | BIGINT | rows_deleted - Filas suprimidas |
| ROWS_INSERTED | BIGINT | rows_inserted - Filas insertadas |
| ROWS_UPDATED | BIGINT | rows_updated - Filas actualizadas |
| ROWS_SELECTED | BIGINT | rows_selected - Filas seleccionadas |
| BINDS_PRECOMPILES | BIGINT | binds_precompiles - Vinculaciones/precompilaciones intentadas |
| OPEN_REM_CURS | BIGINT | open_rem_curs - Cursores remotos abiertos |
| OPEN_REM_CURS_BLK | BIGINT | open_rem_curs_blk - Cursores remotos abiertos con bloqueo |
| REJ_CURS_BLK | BIGINT | rej_curs_blk - Peticiones de cursor de bloqueo rechazadas |
| ACC_CURS_BLK | BIGINT | acc_curs_blk - Peticiones de cursor de bloqueo aceptadas |
| SQL_REQS_SINCE_COMMIT | BIGINT | sql_reqs_since_commit - Peticiones SQL desde la última confirmación |

Tabla 252. Información devuelta por la función de tabla SNAP_GET_APPL (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------|---------------|---|
| LOCK_TIMEOUTS | BIGINT | lock_timeouts - Número de tiempos de espera de bloqueo |
| INT_ROWS_INSERTED | BIGINT | int_rows_inserted - Filas internas insertadas |
| OPEN_LOC_CURS | BIGINT | open_loc_curs - Cursores locales abiertos |
| OPEN_LOC_CURS_BLK | BIGINT | open_loc_curs_blk - Cursores locales abiertos con bloqueo |
| PKG_CACHE_LOOKUPS | BIGINT | pkg_cache_lookups - Búsquedas de antememoria de paquete |
| PKG_CACHE_INSERTS | BIGINT | pkg_cache_inserts - Inserciones de antememoria de paquete |
| CAT_CACHE_LOOKUPS | BIGINT | cat_cache_lookups - Búsquedas de antememoria de catálogo |
| CAT_CACHE_INSERTS | BIGINT | cat_cache_inserts - Inserciones de antememoria de catálogo |
| CAT_CACHE_OVERFLOWS | BIGINT | cat_cache_overflows - Desbordamientos de antememoria de catálogo |
| NUM_AGENTS | BIGINT | num_agents - Número de agentes trabajando en una sentencia |
| AGENTS_STOLEN | BIGINT | agents_stolen - Agentes robados |
| ASSOCIATED_AGENTS_TOP | BIGINT | associated_agents_top - Número máximo de agentes asociados |
| APPL_PRIORITY | BIGINT | appl_priority - Prioridad de agente de aplicación |
| APPL_PRIORITY_TYPE | VARCHAR(16) | appl_priority_type - Tipo de prioridad de la aplicación. Esta interfaz devuelve un identificador de texto, basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • DYNAMIC_PRIORITY • FIXED_PRIORITY |
| PREFETCH_WAIT_TIME | BIGINT | prefetch_wait_time - Tiempo esperado para captación previa |
| APPL_SECTION_LOOKUPS | BIGINT | appl_section_lookups - Búsquedas de sección |
| APPL_SECTION_INSERTS | BIGINT | appl_section_inserts - Inserciones de sección |
| LOCKS_WAITING | BIGINT | locks_waiting - Agentes actuales en espera de bloqueos |
| TOTAL_HASH_JOINS | BIGINT | total_hash_joins - Total de uniones hash |
| TOTAL_HASH_LOOPS | BIGINT | total_hash_loops - Total de bucles hash |

Tabla 252. Información devuelta por la función de tabla SNAP_GET_APPL (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------------|---------------|---|
| HASH_JOIN_OVERFLOWS | BIGINT | hash_join_overflows - Desbordamientos de uniones hash |
| HASH_JOIN_SMALL_OVERFLOWS | BIGINT | hash_join_small_overflows - Desbordamientos pequeños de uniones hash |
| APPL_IDLE_TIME | BIGINT | appl_idle_time - Tiempo de inactividad de aplicación |
| UOW_LOCK_WAIT_TIME | BIGINT | uow_lock_wait_time - Tiempo total que la unidad de trabajo ha esperado en bloqueos |
| UOW_COMP_STATUS | VARCHAR(14) | uow_comp_status - Estado de finalización de la unidad de trabajo. Esta interfaz devuelve un identificador de texto, basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • APPL_END • UOWABEND • UOWCOMMIT • UOWDEADLOCK • UOWLOCKTIMEOUT • UOWROLLBACK • UOWUNKNOWN |
| AGENT_USR_CPU_TIME_S | BIGINT | agent_usr_cpu_time - Tiempo de CPU de usuario utilizado por el agente |
| AGENT_USR_CPU_TIME_MS | BIGINT | agent_usr_cpu_time - Tiempo de CPU de usuario utilizado por el agente |
| AGENT_SYS_CPU_TIME_S | BIGINT | agent_sys_cpu_time - Tiempo de CPU de sistema utilizado por el agente |
| AGENT_SYS_CPU_TIME_MS | BIGINT | agent_sys_cpu_time - Tiempo de CPU de sistema utilizado por el agente |
| APPL_CON_TIME | TIMESTAMP | appl_con_time - Indicación de fecha y hora de inicio de petición de conexión |
| CONN_COMPLETE_TIME | TIMESTAMP | conn_complete_time - Indicación de fecha y hora de finalización de petición de conexión |
| LAST_RESET | TIMESTAMP | last_reset - Indicación de fecha y hora de último restablecimiento |
| UOW_START_TIME | TIMESTAMP | uow_start_time - Indicación de fecha y hora de inicio de unidad de trabajo |
| UOW_STOP_TIME | TIMESTAMP | uow_stop_time - Indicación de fecha y hora de detención de unidad de trabajo |

Tabla 252. Información devuelta por la función de tabla SNAP_GET_APPL (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------------------|---------------|---|
| PREV_UOW_STOP_TIME | TIMESTAMP | prev_uow_stop_time - Indicación de fecha y hora de finalización de unidad de trabajo anterior |
| UOW_ELAPSED_TIME_S | BIGINT | uow_elapsed_time - Tiempo transcurrido de unidad de trabajo más reciente |
| UOW_ELAPSED_TIME_MS | BIGINT | uow_elapsed_time - Tiempo transcurrido de unidad de trabajo más reciente |
| ELAPSED_EXEC_TIME_S | BIGINT | elapsed_exec_time - Tiempo transcurrido de ejecución de sentencia |
| ELAPSED_EXEC_TIME_MS | BIGINT | elapsed_exec_time - Tiempo transcurrido de ejecución de sentencia |
| INBOUND_COMM_ADDRESS | VARCHAR(32) | inbound_comm_address - Dirección de comunicaciones de entrada |
| LOCK_TIMEOUT_VAL | BIGINT | lock_timeout_val - Tiempo de espera excedido de bloqueo (segundos) |
| PRIV_WORKSPACE_NUM_OVERFLOWS | BIGINT | priv_workspace_num_overflows - Desbordamientos de espacio de trabajo privado |
| PRIV_WORKSPACE_SECTION_INSERTS | BIGINT | priv_workspace_section_inserts - Inserciones de sección de espacio de trabajo privado |
| PRIV_WORKSPACE_SECTION_LOOKUPS | BIGINT | priv_workspace_section_lookups - Búsquedas de sección de espacio de trabajo privado |
| PRIV_WORKSPACE_SIZE_TOP | BIGINT | priv_workspace_size_top - Tamaño máximo de espacio de trabajo privado |
| SHR_WORKSPACE_NUM_OVERFLOWS | BIGINT | shr_workspace_num_overflows - Desbordamientos de espacio de trabajo compartido |
| SHR_WORKSPACE_SECTION_INSERTS | BIGINT | shr_workspace_section_inserts - Inserciones de sección de espacio de trabajo compartido |
| SHR_WORKSPACE_SECTION_LOOKUPS | BIGINT | shr_workspace_section_lookups - Búsquedas de sección de espacio de trabajo compartido |
| SHR_WORKSPACE_SIZE_TOP | BIGINT | shr_workspace_size_top - Tamaño máximo de espacio de trabajo compartido |
| DBPARTITIONNUM | SMALLINT | La partición de base de datos desde la que se han recuperado los datos para la fila. |

Tabla 252. Información devuelta por la función de tabla SNAP_GET_APPL (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|--|
| CAT_CACHE_SIZE_TOP | BIGINT | cat_cache_size_top - Marca de límite superior de antememoria de catálogo |

Función de tabla SNAP_GET_APPL_INFO – Recuperación de información de instantáneas del grupo de datos lógico appl_info

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la “Vista administrativa SNAPAPPL_INFO y función de tabla SNAP_GET_APPL_INFO_V95 – Recuperar información de instantáneas del grupo de datos lógicos appl_info” en la página 523.

La función de tabla SNAP_GET_APPLY devuelve información acerca de las aplicaciones a partir de una instantánea de la aplicación, en concreto del grupo de datos lógico appl_info.

La función de tabla SNAP_GET_APPL_INFO, utilizada con las funciones de tabla SNAP_GET_AGENT, SNAP_GET_AGENT_MEMORY_POOL, SNAP_GET_APPL, SNAP_GET_APPL_INFO, SNAP_GET_STMT y SNAP_GET_SUBSECTION, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR ALL APPLICATIONS, pero recupera datos de todas las particiones de base de datos.

Consulte la Tabla 253 en la página 1061 para ver una lista completa de la información que se puede devolver.

Sintaxis

```

▶▶ SNAP_GET_APPL_INFO ( ( nombre-bd [ , nóm-partición-bd ] ) )

```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique una serie vacía para tomar la instantánea desde la base de datos conectada actualmente. Especifique un valor NULL para tomar la instantánea desde todas las bases de datos que se encuentran en la misma instancia que la base de datos conectada actualmente.

núm-particiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *núm-particiónbd* se establece en NULL, -1 se establece implícitamente para *núm-particiónbd*. Si no se utiliza esta

opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_APPL_INFO tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP_GET_APPL_INFO.

Ejemplos

Recuperar el estado de todas las aplicaciones en la partición de base de datos conectada.

```
SELECT SUBSTR(DB_NAME,1,8) AS DB_NAME, AGENT_ID,
       SUBSTR(APPL_NAME,1,10) AS APPL_NAME, APPL_STATUS
FROM TABLE(SNAP_GET_APPL_INFO(CAST(NULL AS VARCHAR(128)),-1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| DB_NAME | AGENT_ID | APPL_NAME | APPL_STATUS |
|---------|----------|-----------|-------------|
| TOOLSDB | 14 | db2bp.exe | CONNECTED |
| SAMPLE | 15 | db2bp.exe | UOWEXEC |
| SAMPLE | 8 | javaw.exe | CONNECTED |
| SAMPLE | 7 | db2bp.exe | UOWWAIT |

4 registro(s) seleccionado(s).

A continuación se muestra lo que se obtiene al ejecutar SELECT sobre el resultado de la función de tabla.

```
SELECT SUBSTR(DB_NAME,1,8) AS DB_NAME, AUTHORITY_LVL
FROM TABLE(SNAP_GET_APPL_INFO_V95(CAST(NULL AS VARCHAR(128)),-1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| DB_NAME | AUTHORITY_LVL |
|---------|---|
| TESTDB | SYSADM(GROUP) + DBADM(USER) + CREATETAB(USER, GROUP) + BINDADD(USER, GROUP) + CONNECT(USER, GROUP) + CREATE_NOT_FENC(USER) + IMPLICIT_SCHEMA(USER, GROUP) + LOAD(USER) + CREATE_EXT_RT(USER) + QUIESCE_CONN(USER) |
| TESTDB | SYSADM(GROUP) + DBADM(USER) + CREATETAB(USER, GROUP) + BINDADD(USER, GROUP) + CONNECT(USER, GROUP) + CREATE_NOT_FENC(USER) + IMPLICIT_SCHEMA(USER, GROUP) + LOAD(USER) + CREATE_EXT_RT(USER) + QUIESCE_CONN(USER) |
| TESTDB | SYSADM(GROUP) + DBADM(USER) + CREATETAB(USER, GROUP) + BINDADD(USER, GROUP) + CONNECT(USER, GROUP) + CREATE_NOT_FENC(USER) + IMPLICIT_SCHEMA(USER, GROUP) + LOAD(USER) + CREATE_EXT_RT(USER) + QUIESCE_CONN(USER) |

3 registro(s) seleccionado(s).

Información devuelta

Tabla 253. Información devuelta por la función de tabla SNAP_GET_APPL_INFO

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| AGENT_ID | BIGINT | agent_id - Descriptor de contexto de aplicación (ID de agente) |
| APPL_STATUS | VARCHAR(22) | <p>appl_status - Estado de aplicación. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes:</p> <ul style="list-style-type: none"> • BACKUP • COMMIT_ACT • COMP • CONNECTED • CONNECTPEND • CREATE_DB • DECOUPLED • DISCONNECTPEND • INTR • IOERROR_WAIT • LOAD • LOCKWAIT • QUIESCE_TABLESPACE • RECOMP • REMOTE_RQST • RESTART • RESTORE • ROLLBACK_ACT • ROLLBACK_TO_SAVEPOINT • TEND • THABRT • THCOMT • TPREP • UNLOAD • UOWEXEC • UOWWAIT • WAITFOR_REMOTE |
| CODEPAGE_ID | BIGINT | codepage_id - ID de página de códigos utilizada por aplicación |
| NUM_ASSOC_AGENTS | BIGINT | num_assoc_agents - Número de agentes asociados |
| COORD_NODE_NUM | SMALLINT | coord_node - Nodo de coordinación |

Tabla 253. Información devuelta por la función de tabla SNAP_GET_APPL_INFO (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|--|
| AUTHORITY_LVL | VARCHAR(512) | <p>authority_lvl - Nivel de autorización del usuario.</p> <p>Esta interfaz devuelve un identificador de texto basado en las autorizaciones de base de datos definidas en sql.h y su fuente y tiene el siguiente formato: authority(source, ...) + authority(source, ...) + ... La fuente de una autorización puede ser múltiple: de un USER, de un GROUP, o de un USER y un GROUP.</p> <p>Valores posibles para "authority":</p> <ul style="list-style-type: none"> • BINDADD • CONNECT • CREATE_EXT_RT • CREATE_NOT_FENC • CREATETAB • DBADM • IMPLICIT_SCHEMA • LOAD • LIBADM • QUIESCE_CONN • SECADM • SYSADM • SYSCtrl • SYSMAINT • SYSMON • SYSQUIESCE <p>Valores posibles para "source":</p> <ul style="list-style-type: none"> • USER – autorización que se otorga al usuario o a un rol otorgado al usuario. • GROUP – autorización que se otorga a un grupo al que pertenece el usuario o a un rol que se ha otorgado al grupo al que pertenece el usuario. |
| CLIENT_PID | BIGINT | client_pid - ID de proceso de cliente |
| COORD_AGENT_PID | BIGINT | coord_agent_pid - Agente coordinador |
| STATUS_CHANGE_TIME | TIMESTAMP | status_change_time - Hora de cambio de estado de aplicación |

Tabla 253. Información devuelta por la función de tabla SNAP_GET_APPL_INFO (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|--|
| CLIENT_PLATFORM | VARCHAR(12) | <p>client_platform - Plataforma operativa de cliente. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h,</p> <ul style="list-style-type: none"> • AIX • AIX64 • AS400_DRDA • DOS • DYNIX • HP • HP64 • HPIA • HPIA64 • LINUX • LINUX390 • LINUXIA64 • LINUXPPC • LINUXPPC64 • LINUXX8664 • LINUXZ64 • MAC • MVS_DRDA • NT • NT64 • OS2 • OS390 • SCO • SGI • SNI • SUN • SUN64 • UNKNOWN • UNKNOWN_DRDA • VM_DRDA • VSE_DRDA • WINDOWS • WINDOWS95 |

Tabla 253. Información devuelta por la función de tabla SNAP_GET_APPL_INFO (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------|---------------|---|
| CLIENT_PROTOCOL | VARCHAR(10) | client_protocol - Protocolo de comunicaciones de cliente. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h, <ul style="list-style-type: none"> • CPIC • LOCAL • NETBIOS • NPIPE • TCPIP (para DB2 UDB) • TCPIP4 • TCPIP6 |
| TERRITORY_CODE | SMALLINT | territory_code - Código de territorio de base de datos |
| APPL_NAME | VARCHAR(256) | appl_name - Nombre de aplicación |
| APPL_ID | VARCHAR(128) | appl_id - ID de aplicación |
| SEQUENCE_NO | VARCHAR(4) | sequence_no - Número de secuencia |
| PRIMARY_AUTH_ID | VARCHAR(128) | auth_id - ID de autorización |
| SESSION_AUTH_ID | VARCHAR(128) | session_auth_id - ID de autorización de sesión |
| CLIENT_NNAME | VARCHAR(128) | El elemento del supervisor client_nname ha quedado obsoleto. El valor devuelto no es un valor válido. |
| CLIENT_PRDID | VARCHAR(128) | client_prdid - ID de producto/versión de cliente |
| INPUT_DB_ALIAS | VARCHAR(128) | input_db_alias - Alias de base de datos de entrada |
| CLIENT_DB_ALIAS | VARCHAR(128) | client_db_alias - Alias de base de datos utilizado por aplicación |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| DB_PATH | VARCHAR(1024) | db_path - Vía de acceso de base de datos |
| EXECUTION_ID | VARCHAR(128) | execution_id - ID de inicio de sesión de usuario |
| CORR_TOKEN | VARCHAR(128) | corr_token - Símbolo de correlación DRDA |
| TPMON_CLIENT_USERID | VARCHAR(256) | tpmon_client_userid - ID de usuario de cliente de supervisor de TP |
| TPMON_CLIENT_WKSTN | VARCHAR(256) | tpmon_client_wkstn - Nombre de estación de trabajo de cliente de supervisor de TP |

Tabla 253. Información devuelta por la función de tabla SNAP_GET_APPL_INFO (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|--|
| TPMON_CLIENT_APP | VARCHAR(256) | tpmon_client_app - Nombre de aplicación de cliente de supervisor de TP |
| TPMON_ACC_STR | VARCHAR(200) | tpmon_acc_str - Serie de contabilidad de cliente de supervisor de TP |
| DBPARTITIONNUM | SMALLINT | La partición de base de datos desde la que se han recuperado los datos para la fila. |

Función de tabla SNAP_GET_BP – Recuperación de la información de instantáneas del grupo de datos lógico bufferpool

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la “Vista administrativa SNAPBP y función de tabla SNAP_GET_BP_V95 – Recuperar información de instantáneas del grupo lógico bufferpool” en la página 540.

La función de tabla SNAP_GET_BP devuelve información acerca de las aplicaciones a partir de una instantánea de la aplicación, en concreto del grupo de datos lógico bufferpool.

La función de tabla SNAP_GET_BP, utilizada con la función de tabla SNAP_GET_BP_PART, ofrece los datos equivalentes al mandato CLP GET SNAPSHOT FOR ALL BUFFERPOOLS.

Consulte la Tabla 254 en la página 1066 para ver una lista completa de la información que se puede devolver.

Sintaxis

```

▶▶ SNAP_GET_BP ( ( nombre_bd [ , numparticiónbd ] ) )

```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique una serie vacía para tomar la instantánea desde la base de datos conectada actualmente. Especifique un valor NULL para tomar la instantánea desde todas las bases de datos que se encuentran en la misma instancia que la base de datos conectada actualmente.

numparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número

válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_BP tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP_GET_BP.

Ejemplo

Recuperar lecturas lógicas y físicas para todas las agrupaciones de almacenamientos intermedios de todas las bases de datos activas en la partición de base de datos conectada actualmente.

```
SELECT SUBSTR(T.DB_NAME,1,10) AS DB_NAME,
       SUBSTR(T.BP_NAME,1,20) AS BP_NAME,
       (T.POOL_DATA_L_READS+T.POOL_INDEX_L_READS) AS TOTAL_LOGICAL_READS,
       (T.POOL_DATA_P_READS+T.POOL_INDEX_P_READS) AS TOTAL_PHYSICAL_READS,
       T.DBPARTITIONNUM
FROM TABLE(SNAP_GET_BP(CAST(NULL AS VARCHAR(128)), -1)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
DB_NAME      BP_NAME      TOTAL_LOGICAL_READS  ...
-----
SAMPLE      IBMDEFAULTBP      0 ...
TOOLSDB     IBMDEFAULTBP      0 ...
TOOLSDB     BP32K0000         0 ...
```

3 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```
... TOTAL_PHYSICAL_READS DBPARTITIONNUM
... -----
...                0                0
...                0                0
...                0                0
```

Información devuelta

Tabla 254. Información devuelta por la función de tabla SNAP_GET_BP

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |

Tabla 254. Información devuelta por la función de tabla SNAP_GET_BP (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------|---------------|---|
| BP_NAME | VARCHAR(128) | bp_name - Nombre de la agrupación de almacenamientos intermedios. |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| DB_PATH | VARCHAR(1024) | db_path - Vía de acceso de base de datos |
| INPUT_DB_ALIAS | VARCHAR(128) | input_db_alias - Alias de base de datos de entrada |
| POOL_DATA_L_READS | BIGINT | pool_data_l_reads - Lecturas lógicas de datos de la agrupación de almacenamientos intermedios |
| POOL_DATA_P_READS | BIGINT | pool_data_p_reads - Lecturas físicas de datos de la agrupación de almacenamientos intermedios |
| POOL_DATA_WRITES | BIGINT | pool_data_writes - Grabaciones de datos de la agrupación de almacenamientos intermedios |
| POOL_INDEX_L_READS | BIGINT | pool_index_l_reads - Lecturas lógicas de índices de la agrupación de almacenamientos intermedios |
| POOL_INDEX_P_READS | BIGINT | pool_index_p_reads - Lecturas físicas de índices de la agrupación de almacenamientos intermedios |
| POOL_INDEX_WRITES | BIGINT | pool_index_writes - Grabaciones de índices de la agrupación de almacenamientos intermedios |
| POOL_XDA_L_READS | BIGINT | pool_xda_l_reads - Lecturas lógicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_XDA_P_READS | BIGINT | pool_xda_p_reads - Lecturas físicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_XDA_WRITES | BIGINT | pool_xda_writes - Grabaciones de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_READ_TIME | BIGINT | pool_read_time - Tiempo total de lectura física de la agrupación de almacenamientos intermedios |
| POOL_WRITE_TIME | BIGINT | pool_write_time - Tiempo total de grabación física de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_READS | BIGINT | pool_async_data_reads - Lecturas de datos asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_WRITES | BIGINT | pool_async_data_writes - Grabaciones de datos asíncronas de agrupación de almacenamientos intermedios |

Tabla 254. Información devuelta por la función de tabla SNAP_GET_BP (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|----------------------------|---------------|---|
| POOL_ASYNC_INDEX_READS | BIGINT | pool_async_index_reads - Lecturas de índice asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_WRITES | BIGINT | pool_async_index_writes - Grabaciones de índice asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_XDA_READS | BIGINT | pool_async_xda_reads - Lecturas de datos XDA asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_XDA_WRITES | BIGINT | pool_async_xda_writes - Grabaciones de datos XDA asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_READ_TIME | BIGINT | pool_async_read_time - Tiempo de lectura asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_WRITE_TIME | BIGINT | pool_async_write_time - Tiempo de grabación asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_READ_REQS | BIGINT | pool_async_data_read_reqs - Peticiones de lectura asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_READ_REQS | BIGINT | pool_async_index_read_reqs - Peticiones de lectura de índice asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_XDA_READ_REQS | BIGINT | pool_async_xda_read_reqs - Peticiones de lectura XDA asíncrona de agrupación de almacenamientos intermedios |
| DIRECT_READS | BIGINT | direct_reads - Lecturas directas de la base de datos |
| DIRECT_WRITES | BIGINT | direct_writes - Grabaciones directas en la base de datos |
| DIRECT_READ_REQS | BIGINT | direct_read_reqs - Peticiones de lectura directa |
| DIRECT_WRITE_REQS | BIGINT | direct_write_reqs - Peticiones de grabación directa |
| DIRECT_READ_TIME | BIGINT | direct_read_time - Tiempo de lectura directa |
| DIRECT_WRITE_TIME | BIGINT | direct_write_time - Tiempo de grabación directa |
| UNREAD_PREFETCH_PAGES | BIGINT | unread_prefetch_pages - Páginas de captación previa no leídas |

Tabla 254. Información devuelta por la función de tabla SNAP_GET_BP (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|--|
| FILES_CLOSED | BIGINT | files_closed - Archivos de base de datos cerrados |
| POOL_TEMP_DATA_L_READS | BIGINT | pool_temp_data_l_reads - Lecturas lógicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_P_READS | BIGINT | pool_temp_data_p_reads - Lecturas físicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_L_READS | BIGINT | pool_temp_index_l_reads - Lecturas lógicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_P_READS | BIGINT | pool_temp_index_p_reads - Lecturas físicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_L_READS | BIGINT | pool_temp_xda_l_reads - Lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_P_READS | BIGINT | pool_temp_xda_p_reads - Elemento del supervisor de las lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |
| POOL_NO_VICTIM_BUFFER | BIGINT | pool_no_victim_buffer - Almacenamiento intermedio sin víctimas de la agrupación de almacenamientos intermedios |
| PAGES_FROM_BLOCK_IOS | BIGINT | pages_from_block_ios - Número total de páginas leídas por E/S de bloque |
| PAGES_FROM_VECTORED_IOS | BIGINT | pages_from_vectored_ios - Número total de páginas leídas por E/S de vector |
| PHYSICAL_PAGE_MAPS | BIGINT | El elemento del supervisor physical_page_maps se ha dejado de mantener. El sistema devolverá un valor NULL para el elemento del supervisor que se ha dejado de mantener. |
| VECTORED_IOS | BIGINT | vectored_ios - Número de peticiones de E/S de vector |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

SNAP_GET_CONTAINER

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la "Vista administrativa SNAPCONTAINER y función de tabla SNAP_GET_CONTAINER_V91 – Recuperar información acerca de la instantánea de grupo de datos lógicos tablespace_container" en la página 549

►►—SNAP_GET_CONTAINER—(—nombre-bd—,—númparticiónbd—)—————►►

El esquema es SYSPROC.

La función de tabla SNAP_GET_CONTAINER devuelve información de instantáneas desde el grupo de datos lógicos tablespace_container.

nombre-bd

Argumento de entrada de tipo VARCHAR(255) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique el valor nulo para tomar la instantánea desde la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual. Si se especifica el valor nulo, se establece -1 de forma implícita.

Si ambos parámetros se establecen en NULL, la instantánea sólo se tomará si anteriormente no se ha creado un archivo mediante el procedimiento almacenado SNAPSHOT_FILEW para el tipo de petición de API de instantánea correspondiente.

La función devuelve una tabla tal como se muestra más abajo.

Tabla 255. Información devuelta por la función de tabla SNAP_GET_CONTAINER

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| TBSP_NAME | VARCHAR(128) | tablespace_name - Nombre de espacio de tablas |
| TBSP_ID | BIGINT | tablespace_id - Identificación de espacio de tablas |
| CONTAINER_NAME | VARCHAR(256) | container_name - Nombre de contenedor |
| CONTAINER_ID | BIGINT | container_id - Identificación de contenedor |
| CONTAINER_TYPE | SMALLINT | container_type - Tipo de contenedor. |
| TOTAL_PAGES | BIGINT | container_total_pages - Total de páginas en contenedor |

Tabla 255. Información devuelta por la función de tabla SNAP_GET_CONTAINER (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|--|
| USABLE_PAGES | BIGINT | container_usable_pages - Páginas utilizables en contenedor |
| ACCESSIBLE | SMALLINT | container_accessible - Accesibilidad de contenedor |
| STRIPE_SET | BIGINT | container_stripe_set - Conjunto de bandas |
| DBPARTITIONNUM | SMALLINT | node_number - Número de nodo |

SNAP_GET_DB

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la "Función de tabla SNAP_GET_DB_V91 - Recuperar información de instantánea desde el grupo lógico dbase" en la página 1083

►► SNAP_GET_DB(—nombre-bd—,—númparticiónbd—)◄◄

El esquema es SYSPROC.

La función de tabla SNAP_GET_DB devuelve información de instantáneas de la base de datos.

nombrebd

Argumento de entrada de tipo VARCHAR(255) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique el valor nulo para tomar la instantánea desde la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

Si ambos parámetros se establecen en NULL, la instantánea sólo se tomará si el procedimiento almacenado SNAPSHOT_FILEW no ha creado anteriormente un archivo para el tipo de petición de API de instantánea correspondiente.

La función devuelve una tabla tal como se muestra más abajo.

Tabla 256. Información devuelta por la función de tabla `SNAP_GET_DB`

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| DB_PATH | VARCHAR(1024) | db_path - Vía de acceso de base de datos |
| INPUT_DB_ALIAS | VARCHAR(128) | input_db_alias - Alias de base de datos de entrada |
| DB_STATUS | BIGINT | db_status - Estado de base de datos |
| CATALOG_PARTITION | SMALLINT | catalog_node - Número de nodo de catálogo |
| CATALOG_PARTITION_NAME | VARCHAR(128) | catalog_node_name - Nombre de red de nodo de catálogo |
| SERVER_PLATFORM | INTEGER | server_platform - Sistema operativo de servidor |
| DB_LOCATION | INTEGER | db_location - Ubicación de base de datos |
| DB_CONN_TIME | TIMESTAMP | db_conn_time - Indicación de fecha y hora de activación de base de datos |
| LAST_RESET | TIMESTAMP | last_reset - Indicación de fecha y hora de último restablecimiento |
| LAST_BACKUP | TIMESTAMP | last_backup - Indicación de fecha y hora de última copia de seguridad |
| CONNECTIONS_TOP | BIGINT | connections_top - Número máximo de conexiones simultáneas |
| TOTAL_CONS | BIGINT | total_cons - Conexiones desde la activación de base de datos |
| TOTAL_SEC_CONS | BIGINT | total_sec_cons - Conexiones secundarias |
| APPLS_CUR_CONS | BIGINT | appls_cur_cons - Aplicaciones conectadas actualmente |
| APPLS_IN_DB2 | BIGINT | appls_in_db2 - Aplicaciones que se ejecutan en la base de datos actualmente |
| NUM_ASSOC_AGENTS | BIGINT | num_assoc_agents - Número de agentes asociados |
| AGENTS_TOP | BIGINT | agents_top - Número de agentes creados |
| COORD_AGENTS_TOP | BIGINT | coord_agents_top - Número máximo de agentes de coordinación |
| LOCKS_HELD | BIGINT | locks_held - Bloqueos retenidos |

Tabla 256. Información devuelta por la función de tabla SNAP_GET_DB (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------|---------------|---|
| LOCK_WAITS | BIGINT | lock_waits - Esperas de bloqueo |
| LOCK_WAIT_TIME | BIGINT | lock_wait_time - Tiempo esperado en bloqueos |
| LOCK_LIST_IN_USE | BIGINT | lock_list_in_use - Memoria total de lista de bloqueos en uso |
| DEADLOCKS | BIGINT | deadlocks - Puntos muertos detectados |
| LOCK_ESCALS | BIGINT | lock_escals - Número de escalamientos de bloqueo |
| X_LOCK_ESCALS | BIGINT | x_lock_escals - Escalas de bloqueo exclusivas |
| LOCKS_WAITING | BIGINT | locks_waiting - Agentes actuales esperando en bloqueos |
| LOCK_TIMEOUTS | BIGINT | lock_timeouts - Número de tiempos de espera de bloqueo |
| NUM_INDOUBT_TRANS | BIGINT | num_indoubt_trans - Número de transacciones dudosas |
| SORT_HEAP_ALLOCATED | BIGINT | sort_heap_allocated - Total de pila de clasificación asignada |
| SORT_SHRHEAP_ALLOCATED | BIGINT | sort_shrheap_allocated - Pila de compartimiento de clasificación asignada actualmente |
| SORT_SHRHEAP_TOP | BIGINT | sort_shrheap_top - Marca de límite superior de pila de compartimiento de clasificación |
| TOTAL_SORTS | BIGINT | total_sorts - Clasificaciones totales |
| TOTAL_SORT_TIME | BIGINT | total_sort_time - Tiempo de clasificación total |
| SORT_OVERFLOWS | BIGINT | sort_overflows - Desbordamientos de clasificación |
| ACTIVE_SORTS | BIGINT | active_sorts - Clasificaciones activas |
| POOL_DATA_L_READS | BIGINT | pool_data_l_reads - Lecturas lógicas de datos de agrupación de almacenamientos intermedios |
| POOL_DATA_P_READS | BIGINT | pool_data_p_reads - Lecturas físicas de datos de agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_L_READS | BIGINT | pool_temp_data_l_reads - Lecturas lógicas de datos temporales de agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_P_READS | BIGINT | pool_temp_data_p_reads - Lecturas físicas de datos temporales de agrupación de almacenamientos intermedios |

Tabla 256. Información devuelta por la función de tabla SNAP_GET_DB (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|---|
| POOL_ASYNC_DATA_READS | BIGINT | pool_async_data_reads - Lecturas de datos asíncronas de agrupación de almacenamientos intermedios |
| POOL_DATA_WRITES | BIGINT | pool_data_writes - Grabaciones de datos de agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_WRITES | BIGINT | pool_async_data_writes - Grabaciones de datos asíncronas de agrupación de almacenamientos intermedios |
| POOL_INDEX_L_READS | BIGINT | pool_index_l_reads - Lecturas lógicas de índice de agrupación de almacenamientos intermedios |
| POOL_INDEX_P_READS | BIGINT | pool_index_p_reads - Lecturas físicas de índice de agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_L_READS | BIGINT | pool_temp_index_l_reads - Lecturas lógicas de índice temporales de agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_P_READS | BIGINT | pool_temp_index_p_reads - Lecturas físicas de índice temporales de agrupación de almacenamientos intermedios |
| POOL_INDEX_WRITES | BIGINT | pool_index_writes - Grabaciones de índice de agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_READS | BIGINT | pool_async_index_reads - Lecturas de índice asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_WRITES | BIGINT | pool_async_index_writes - Grabaciones de índice asíncronas de agrupación de almacenamientos intermedios |
| POOL_READ_TIME | BIGINT | pool_read_time - Tiempo total de lectura física de agrupación de almacenamientos intermedios |
| POOL_WRITE_TIME | BIGINT | pool_write_time - Tiempo total de grabación física de agrupación de almacenamientos intermedios |
| POOL_ASYNC_READ_TIME | BIGINT | pool_async_read_time - Tiempo de lectura asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_WRITE_TIME | BIGINT | pool_async_write_time - Tiempo de grabación asíncrona de agrupación de almacenamientos intermedios |

Tabla 256. Información devuelta por la función de tabla SNAP_GET_DB (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|----------------------------|---------------|---|
| POOL_ASYNC_DATA_READ_REQS | BIGINT | pool_async_data_read_reqs - Peticiones de lectura asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_READ_REQS | BIGINT | pool_async_index_read_reqs - Peticiones de lectura de índice asíncrona de agrupación de almacenamientos intermedios |
| POOL_NO_VICTIM_BUFFER | BIGINT | pool_no_victim_buffer - Almacenamientos intermedios sin víctimas de agrupación de almacenamientos intermedios |
| POOL_LSN_GAP_CLNS | BIGINT | pool_lsn_gap_clns - Limpiadores de espacio de anotaciones cronológicas de agrupación de almacenamientos intermedios activados |
| POOL_DRTY_PG_STEAL_CLNS | BIGINT | pool_drtv_pg_steal_clns - Limpiadores de páginas víctima de agrupación de almacenamientos intermedios activados |
| POOL_DRTY_PG_THRSH_CLNS | BIGINT | pool_drtv_pg_thrsh_clns - Limpiadores de umbral de agrupación de almacenamientos intermedios activados |
| PREFETCH_WAIT_TIME | BIGINT | prefetch_wait_time - Tiempo esperado para captación previa |
| UNREAD_PREFETCH_PAGES | BIGINT | unread_prefetch_pages - Páginas de captación previa no leídas |
| DIRECT_READS | BIGINT | direct_reads - Lecturas directas de la base de datos |
| DIRECT_WRITES | BIGINT | direct_writes - Grabaciones directas en la base de datos |
| DIRECT_READ_REQS | BIGINT | direct_read_reqs - Peticiones de lectura directa |
| DIRECT_WRITE_REQS | BIGINT | direct_write_reqs - Peticiones de grabación directa |
| DIRECT_READ_TIME | BIGINT | direct_read_time - Tiempo de lectura directa |
| DIRECT_WRITE_TIME | BIGINT | direct_write_time - Tiempo de grabación directa |
| FILES_CLOSED | BIGINT | files_closed - Archivos de bases de datos cerrados |
| POOL_DATA_TO_ESTORE | BIGINT | El elemento supervisor ESTORE pool_data_to_estore se ha dejado de mantener. Se devuelve un valor NULL para el elemento de supervisor que se ha dejado de mantener. |

Tabla 256. Información devuelta por la función de tabla SNAP_GET_DB (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------|---------------|---|
| POOL_INDEX_TO_ESTORE | BIGINT | El elemento de supervisor ESTORE pool_index_to_estore se ha dejado de mantener. Se devuelve un valor NULL para el elemento de supervisor que se ha dejado de mantener. |
| POOL_INDEX_FROM_ESTORE | BIGINT | El elemento de supervisor ESTORE pool_index_from_estore se ha dejado de mantener. Se devuelve un valor NULL para el elemento de supervisor que se ha dejado de mantener. |
| POOL_DATA_FROM_ESTORE | BIGINT | El elemento de supervisor ESTORE pool_data_from_estore se ha dejado de mantener. Se devuelve un valor NULL para el elemento de supervisor que se ha dejado de mantener. |
| ELAPSED_EXEC_TIME_S | BIGINT | elapsed_exec_time - Tiempo transcurrido de ejecución de sentencia |
| ELAPSED_EXEC_TIME_MS | BIGINT | elapsed_exec_time - Tiempo transcurrido de ejecución de sentencia |
| COMMIT_SQL_STMTS | BIGINT | commit_sql_stmts - Sentencias de confirmación intentadas |
| ROLLBACK_SQL_STMTS | BIGINT | rollback_sql_stmts - Sentencias de retrotracción intentadas |
| DYNAMIC_SQL_STMTS | BIGINT | dynamic_sql_stmts - Sentencias de SQL dinámico intentadas |
| STATIC_SQL_STMTS | BIGINT | static_sql_stmts - Sentencias de SQL estático intentadas |
| FAILED_SQL_STMTS | BIGINT | failed_sql_stmts - Operaciones de sentencia anómalas |
| SELECT_SQL_STMTS | BIGINT | select_sql_stmts - Sentencias de selección de SQL ejecutadas |
| UID_SQL_STMTS | BIGINT | uid_sql_stmts - Sentencias UPDATE/INSERT/DELETE de SQL ejecutadas |
| DDL_SQL_STMTS | BIGINT | ddl_sql_stmts - Sentencias de lenguaje de definición de datos (DDL) de SQL |
| INT_AUTO_REBINDS | BIGINT | int_auto_rebinds - Revinculaciones automáticas internas |
| INT_ROWS_DELETED | BIGINT | int_rows_deleted - Filas internas suprimidas |
| INT_ROWS_INSERTED | BIGINT | int_rows_inserted - Filas internas insertadas |

Tabla 256. Información devuelta por la función de tabla SNAP_GET_DB (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------|---------------|--|
| INT_ROWS_UPDATED | BIGINT | int_rows_updated - Filas internas actualizadas |
| INT_COMMITS | BIGINT | int_commits - Confirmaciones internas |
| INT_ROLLBACKS | BIGINT | int_rollback - Retrotracciones internas |
| INT_DEADLOCK_ROLLBACKS | BIGINT | int_deadlock_rollback - Retrotracciones internas debido a punto muerto |
| ROWS_DELETED | BIGINT | rows_deleted - Filas suprimidas |
| ROWS_INSERTED | BIGINT | rows_inserted - Filas insertadas |
| ROWS_UPDATED | BIGINT | rows_updated - Filas actualizadas |
| ROWS_SELECTED | BIGINT | rows_selected - Filas seleccionadas |
| ROWS_READ | BIGINT | rows_read - Filas leídas |
| BINDS_PRECOMPILES | BIGINT | binds_precompiles - Vinculaciones/precompilaciones intentadas |
| TOTAL_LOG_AVAILABLE | BIGINT | total_log_available - Anotaciones cronológicas totales disponibles |
| TOTAL_LOG_USED | BIGINT | total_log_used - Espacio utilizado de anotaciones cronológicas totales |
| SEC_LOG_USED_TOP | BIGINT | sec_log_used_top - Espacio máximo utilizado de anotaciones cronológicas secundarias |
| TOT_LOG_USED_TOP | BIGINT | tot_log_used_top - Espacio máximo utilizado de anotaciones cronológicas totales |
| SEC_LOGS_ALLOCATED | BIGINT | sec_logs_allocated - Anotaciones cronológicas secundarias asignadas actualmente |
| LOG_READS | BIGINT | log_reads - Número de páginas de anotaciones cronológicas leídas |
| LOG_READ_TIME_S | BIGINT | log_read_time - Tiempo de lectura de anotaciones cronológicas |
| LOG_READ_TIME_NS | BIGINT | log_read_time - Tiempo de lectura de anotaciones cronológicas |
| LOG_WRITES | BIGINT | log_writes - Número de páginas de anotaciones cronológicas grabadas |
| LOG_WRITE_TIME_S | BIGINT | log_write_time - Tiempo de grabación de anotaciones cronológicas |
| LOG_WRITE_TIME_NS | BIGINT | log_write_time - Tiempo de grabación de anotaciones cronológicas |

Tabla 256. Información devuelta por la función de tabla SNAP_GET_DB (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------------|---------------|---|
| NUM_LOG_WRITE_IO | BIGINT | num_log_write_io - Número de grabaciones de anotaciones cronológicas |
| NUM_LOG_READ_IO | BIGINT | num_log_read_io - Número de lecturas de anotaciones cronológicas |
| NUM_LOG_PART_PAGE_IO | BIGINT | num_log_part_page_io - Número de grabaciones de páginas de anotaciones cronológicas parciales |
| NUM_LOG_BUFFER_FULL | BIGINT | num_log_buffer_full - Número de almacenamientos intermedios de anotaciones cronológicas llenos |
| NUM_LOG_DATA_FOUND_IN_BUFFER | BIGINT | num_log_data_found_in_buffer - Número de datos de anotaciones cronológicas encontrados en el almacenamiento intermedio |
| APPL_ID_OLDEST_XACT | BIGINT | appl_id_oldest_xact - Aplicación con la transacción más antigua |
| LOG_TO_REDO_FOR_RECOVERY | BIGINT | log_to_redo_for_recovery - Cantidad de anotaciones cronológicas a rehacer para la recuperación |
| LOG_HELD_BY_DIRTY_PAGES | BIGINT | log_held_by_dirty_pages - Cantidad de espacio de anotaciones cronológicas contabilizado por páginas sucias |
| PKG_CACHE_LOOKUPS | BIGINT | pkg_cache_lookups - Búsquedas de antememoria de paquete |
| PKG_CACHE_INSERTS | BIGINT | pkg_cache_inserts - Inserciones de antememoria de paquete |
| PKG_CACHE_NUM_OVERFLOWS | BIGINT | pkg_cache_num_overflows - Desbordamientos de antememoria de paquete |
| PKG_CACHE_SIZE_TOP | BIGINT | pkg_cache_size_top - Marca de límite superior de antememoria de paquete |
| APPL_SECTION_LOOKUPS | BIGINT | appl_section_lookups - Búsqueda de sección |
| APPL_SECTION_INSERTS | BIGINT | appl_section_inserts - Inserciones de sección |
| CAT_CACHE_LOOKUPS | BIGINT | cat_cache_lookups - Búsquedas de antememoria de catálogo |
| CAT_CACHE_INSERTS | BIGINT | cat_cache_inserts - Inserciones de antememoria de catálogo |
| CAT_CACHE_OVERFLOWS | BIGINT | cat_cache_overflows - Desbordamientos de antememoria de catálogo |

Tabla 256. Información devuelta por la función de tabla SNAP_GET_DB (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------------------|---------------|--|
| CAT_CACHE_SIZE_TOP | BIGINT | cat_cache_size_top - Marca de límite superior de antememoria de catálogo |
| PRIV_WORKSPACE_SIZE_TOP | BIGINT | priv_workspace_size_top - Tamaño máximo de espacio de trabajo privado |
| PRIV_WORKSPACE_NUM_OVERFLOWS | BIGINT | priv_workspace_num_overflows - Desbordamientos de espacio de trabajo privado |
| PRIV_WORKSPACE_SECTION_INSERTS | BIGINT | priv_workspace_section_inserts - Inserciones de sección de espacio de trabajo privado |
| PRIV_WORKSPACE_SECTION_LOOKUPS | BIGINT | priv_workspace_section_lookups - Búsquedas de sección de espacio de trabajo privado |
| SHR_WORKSPACE_SIZE_TOP | BIGINT | shr_workspace_size_top - Tamaño máximo de espacio de trabajo compartido |
| SHR_WORKSPACE_NUM_OVERFLOWS | BIGINT | shr_workspace_num_overflows - Desbordamientos de espacio de trabajo compartido |
| SHR_WORKSPACE_SECTION_INSERTS | BIGINT | shr_workspace_section_inserts - Inserciones de sección de espacio de trabajo compartido |
| SHR_WORKSPACE_SECTION_LOOKUPS | BIGINT | shr_workspace_section_lookups - Búsquedas de sección de espacio de trabajo compartido |
| TOTAL_HASH_JOINS | BIGINT | total_hash_joins - Total de uniones hash |
| TOTAL_HASH_LOOPS | BIGINT | total_hash_loops - Total de bucles hash |
| HASH_JOIN_OVERFLOWS | BIGINT | hash_join_overflows - Desbordamientos de uniones hash |
| HASH_JOIN_SMALL_OVERFLOWS | BIGINT | hash_join_small_overflows - Desbordamientos pequeños de uniones hash |
| NUM_DB_STORAGE_PATHS | BIGINT | num_db_storage_paths - Número de vías de acceso de almacenamiento automático |
| DBPARTITIONNUM | SMALLINT | node_number - Número de nodo |

Función de tabla SNAP_GET_DBM – Recuperación de la información de instantáneas del grupo de datos lógicos dbm

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la “Vista administrativa SNAPDBM y función de tabla SNAP_GET_DBM_V95 – Recuperar la información de instantáneas del grupo de datos lógicos dbm” en la página 572.

La tabla de función SNAP_GET_DBM devuelve la información de los grupos lógicos del gestor de bases de datos (dbm) DB2 y del supervisor de instantáneas.

La función de tabla SNAP_GET_DBM, utilizada con las funciones de tabla SNAP_GET_DBM_MEMORY_POOL, SNAP_GET_FCM, SNAP_GET_FCM_PART y SNAP_GET_SWITCHES, ofrece los datos equivalentes al mandato GET SNAPSHOT FOR DBM.

Consulte la Tabla 257 en la página 1081 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
▶▶ SNAP_GET_DBM ( ( númparticiónbd ) ) ▶▶
```

El esquema es SYSPROC.

Parámetro de la función de tabla

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si no se utiliza esta opción de entrada, se devolverán datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *núm_partición_bd* se establece en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_DBM toma la instantánea de la memoria.

Autorización

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP_GET_DBM.

Ejemplo

Recuperar la hora de inicio y el estado actual del número 2 de partición de base de datos.

```
SELECT DB2START_TIME, DB2_STATUS FROM TABLE(SNAP_GET_DBM(2)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
DB2START_TIME          DB2_STATUS
-----
2006-01-06-14.59.59.062798 ACTIVE
```

Información devuelta

Tabla 257. Información devuelta por la tabla de función SNAP_GET_DBM

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| SORT_HEAP_ALLOCATED | BIGINT | sort_heap_allocated - Total de pila de clasificación asignada |
| POST_THRESHOLD_SORTS | BIGINT | post_threshold_sorts - Clasificaciones de umbral posteriores |
| PIPED_SORTS_REQUESTED | BIGINT | piped_sorts_requested - Clasificaciones mediante conexiones solicitadas |
| PIPED_SORTS_ACCEPTED | BIGINT | piped_sorts_accepted - Clasificaciones mediante conexiones aceptadas |
| REM_CONS_IN | BIGINT | rem_cons_in - Conexiones remotas con gestor de bases de datos |
| REM_CONS_IN_EXEC | BIGINT | rem_cons_in_exec - Conexiones remotas que se ejecutan en el elemento de supervisor de gestor de bases de datos |
| LOCAL_CONS | BIGINT | local_cons - Conexiones locales |
| LOCAL_CONS_IN_EXEC | BIGINT | local_cons_in_exec - Conexiones locales que se ejecutan en el elemento de supervisor de gestor de bases de datos |
| CON_LOCAL_DBASES | BIGINT | con_local_dbases - Bases de datos locales con conexiones actuales |
| AGENTS_REGISTERED | BIGINT | agents_registered - Agentes registrados |
| AGENTS_WAITING_ON_TOKEN | BIGINT | agents_waiting_on_token - Agentes en espera de una señal |
| DB2_STATUS | VARCHAR(12) | db2_status - Estado de instancia de DB2. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • ACTIVE • QUIESCE_PEND • QUIESCED |
| AGENTS_REGISTERED_TOP | BIGINT | agents_registered_top - Número máximo de agentes registrados |
| AGENTS_WAITING_TOP | BIGINT | agents_waiting_top - Número máximo de agentes en espera |
| COMM_PRIVATE_MEM | BIGINT | comm_private_mem - Memoria privada confirmada |
| IDLE_AGENTS | BIGINT | idle_agents - Número de agentes desocupados |

Tabla 257. Información devuelta por la tabla de función SNAP_GET_DBM (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------------|---------------|---|
| AGENTS_FROM_POOL | BIGINT | agents_from_pool - Agentes asignados de la agrupación |
| AGENTS_CREATED_EMPTY_POOL | BIGINT | agents_created_empty_pool - Agentes creados porque la agrupación de agentes está vacía |
| COORD_AGENTS_TOP | BIGINT | coord_agents_top - Número máximo de agentes de coordinación |
| MAX_AGENT_OVERFLOW | BIGINT | max_agent_overflows - Número máximo de desbordamientos de agente |
| AGENTS_STOLEN | BIGINT | agents_stolen - Agentes robados |
| GW_TOTAL_CONS | BIGINT | gw_total_cons - Número total de conexiones intentadas para DB2 Connect |
| GW_CUR_CONS | BIGINT | gw_cur_cons - Número actual de conexiones para DB2 Connect |
| GW_CONS_WAIT_HOST | BIGINT | gw_cons_wait_host - Número de conexiones en espera para que conteste el sistema principal |
| GW_CONS_WAIT_CLIENT | BIGINT | gw_cons_wait_client - Número de conexiones en espera para que el cliente envíe petición |
| POST_THRESHOLD_HASH_JOINS | BIGINT | post_threshold_hash_joins - Umbral de uniones hash |
| NUM_GW_CONN_SWITCHES | BIGINT | num_gw_conn_switches - Conmutadores de conexión |
| DB2START_TIME | TIMESTAMP | db2start_time - Indicación de fecha y hora de inicio de gestor de bases de datos |
| LAST_RESET | TIMESTAMP | last_reset - Indicación de fecha y hora de último restablecimiento |
| NUM_NODES_IN_DB2_INSTANCE | INTEGER | num_nodes_in_db2_instance - Número de nodos en la partición de base de datos |
| PRODUCT_NAME | VARCHAR(32) | product_name - Nombre del producto |
| SERVICE_LEVEL | VARCHAR(18) | service_level - Nivel de servicio |
| SORT_HEAP_TOP | BIGINT | sort_heap_top - Marca de límite superior de pila privada de clasificación |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Función de tabla SNAP_GET_DB_V91 - Recuperar información de instantánea desde el grupo lógico dbase

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la “Vista administrativa SNAPDB y función de tabla SNAP_GET_DB_V95 – Recuperar información de instantánea desde el grupo lógico dbase” en la página 554.

La función de tabla SNAP_GET_DB_V91 devuelve información de instantánea del grupo de datos lógicos de base de datos (dbase).

La función de tabla SNAP_GET_DB_V91, utilizada con las funciones de tabla SNAP_GET_DB_MEMORY_POOL, SNAP_GET_DETAILLOG_V91, SNAP_GET_HADR y SNAP_GET_STORAGE_PATHS, proporciona información que es equivalente a la que proporciona el mandato CLP GET SNAPSHOT FOR ALL DATABASES.

En la Tabla 258 en la página 1085 encontrará una lista completa de la información que se devuelve.

Sintaxis

```
▶▶ SNAP_GET_DB_V91 ( ( nombre-bd [ , numparticiónbd ] ) ) ▶▶
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique una serie vacía para tomar la instantánea desde la base de datos conectada actualmente. Especifique un valor NULL para tomar la instantánea desde todas las bases de datos que se encuentran en la misma instancia que la base de datos conectada actualmente.

numparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *numparticiónbd* se establece en NULL, -1 se establece implícitamente para *numparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *numparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_DB_V91 tomará una instantánea del número de partición de base de

datos y la base de datos conectada actualmente.

Autorización

Se necesita una de las autorizaciones o uno de los privilegios siguientes:

- Privilegio EXECUTE para la función de tabla SNAP_GET_DB_V91.
- Autorización DATAACCESS
- Autorización SYSMON
- Autorización SYSMAINT
- Autorización SYSCTRL
- Autorización SYSADM

Ejemplos

Ejemplo 1: Recuperar el estado, la plataforma, la ubicación y el tiempo de conexión como vista de agregación de todas las particiones de base de datos de la base de datos conectada actualmente.

```
SELECT SUBSTR(DB_NAME, 1, 20) AS DB_NAME, DB_STATUS, SERVER_PLATFORM,
       DB_LOCATION, DB_CONN_TIME FROM TABLE(SNAP_GET_DB_V91(' ', -2)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| DB_NAME | DB_STATUS | SERVER_PLATFORM | ... |
|---------|-----------|-----------------|-----|
| ... | ----- | ----- | ... |
| SAMPLE | ACTIVE | AIX64 | ... |

1 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

| ... | DB_LOCATION | DB_CONN_TIME |
|-----|-------------|----------------------------|
| ... | ----- | ----- |
| ... | LOCAL | 2005-07-24-22.09.22.013196 |

Ejemplo 2: Recuperar el estado, la plataforma, la ubicación y el tiempo de conexión como vista de agregación de todas las particiones de base de datos para todas las bases de datos activas que se encuentran en la misma instancia que contiene la base de datos conectada actualmente.

```
SELECT SUBSTR(DB_NAME, 1, 20) AS DB_NAME, DB_STATUS, SERVER_PLATFORM,
       DB_LOCATION, DB_CONN_TIME
FROM TABLE(SNAP_GET_DB_V91(CAST (NULL AS VARCHAR(128)), -2)) AS T
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| DB_NAME | DB_STATUS | SERVER_PLATFORM | ... |
|---------|-----------|-----------------|-----|
| ... | ----- | ----- | ... |
| TOOLSDB | ACTIVE | AIX64 | ... |
| SAMPLE | ACTIVE | AIX64 | ... |

Datos de salida de esta consulta (continuación).

| ... | DB_LOCATION | DB_CONN_TIME |
|-----|-------------|----------------------------|
| ... | ----- | ----- |
| ... | LOCAL | 2005-07-24-22.26.54.396335 |
| ... | LOCAL | 2005-07-24-22.09.22.013196 |

Metadatos de la función de tabla SNAP_GET_DB_V91

Tabla 258. Información devuelta por la función de tabla SNAP_GET_DB_V91

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| DB_PATH | VARCHAR(1024) | db_path - Vía de acceso de base de datos |
| INPUT_DB_ALIAS | VARCHAR(128) | input_db_alias - Alias de base de datos de entrada |
| DB_STATUS | VARCHAR(12) | db_status - Estado de base de datos. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none">• ACTIVE• QUIESCE_PEND• QUIESCED• ROLLFWD |
| CATALOG_PARTITION | SMALLINT | catalog_node - Número de nodo de catálogo |
| CATALOG_PARTITION_NAME | VARCHAR(128) | catalog_node_name - Nombre de red de nodo de catálogo |

Tabla 258. Información devuelta por la función de tabla
SNAP_GET_DB_V91 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|--|
| SERVER_PLATFORM | VARCHAR(12) | <p>server_platform - Sistema operativo del servidor. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes:</p> <ul style="list-style-type: none"> • AIX • AIX64 • AS400_DRDA • DOS • DYNIX • HP • HP64 • HPIA • HPIA64 • LINUX • LINUX390 • LINUXIA64 • LINUXPPC • LINUXPPC64 • LINUXX8664 • LINUXZ64 • MAC • MVS_DRDA • NT • NT64 • OS2 • OS390 • SCO • SGI • SNI • SUN • SUN64 • UNKNOWN • UNKNOWN_DRDA • VM_DRDA • VSE_DRDA • WINDOWS • WINDOWS95 |

Tabla 258. Información devuelta por la función de tabla SNAP_GET_DB_V91 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|---|
| DB_LOCATION | VARCHAR(12) | db_location - Ubicación de la base de datos. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> • LOCAL • REMOTE |
| DB_CONN_TIME | TIMESTAMP | db_conn_time - Indicación de fecha y hora de activación de base de datos |
| LAST_RESET | TIMESTAMP | last_reset - Indicación de fecha y hora de último restablecimiento |
| LAST_BACKUP | TIMESTAMP | last_backup - Indicación de fecha y hora de última copia de seguridad |
| CONNECTIONS_TOP | BIGINT | connections_top - Número máximo de conexiones simultáneas |
| TOTAL_CONS | BIGINT | total_cons - Conexiones desde la activación de base de datos |
| TOTAL_SEC_CONS | BIGINT | total_sec_cons - Conexiones secundarias |
| APPLS_CUR_CONS | BIGINT | appls_cur_cons - Aplicaciones conectadas actualmente |
| APPLS_IN_DB2 | BIGINT | appls_in_db2 - Aplicaciones que se ejecutan en la base de datos actualmente |
| NUM_ASSOC_AGENTS | BIGINT | num_assoc_agents - Número de agentes asociados |
| AGENTS_TOP | BIGINT | agents_top - Número de agentes creados |
| COORD_AGENTS_TOP | BIGINT | coord_agents_top - Número máximo de agentes de coordinación |
| LOCKS_HELD | BIGINT | locks_held - Bloqueos retenidos |
| LOCK_WAITS | BIGINT | lock_waits - Esperas de bloqueo |
| LOCK_WAIT_TIME | BIGINT | lock_wait_time - Tiempo esperado en bloqueos |
| LOCK_LIST_IN_USE | BIGINT | lock_list_in_use - Memoria total de lista de bloqueos en uso |
| DEADLOCKS | BIGINT | deadlocks - Puntos muertos detectados |
| LOCK_ESCALS | BIGINT | lock_escals - Número de escalamientos de bloqueo |
| X_LOCK_ESCALS | BIGINT | x_lock_escals - Escalas de bloqueo exclusivas |
| LOCKS_WAITING | BIGINT | locks_waiting - Agentes actuales en espera de bloqueos |

Tabla 258. Información devuelta por la función de tabla SNAP_GET_DB_V91 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|---|
| LOCK_TIMEOUTS | BIGINT | lock_timeouts - Número de tiempos de espera de bloqueo |
| NUM_INDOUBT_TRANS | BIGINT | num_indoubt_trans - Número de transacciones dudosas |
| SORT_HEAP_ALLOCATED | BIGINT | sort_heap_allocated - Total de pila de clasificación asignada |
| SORT_SHRHEAP_ALLOCATED | BIGINT | sort_shrheap_allocated - Pila de compartimiento de clasificación asignada actualmente |
| SORT_SHRHEAP_TOP | BIGINT | sort_shrheap_top - Marca de límite superior de pila de compartimiento de clasificación |
| POST_SHRTHRESHOLD_SORTS | BIGINT | post_shrthreshold_sorts - Clasificaciones de umbral compartidas posteriores |
| TOTAL_SORTS | BIGINT | total_sorts - Clasificaciones totales |
| TOTAL_SORT_TIME | BIGINT | total_sort_time - Tiempo de clasificación total |
| SORT_OVERFLOWS | BIGINT | sort_overflows - Desbordamientos de clasificación |
| ACTIVE_SORTS | BIGINT | active_sorts - Clasificaciones activas |
| POOL_DATA_L_READS | BIGINT | pool_data_l_reads - Lecturas lógicas de datos de la agrupación de almacenamientos intermedios |
| POOL_DATA_P_READS | BIGINT | pool_data_p_reads - Lecturas físicas de datos de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_L_READS | BIGINT | pool_temp_data_l_reads - Lecturas lógicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_P_READS | BIGINT | pool_temp_data_p_reads - Lecturas físicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_READS | BIGINT | pool_async_data_reads - Lecturas de datos asíncronas de agrupación de almacenamientos intermedios |
| POOL_DATA_WRITES | BIGINT | pool_data_writes - Grabaciones de datos de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_WRITES | BIGINT | pool_async_data_writes - Grabaciones de datos asíncronas de agrupación de almacenamientos intermedios |

Tabla 258. Información devuelta por la función de tabla SNAP_GET_DB_V91 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|--|
| POOL_INDEX_L_READS | BIGINT | pool_index_l_reads - Lecturas lógicas de índices de la agrupación de almacenamientos intermedios |
| POOL_INDEX_P_READS | BIGINT | pool_index_p_reads - Lecturas físicas de índices de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_L_READS | BIGINT | pool_temp_index_l_reads - Lecturas lógicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_P_READS | BIGINT | pool_temp_index_p_reads - Lecturas físicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_READS | BIGINT | pool_async_index_reads - Lecturas de índice asíncronas de agrupación de almacenamientos intermedios |
| POOL_INDEX_WRITES | BIGINT | pool_index_writes - Grabaciones de índices de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_WRITES | BIGINT | pool_async_index_writes - Grabaciones de índice asíncronas de agrupación de almacenamientos intermedios |
| POOL_XDA_P_READS | BIGINT | pool_xda_p_reads - Lecturas físicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_XDA_L_READS | BIGINT | pool_xda_l_reads - Lecturas lógicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_XDA_WRITES | BIGINT | pool_xda_writes - Grabaciones de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_XDA_READS | BIGINT | pool_async_xda_reads - Lecturas de datos XDA asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_XDA_WRITES | BIGINT | pool_async_xda_writes - Grabaciones de datos XDA asíncronas de agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_P_READS | BIGINT | pool_temp_xda_p_reads - Elemento del supervisor de las lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |

Tabla 258. Información devuelta por la función de tabla SNAP_GET_DB_V91 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|----------------------------|---------------|---|
| POOL_TEMP_XDA_L_READS | BIGINT | pool_temp_xda_l_reads - Lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |
| POOL_READ_TIME | BIGINT | pool_read_time - Tiempo total de lectura física de la agrupación de almacenamientos intermedios |
| POOL_WRITE_TIME | BIGINT | pool_write_time - Tiempo total de grabación física de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_READ_TIME | BIGINT | pool_async_read_time - Tiempo de lectura asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_WRITE_TIME | BIGINT | pool_async_write_time - Tiempo de grabación asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_READ_REQS | BIGINT | pool_async_data_read_reqs - Peticiones de lectura asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_READ_REQS | BIGINT | pool_async_index_read_reqs - Peticiones de lectura de índice asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_XDA_READ_REQS | BIGINT | pool_async_xda_read_reqs - Peticiones de lectura XDA asíncrona de agrupación de almacenamientos intermedios |
| POOL_NO_VICTIM_BUFFER | BIGINT | pool_no_victim_buffer - Almacenamiento intermedio sin víctimas de la agrupación de almacenamientos intermedios |
| POOL_LSN_GAP_CLNS | BIGINT | pool_lsn_gap_clns - Limpiadores de espacio de anotación cronológica de agrupación de almacenamientos intermedios activados |
| POOL_DRTY_PG_STEAL_CLNS | BIGINT | pool_drty_pg_steal_clns - Limpiadores de páginas víctima de agrupación de agrupación de almacenamientos intermedios activados |
| POOL_DRTY_PG_THRSH_CLNS | BIGINT | pool_drty_pg_thrsh_clns - Limpiadores de umbral de agrupación de almacenamientos intermedios activados |
| PREFETCH_WAIT_TIME | BIGINT | prefetch_wait_time - Tiempo esperado para captación previa |

Tabla 258. Información devuelta por la función de tabla SNAP_GET_DB_V91 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------|---------------|--|
| UNREAD_PREFETCH_PAGES | BIGINT | unread_prefetch_pages - Páginas de captación previa no leídas |
| DIRECT_READS | BIGINT | direct_reads - Lecturas directas de la base de datos |
| DIRECT_WRITES | BIGINT | direct_writes - Grabaciones directas en la base de datos |
| DIRECT_READ_REQS | BIGINT | direct_read_reqs - Peticiones de lectura directa |
| DIRECT_WRITE_REQS | BIGINT | direct_write_reqs - Peticiones de grabación directa |
| DIRECT_READ_TIME | BIGINT | direct_read_time - Tiempo de lectura directa |
| DIRECT_WRITE_TIME | BIGINT | direct_write_time - Tiempo de grabación directa |
| FILES_CLOSED | BIGINT | files_closed - Archivos de base de datos cerrados |
| ELAPSED_EXEC_TIME_S | BIGINT | elapsed_exec_time - Tiempo transcurrido de ejecución de sentencia |
| ELAPSED_EXEC_TIME_MS | BIGINT | elapsed_exec_time - Tiempo transcurrido de ejecución de sentencia |
| COMMIT_SQL_STMTS | BIGINT | commit_sql_stmts - Sentencias de confirmación intentadas |
| ROLLBACK_SQL_STMTS | BIGINT | rollback_sql_stmts - Sentencias de retrotracción intentadas |
| DYNAMIC_SQL_STMTS | BIGINT | dynamic_sql_stmts - Sentencias de SQL dinámico intentadas |
| STATIC_SQL_STMTS | BIGINT | static_sql_stmts - Sentencias de SQL estático intentadas |
| FAILED_SQL_STMTS | BIGINT | failed_sql_stmts - Operaciones de sentencia anómalas |
| SELECT_SQL_STMTS | BIGINT | select_sql_stmts - Sentencias Select de SQL ejecutadas |
| UID_SQL_STMTS | BIGINT | uid_sql_stmts - Sentencias UPDATE/INSERT/DELETE de SQL ejecutadas |
| DDL_SQL_STMTS | BIGINT | ddl_sql_stmts - Sentencias de lenguaje de definición de datos (DDL) de SQL |
| INT_AUTO_REBINDS | BIGINT | int_auto_rebinds - Revinculaciones automáticas internas |
| INT_ROWS_DELETED | BIGINT | int_rows_deleted - Filas internas suprimidas |
| INT_ROWS_INSERTED | BIGINT | int_rows_inserted - Filas internas insertadas |

Tabla 258. Información devuelta por la función de tabla SNAP_GET_DB_V91 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------|---------------|---|
| INT_ROWS_UPDATED | BIGINT | int_rows_updated - Filas internas actualizadas |
| INT_COMMITS | BIGINT | int_commits - Confirmaciones internas |
| INT_ROLLBACKS | BIGINT | int_rollback - Retrotracciones internas |
| INT_DEADLOCK_ROLLBACKS | BIGINT | int_deadlock_rollback - Retrotracciones internas debido a punto muerto |
| ROWS_DELETED | BIGINT | rows_deleted - Filas suprimidas |
| ROWS_INSERTED | BIGINT | rows_inserted - Filas insertadas |
| ROWS_UPDATED | BIGINT | rows_updated - Filas actualizadas |
| ROWS_SELECTED | BIGINT | rows_selected - Filas seleccionadas |
| ROWS_READ | BIGINT | rows_read - Filas leídas |
| BINDS_PRECOMPILES | BIGINT | binds_precompiles - Vinculaciones/precompilaciones intentadas |
| TOTAL_LOG_AVAILABLE | BIGINT | total_log_available - Anotaciones cronológicas totales disponibles |
| TOTAL_LOG_USED | BIGINT | total_log_used - Espacio total de anotaciones cronológicas utilizado |
| SEC_LOG_USED_TOP | BIGINT | sec_log_used_top - Espacio máximo de anotaciones cronológicas secundarias utilizado |
| TOT_LOG_USED_TOP | BIGINT | tot_log_used_top - Espacio máximo de anotaciones cronológicas totales utilizado |
| SEC_LOGS_ALLOCATED | BIGINT | sec_logs_allocated - Anotaciones cronológicas secundarias asignadas actualmente |
| LOG_READS | BIGINT | log_reads - Número de páginas de anotaciones cronológicas leídas |
| LOG_READ_TIME_S | BIGINT | log_read_time - Tiempo de lectura de anotaciones cronológicas |
| LOG_READ_TIME_NS | BIGINT | log_read_time - Tiempo de lectura de anotaciones cronológicas |
| LOG_WRITES | BIGINT | log_writes - Número de páginas de anotaciones cronológicas escritas |
| LOG_WRITE_TIME_S | BIGINT | log_write_time - Tiempo de grabación de anotaciones cronológicas |
| LOG_WRITE_TIME_NS | BIGINT | log_write_time - Tiempo de grabación de anotaciones cronológicas |

Tabla 258. Información devuelta por la función de tabla SNAP_GET_DB_V91 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------------|---------------|--|
| NUM_LOG_WRITE_IO | BIGINT | num_log_write_io - Número de grabaciones de anotaciones cronológicas |
| NUM_LOG_READ_IO | BIGINT | num_log_read_io - Número de lecturas de anotaciones cronológicas |
| NUM_LOG_PART_PAGE_IO | BIGINT | num_log_part_page_io - Número de grabaciones de páginas de anotaciones cronológicas parciales |
| NUM_LOG_BUFFER_FULL | BIGINT | num_log_buffer_full - Número de almacenamientos intermedios de anotaciones cronológicas llenos |
| NUM_LOG_DATA_FOUND_IN_BUFFER | BIGINT | num_log_data_found_in_buffer - Número de datos de anotaciones cronológicas encontrados en el almacenamiento intermedio |
| APPL_ID_OLDEST_XACT | BIGINT | appl_id_oldest_xact - Aplicación con la transacción más antigua |
| LOG_TO_REDO_FOR_RECOVERY | BIGINT | log_to_redo_for_recovery - Cantidad de anotaciones cronológicas a rehacer para la recuperación |
| LOG_HELD_BY_DIRTY_PAGES | BIGINT | log_held_by_dirty_pages - Cantidad de espacio de anotaciones cronológicas contabilizado por páginas sucias |
| PKG_CACHE_LOOKUPS | BIGINT | pkg_cache_lookups - Búsquedas de antememoria de paquete |
| PKG_CACHE_INSERTS | BIGINT | pkg_cache_inserts - Inserciones de antememoria de paquete |
| PKG_CACHE_NUM_OVERFLOWS | BIGINT | pkg_cache_num_overflows - Desbordamientos de antememoria de paquete |
| PKG_CACHE_SIZE_TOP | BIGINT | pkg_cache_size_top - Marca de límite superior de antememoria de paquete |
| APPL_SECTION_LOOKUPS | BIGINT | appl_section_lookups - Búsquedas de sección |
| APPL_SECTION_INSERTS | BIGINT | appl_section_inserts - Inserciones de sección |
| CAT_CACHE_LOOKUPS | BIGINT | cat_cache_lookups - Búsquedas de antememoria de catálogo |
| CAT_CACHE_INSERTS | BIGINT | cat_cache_inserts - Inserciones de antememoria de catálogo |
| CAT_CACHE_OVERFLOWS | BIGINT | cat_cache_overflows - Desbordamientos de antememoria de catálogo |

Tabla 258. Información devuelta por la función de tabla SNAP_GET_DB_V91 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------------------|---------------|---|
| CAT_CACHE_SIZE_TOP | BIGINT | cat_cache_size_top - Marca de límite superior de antememoria de catálogo |
| PRIV_WORKSPACE_SIZE_TOP | BIGINT | priv_workspace_size_top - Tamaño máximo de espacio de trabajo privado |
| PRIV_WORKSPACE_NUM_OVERFLOWS | BIGINT | priv_workspace_num_overflows - Desbordamientos de espacio de trabajo privado |
| PRIV_WORKSPACE_SECTION_INSERTS | BIGINT | priv_workspace_section_inserts - Inserciones de sección de espacio de trabajo privado |
| PRIV_WORKSPACE_SECTION_LOOKUPS | BIGINT | priv_workspace_section_lookups - Búsquedas de sección de espacio de trabajo privado |
| SHR_WORKSPACE_SIZE_TOP | BIGINT | shr_workspace_size_top - Tamaño máximo de espacio de trabajo compartido |
| SHR_WORKSPACE_NUM_OVERFLOWS | BIGINT | shr_workspace_num_overflows - Desbordamientos de espacio de trabajo compartido |
| SHR_WORKSPACE_SECTION_INSERTS | BIGINT | shr_workspace_section_inserts - Inserciones de sección de espacio de trabajo compartido |
| SHR_WORKSPACE_SECTION_LOOKUPS | BIGINT | shr_workspace_section_lookups - Búsquedas de sección de espacio de trabajo compartido |
| TOTAL_HASH_JOINS | BIGINT | total_hash_joins - Total de uniones hash |
| TOTAL_HASH_LOOPS | BIGINT | total_hash_loops - Total de bucles hash |
| HASH_JOIN_OVERFLOWS | BIGINT | hash_join_overflows - Desbordamientos de uniones hash |
| HASH_JOIN_SMALL_OVERFLOWS | BIGINT | hash_join_small_overflows - Desbordamientos pequeños de uniones hash |
| POST_SHRTHRESHOLD_HASH_JOINS | BIGINT | post_shrthreshold_hash_joins - Uniones hash de umbral posteriores |
| ACTIVE_HASH_JOINS | BIGINT | active_hash_joins - Uniones hash activas |
| NUM_DB_STORAGE_PATHS | BIGINT | num_db_storage_paths - Número de vías de acceso de almacenamiento automático |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

Tabla 258. Información devuelta por la función de tabla SNAP_GET_DB_V91 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|---|
| SMALLEST_LOG_AVAIL_NODE | INTEGER | smallest_log_avail_node - Nodo con menos espacio de anotaciones cronológicas disponible |

Función de tabla SNAP_GET_DYN_SQL_V91 - Recuperar información de la instantánea del grupo de datos lógicos dynsql

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la "Vista administrativa SNAPDYN_SQL y función de tabla SNAP_GET_DYN_SQL_V95 – Recuperar información de instantáneas del grupo de datos lógicos dynsql" en la página 583.

La función de tabla SNAP_GET_DYN_SQL_V91 devuelve información de instantáneas desde el grupo de datos lógicos dynsql.

Esta función de tabla devuelve información equivalente a la que proporciona el mandato CLP GET SNAPSHOT FOR DYNAMIC SQL ON alias de base de datos.

Consulte la Tabla 259 en la página 1097 para ver una lista completa de la información que se puede devolver.

Sintaxis

```
▶▶—SNAP_GET_DYN_SQL_V91—(—nombre-bd— [ , nóm-partición-bd ] )—▶▶
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombrebd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique NULL o una serie vacía para tomar la instantánea de la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada opcional de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para un agregado de todas las particiones de base de datos activas. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, -1 se establece implícitamente para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombrebd*, se devuelven datos de todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si *nombrebd* y *númparticiónbd* se establecen en NULL, se intenta leer los datos del archivo creado por el procedimiento SNAP_WRITE_FILE. Tenga en cuenta que este archivo podría haberse creado en cualquier momento, lo que significa que es posible que los datos no estén actualizados. Si no existe un archivo con el correspondiente tipo de petición de API de instantánea, la función de tabla SNAP_GET_DYN_SQL_V91 tomará una instantánea del número de partición de base de datos y la base de datos conectada actualmente.

Autorización

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP_GET_DYN_SQL_V91.

Ejemplo

Recuperar una lista de sentencias de SQL dinámico ejecutadas en la partición de base de datos correspondiente a la base de datos conectada actualmente, ordenadas por el número de filas leídas.

```
SELECT PREP_TIME_WORST, NUM_COMPILATIONS, SUBSTR(STMT_TEXT, 1, 60)
      AS STMT_TEXT FROM TABLE(SNAP_GET_DYN_SQL_V91('','-1)) as T
      ORDER BY ROWS_READ
```

El ejemplo siguiente muestra los datos de salida de la consulta.

```
PREP_TIME_WORST      ...
-----            ...
                   0 ...
                   3 ...
                   ...
                   4 ...
                   ...
                   4 ...
                   ...
                   4 ...
                   ...
                   3 ...
                   ...
                   4 ...
                   ...
```

Datos de salida de esta consulta (continuación).

```
... NUM_COMPILATIONS  STMT_TEXT
... -----
...                   0 SET CURRENT LOCALE LC_CTYPE = 'en_US'
...                   1 select rows_read, rows_written,
...                     substr(stmt_text, 1, 40) as
...                   1 select * from table
...                     (snap_get_dyn_sqlv9('','-1)) as t
...                   1 select * from table
...                     (snap_getdetaillog9('','-1)) as t
...                   1 select * from table
...                     (snap_get_hadr('','-1)) as t
...                   1 select prep_time_worst, num_compilations,
...                     substr(stmt_text,
...                   1 select prep_time_worst, num_compilations,
...                     substr(stmt_text,
```

Información devuelta

Tabla 259. Información devuelta por la función de tabla SNAP_GET_DYN_SQL_V91

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| NUM_EXECUTIONS | BIGINT | num_executions - Ejecuciones de sentencia |
| NUM_COMPILATIONS | BIGINT | num_compilations - Compilaciones de sentencia |
| PREP_TIME_WORST | BIGINT | prep_time_worst - Peor tiempo de preparación de sentencia |
| PREP_TIME_BEST | BIGINT | prep_time_best - Mejor tiempo de preparación de sentencia |
| INT_ROWS_DELETED | BIGINT | int_rows_deleted - Filas internas suprimidas |
| INT_ROWS_INSERTED | BIGINT | int_rows_inserted - Filas internas insertadas |
| INT_ROWS_UPDATED | BIGINT | int_rows_updated - Filas internas actualizadas |
| ROWS_READ | BIGINT | rows_read - Filas leídas |
| ROWS_WRITTEN | BIGINT | rows_written - Filas grabadas |
| STMT_SORTS | BIGINT | stmt_sorts - Clasificaciones de sentencias |
| SORT_OVERFLOWS | BIGINT | sort_overflows - Desbordamientos de clasificación |
| TOTAL_SORT_TIME | BIGINT | total_sort_time - Tiempo de clasificación total |
| POOL_DATA_L_READS | BIGINT | pool_data_l_reads - Lecturas lógicas de datos de la agrupación de almacenamientos intermedios |
| POOL_DATA_P_READS | BIGINT | pool_data_p_reads - Lecturas físicas de datos de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_L_READS | BIGINT | pool_temp_data_l_reads - Lecturas lógicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_P_READS | BIGINT | pool_temp_data_p_reads - Lecturas físicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_INDEX_L_READS | BIGINT | pool_index_l_reads - Lecturas lógicas de índices de la agrupación de almacenamientos intermedios |
| POOL_INDEX_P_READS | BIGINT | pool_index_p_reads - Lecturas físicas de índices de la agrupación de almacenamientos intermedios |

Tabla 259. Información devuelta por la función de tabla SNAP_GET_DYN_SQL_V91 (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|--|
| POOL_TEMP_INDEX_L_READS | BIGINT | pool_temp_index_l_reads - Lecturas lógicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_P_READS | BIGINT | pool_temp_index_p_reads - Lecturas físicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_XDA_L_READS | BIGINT | pool_xda_l_reads - Lecturas lógicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_XDA_P_READS | BIGINT | pool_xda_p_reads - Lecturas físicas de datos de XDA de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_L_READS | BIGINT | pool_temp_xda_l_reads - Lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |
| POOL_TEMP_XDA_P_READS | BIGINT | pool_temp_xda_p_reads - Elemento del supervisor de las lecturas lógicas de datos temporales de XDA de la agrupación de almacenamientos intermedios |
| TOTAL_EXEC_TIME | BIGINT | total_exec_time - Tiempo transcurrido de ejecución de sentencia |
| TOTAL_EXEC_TIME_MS | BIGINT | total_exec_time - Tiempo transcurrido de ejecución de sentencia |
| TOTAL_USR_CPU_TIME | BIGINT | total_usr_cpu_time - Total de CPU de usuario para una sentencia |
| TOTAL_USR_CPU_TIME_MS | BIGINT | total_usr_cpu_time - Total de CPU de usuario para una sentencia |
| TOTAL_SYS_CPU_TIME | BIGINT | total_sys_cpu_time - Total de CPU de sistema para una sentencia |
| TOTAL_SYS_CPU_TIME_MS | BIGINT | total_sys_cpu_time - Total de CPU de sistema para una sentencia |
| STMT_TEXT | CLOB(2 M) | stmt_text - Texto de sentencia de SQL |
| DBPARTITIONNUM | SMALLINT | Partición de base de datos desde la que se han recuperado los datos para esta fila. |

SNAP_GET_DYN_SQL

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la "Función de tabla SNAP_GET_DYN_SQL_V91 - Recuperar información de la instantánea del grupo de datos lógicos dynsql" en la página 1095

►►—SNAP_GET_DYN_SQL—(—nombre-bd—,—númparticiónbd—)—————►►

El esquema es SYSPROC.

La función de tabla SNAP_GET_DYN_SQL devuelve información de instantáneas desde el grupo de datos lógicos dynsql.

nombrebd

Argumento de entrada de tipo VARCHAR(255) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique el valor nulo para tomar la instantánea desde la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

Si ambos parámetros se establecen en NULL, la instantánea sólo se tomará si el procedimiento almacenado SNAPSHOT_FILEW no ha creado anteriormente un archivo para el tipo de petición de API de instantánea correspondiente.

La función devuelve una tabla tal como se muestra más abajo.

Tabla 260. Información devuelta por la función de tabla SNAP_GET_DYN_SQL

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| NUM_EXECUTIONS | BIGINT | num_executions - Ejecuciones de sentencia |
| NUM_COMPILATIONS | BIGINT | num_compilations - Compilaciones de sentencia |
| PREP_TIME_WORST | BIGINT | prep_time_worst - Peor tiempo de preparación de sentencia |
| PREP_TIME_BEST | BIGINT | prep_time_best - Mejor tiempo de preparación de sentencia |
| INT_ROWS_DELETED | BIGINT | int_rows_deleted - Filas internas suprimidas |

Tabla 260. Información devuelta por la función de tabla SNAP_GET_DYN_SQL (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|--|
| INT_ROWS_INSERTED | BIGINT | int_rows_inserted - Filas internas insertadas |
| INT_ROWS_UPDATED | BIGINT | int_rows_updated - Filas internas actualizadas |
| ROWS_READ | BIGINT | rows_read - Filas leídas |
| ROWS_WRITTEN | BIGINT | rows_written - Filas grabadas |
| STMT_SORTS | BIGINT | stmt_sorts - Clasificaciones de sentencias |
| SORT_OVERFLOWS | BIGINT | sort_overflows - Desbordamientos de clasificación |
| TOTAL_SORT_TIME | BIGINT | total_sort_time - Tiempo de clasificación total |
| POOL_DATA_L_READS | BIGINT | pool_data_l_reads - Lecturas lógicas de datos de la agrupación de almacenamientos intermedios |
| POOL_DATA_P_READS | BIGINT | pool_data_p_reads - Lecturas físicas de datos de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_L_READS | BIGINT | pool_temp_data_l_reads - Lecturas lógicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_P_READS | BIGINT | pool_temp_data_p_reads - Lecturas físicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_INDEX_L_READS | BIGINT | pool_index_l_reads - Lecturas lógicas de índices de la agrupación de almacenamientos intermedios |
| POOL_INDEX_P_READS | BIGINT | pool_index_p_reads - Lecturas físicas de índices de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_L_READS | BIGINT | pool_temp_index_l_reads - Lecturas lógicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_P_READS | BIGINT | pool_temp_index_p_reads - Lecturas físicas de índices temporales de la agrupación de almacenamientos intermedios |
| TOTAL_EXEC_TIME | BIGINT | total_exec_time - Tiempo transcurrido de ejecución de sentencia |
| TOTAL_EXEC_TIME_MS | BIGINT | total_exec_time - Tiempo transcurrido de ejecución de sentencia |
| TOTAL_USR_TIME | BIGINT | total_usr_cpu_time - Total de CPU de usuario para una sentencia |

Tabla 260. Información devuelta por la función de tabla SNAP_GET_DYN_SQL (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|---|
| TOTAL_USR_TIME_MS | BIGINT | total_usr_cpu_time - Total de CPU de usuario para una sentencia |
| TOTAL_SYS_TIME | BIGINT | total_sys_cpu_time - Total de CPU de sistema para una sentencia |
| TOTAL_SYS_TIME_MS | BIGINT | total_sys_cpu_time - Total de CPU de sistema para una sentencia |
| STMT_TEXT | CLOB | stmt_text - Texto de sentencia de SQL |

SNAP_GET_STO_PATHS

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la "Vista administrativa SNAPSTORAGE_PATHS y función de tabla SNAP_GET_STORAGE_PATHS – Recuperar información de vía de acceso del almacenamiento automático" en la página 770

►► SNAP_GET_STO_PATHS (—nombre-bd—, —númparticiónbd—) ◀◀

El esquema es SYSPROC.

La función de tabla SNAP_GET_STO_PATHS devuelve información de instantáneas del grupo de datos lógicos storage_paths.

nombrebd

Argumento de entrada de tipo VARCHAR(255) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique el valor NULL para tomar la instantánea de la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

Si ambos parámetros se establecen en NULL, la instantánea sólo se tomará si el procedimiento almacenado SNAPSHOT_FILEW no ha creado anteriormente un archivo para el tipo de petición de API de instantánea correspondiente.

La función devuelve una tabla tal como se muestra más abajo.

Tabla 261. Información devuelta por la función de tabla *SNAP_GET_STO_PATHS*

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| DB_STORAGE_PATH | VARCHAR(256) | db_storage_path - Vía de acceso de almacenamiento automático |

SNAP_GET_TAB

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la "Vista administrativa SNAPTAB y función de tabla *SNAP_GET_TAB_V91* - Recuperar información de instantáneas del grupo de datos lógicos table" en la página 628

►► *SNAP_GET_TAB* (—*nombre-bd*—, —*númparticiónbd*—) ◄◄

El esquema es SYSPROC.

La función de tabla *SNAP_GET_TAB* devuelve información de instantáneas desde el grupo de datos lógicos table.

nombrebd

Argumento de entrada de tipo VARCHAR(255) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique el valor NULL para tomar la instantánea de la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

Si ambos parámetros se establecen en NULL, la instantánea sólo se tomará si el procedimiento almacenado *SNAPSHOT_FILEW* no ha creado anteriormente un archivo para el tipo de petición de API de instantánea correspondiente.

La función devuelve una tabla tal como se muestra más abajo.

Tabla 262. Información devuelta por la función de tabla *SNAP_GET_TAB*

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |

Tabla 262. Información devuelta por la función de tabla SNAP_GET_TAB (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| TABSCHEMA | VARCHAR(128) | table_schema - Nombre de esquema de tabla |
| TABNAME | VARCHAR(128) | table_name - Nombre de tabla |
| TAB_FILE_ID | BIGINT | table_file_id - Identificación de archivo de tabla |
| TAB_TYPE | BIGINT | table_type - Tipo de tabla |
| DATA_OBJECT_PAGES | BIGINT | data_object_pages - Páginas de objeto de datos |
| INDEX_OBJECT_PAGES | BIGINT | index_object_pages - Páginas de objeto de índice |
| LOB_OBJECT_PAGES | BIGINT | lob_object_pages - Páginas de objeto LOB |
| LONG_OBJECT_PAGES | BIGINT | long_object_pages - Páginas de objeto largo |
| ROWS_READ | BIGINT | rows_read - Filas leídas |
| ROWS_WRITTEN | BIGINT | rows_written - Filas grabadas |
| OVERFLOW_ACCESSES | BIGINT | overflow_accesses - Accesos a registros desbordados |
| PAGE_REORGS | BIGINT | page_reorgs - Reorganizaciones de página |
| DBPARTITIONNUM | SMALLINT | node_number - Número de nodo |

SNAP_GET_TBSP

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la "Vista administrativa SNAPTbsp y función de tabla SNAP_GET_TBSP_V91 – Recuperar información de instantáneas del grupo de datos lógicos de espacio de tablas" en la página 639

►►—SNAP_GET_TBSP—(—nombre-bd—,—númparticiónbd—)—◄◄

El esquema es SYSPROC.

La función de tabla SNAP_GET_TBSP devuelve información de instantáneas desde el grupo de datos lógicos de espacio de tablas.

nombre-bd

Argumento de entrada de tipo VARCHAR(255) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique el valor nulo para tomar la instantánea desde la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual. Si se especifica el valor nulo, se establece -1 de forma implícita.

Si ambos parámetros se establecen en NULL, la instantánea sólo se tomará si anteriormente no se ha creado un archivo mediante el procedimiento almacenado SNAPSHOT_FILEW para el tipo de petición de API de instantánea correspondiente.

La función devuelve una tabla tal como se muestra más abajo.

Tabla 263. Información devuelta por la función de tabla SNAP_GET_TBSP

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| TBSP_NAME | VARCHAR(128) | tablespace_name - Nombre de espacio de tablas |
| TBSP_ID | BIGINT | tablespace_id - Identificación de espacio de tablas |
| TBSP_TYPE | SMALLINT | tablespace_type - Tipo de espacio de tablas |
| TBSP_CONTENT_TYPE | SMALLINT | tablespace_content_type - Tipo de contenido de espacio de tablas |
| TBSP_PAGE_SIZE | BIGINT | tablespace_page_size - Tamaño de página de espacio de tablas |
| TBSP_EXTENT_SIZE | BIGINT | tablespace_extent_size - Tamaño de extensión de espacio de tablas |
| TBSP_PREFETCH_SIZE | BIGINT | tablespace_prefetch_size - Tamaño de captación previa de espacio de tablas |
| TBSP_CUR_POOL_ID | BIGINT | tablespace_cur_pool_id - Agrupación de almacenamiento intermedio utilizada actualmente |
| TBSP_NEXT_POOL_ID | BIGINT | tablespace_next_pool_id - Agrupación de almacenamiento intermedio que se utilizará en el siguiente arranque |
| FS_CACHING ¹ | SMALLINT | fs_caching - Antememoria del sistema de archivos |
| POOL_DATA_L_READS | BIGINT | pool_data_l_reads - Lecturas lógicas de datos de la agrupación de almacenamientos intermedios |
| POOL_DATA_P_READS | BIGINT | pool_data_p_reads - Lecturas físicas de datos de la agrupación de almacenamientos intermedios |
| POOL_TEMP_DATA_L_READS | BIGINT | pool_temp_data_l_reads - Lecturas lógicas de datos temporales de la agrupación de almacenamientos intermedios |

Tabla 263. Información devuelta por la función de tabla SNAP_GET_TBSP (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|--|
| POOL_TEMP_DATA_P_READS | BIGINT | pool_temp_data_p_reads - Lecturas físicas de datos temporales de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_READS | BIGINT | pool_async_data_reads - Lecturas de datos asíncronas de agrupación de almacenamientos intermedios |
| POOL_DATA_WRITES | BIGINT | pool_data_writes - Grabaciones de datos de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_WRITES | BIGINT | pool_async_data_writes - Grabaciones de datos asíncronas de agrupación de almacenamientos intermedios |
| POOL_INDEX_L_READS | BIGINT | pool_index_l_reads - Lecturas lógicas de índices de la agrupación de almacenamientos intermedios |
| POOL_INDEX_P_READS | BIGINT | pool_index_p_reads - Lecturas físicas de índices de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_L_READS | BIGINT | pool_temp_index_l_reads - Lecturas lógicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_TEMP_INDEX_P_READS | BIGINT | pool_temp_index_p_reads - Lecturas físicas de índices temporales de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_READS | BIGINT | pool_async_index_reads - Lecturas de índice asíncronas de agrupación de almacenamientos intermedios |
| POOL_INDEX_WRITES | BIGINT | pool_index_writes - Grabaciones de índices de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_WRITES | BIGINT | pool_async_index_writes - Grabaciones de índice asíncronas de agrupación de almacenamientos intermedios |
| POOL_READ_TIME | BIGINT | pool_read_time - Tiempo total de lectura física de la agrupación de almacenamientos intermedios |
| POOL_WRITE_TIME | BIGINT | pool_write_time - Tiempo total de grabación física de la agrupación de almacenamientos intermedios |
| POOL_ASYNC_READ_TIME | BIGINT | pool_async_read_time - Tiempo de lectura asíncrona de agrupación de almacenamientos intermedios |

Tabla 263. Información devuelta por la función de tabla SNAP_GET_TBSP (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|----------------------------|---------------|---|
| POOL_ASYNC_WRITE_TIME | BIGINT | pool_async_write_time - Tiempo de grabación asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_READ_REQS | BIGINT | pool_async_data_read_reqs - Peticiones de lectura asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_READ_REQS | BIGINT | pool_async_index_read_reqs - Peticiones de lectura de índice asíncrona de agrupación de almacenamientos intermedios |
| POOL_NO_VICTIM_BUFFER | BIGINT | pool_no_victim_buffer - Almacenamiento intermedio sin víctimas de la agrupación de almacenamientos intermedios |
| DIRECT_READS | BIGINT | direct_reads - Lecturas directas de la base de datos |
| DIRECT_WRITES | BIGINT | direct_writes - Grabaciones directas en la base de datos |
| DIRECT_READ_REQS | BIGINT | direct_read_reqs - Peticiones de lectura directa |
| DIRECT_WRITE_REQS | BIGINT | direct_write_reqs - Peticiones de grabación directa |
| DIRECT_READ_TIME | BIGINT | direct_read_time - Tiempo de lectura directa |
| DIRECT_WRITE_TIME | BIGINT | direct_write_time - Tiempo de grabación directa |
| FILES_CLOSED | BIGINT | files_closed - Archivos de base de datos cerrados |
| UNREAD_PREFETCH_PAGES | BIGINT | unread_prefetch_pages - Páginas de captación previa no leídas |
| POOL_DATA_TO_ESTORE | BIGINT | El elemento del supervisor ESTORE pool_data_to_estore se ha dejado de mantener. El sistema devolverá un valor NULL para el elemento del supervisor que se ha dejado de mantener. |
| POOL_INDEX_TO_ESTORE | BIGINT | El elemento del supervisor ESTORE pool_index_to_estore se ha dejado de mantener. El sistema devolverá un valor NULL para el elemento del supervisor que se ha dejado de mantener. |
| POOL_INDEX_FROM_ESTORE | BIGINT | El elemento del supervisor ESTORE pool_index_from_estore se ha dejado de mantener. El sistema devolverá un valor NULL para el elemento del supervisor que se ha dejado de mantener. |

Tabla 263. Información devuelta por la función de tabla SNAP_GET_TBSP (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--|---------------|--|
| POOL_DATA_FROM_ESTORE | BIGINT | El elemento del supervisor ESTORE pool_data_from_estore se ha dejado de mantener. El sistema devolverá un valor NULL para el elemento del supervisor que se ha dejado de mantener. |
| TBSP_REBALANCER_MODE | BIGINT | tablespace_rebalancer_mode - Modalidad de reequilibrador |
| TBSP_USING_AUTO_STORAGE | SMALLINT | tablespace_using_auto_storage - Espacio de tablas habilitado para el almacenamiento automático |
| TBSP_AUTO_RESIZE_ENABLED | SMALLINT | tablespace_auto_resize_enabled - Espacio de tablas habilitado para la modificación de tamaño automática |
| ¹ Si FS_CACHING es 0, el almacenamiento en antememoria del sistema de archivos está habilitado; si FS_CACHING es 1, el almacenamiento en antememoria del sistema de archivos está inhabilitado. | | |

SNAP_GET_TBSP_PART

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la "Vista administrativa SNAPTbsp_Part y función de tabla SNAP_GET_TBSP_PART_V91 – Recuperar información de instantáneas del grupo de datos lógicos tablespace_nodeinfo" en la página 797

►►—SNAP_GET_TBSP_PART—(—nombre-bd—,—númparticiónbd—)—◄◄

El esquema es SYSPROC.

La función de tabla SNAP_GET_TBSP_PART devuelve información de instantáneas del grupo de datos lógicos tablespace_nodeinfo.

nombre-bd

Argumento de entrada de tipo VARCHAR(255) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique el valor nulo para tomar la instantánea desde la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual. Si se especifica el valor nulo, se establece -1 de forma implícita.

Si ambos parámetros se establecen en NULL, la instantánea sólo se tomará si anteriormente no se ha creado un archivo mediante el procedimiento almacenado SNAPSHOT_FILEW para el tipo de petición de API de instantánea correspondiente.

La función devuelve una tabla tal como se muestra más abajo.

Tabla 264. Información devuelta por la función de tabla SNAP_GET_TBSP_PART

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| TBSP_NAME | VARCHAR (128) | tablespace_name - Nombre de espacio de tablas |
| TBSP_ID | BIGINT | tablespace_id - Identificación de espacio de tablas |
| TBSP_STATE | BIGINT | tablespace_state - Estado de espacio de tablas |
| TBSP_PREFETCH_SIZE | BIGINT | tablespace_prefetch_size - Tamaño de captación previa de espacio de tablas |
| TBSP_NUM QUIESCERS | BIGINT | tablespace_num_quiescers - Número de inmovilizadores |
| TBSP_STATE_CHANGE_OBJECT_ID | BIGINT | tablespace_state_change_object_id - Identificación de objeto de cambio de estado |
| TBSP_STATE_CHANGE_TBSP_ID | BIGINT | tablespace_state_change_ts_id - Identificación de espacio de tablas de cambio de estado |
| TBSP_MIN_RECOVERY_TIME | TIMESTAMP | tablespace_min_recovery_time - Tiempo mínimo de recuperación para la recuperación en avance |
| TBSP_TOTAL_PAGES | BIGINT | tablespace_total_pages - Total de páginas en espacio de tablas |
| TBSP_USABLE_PAGES | BIGINT | tablespace_usable_pages - Páginas utilizables de espacio de tablas |
| TBSP_USED_PAGES | BIGINT | tablespace_used_pages - Páginas utilizadas en espacio de tablas |
| TBSP_FREE_PAGES | BIGINT | tablespace_free_pages - Páginas libres en espacio de tablas |
| TBSP_PENDING_FREE_PAGES | BIGINT | tablespace_pending_free_pages - Páginas libres pendientes en espacio de tablas |
| TBSP_PAGE_TOP | BIGINT | tablespace_page_top - Nivel límite de espacio de tablas |
| REBALANCER_MODE | BIGINT | tablespace_rebalancer_mode - Modalidad de reequilibrador |
| REBALANCER_EXTENTS_REMAINING | BIGINT | tablespace_rebalancer_extents_remaining - Número total de extensiones que el reequilibrador debe procesar |
| REBALANCER_EXTENTS_PROCESSED | BIGINT | tablespace_rebalancer_extents_processed - Número de extensiones que el reequilibrador ha procesado |
| REBALANCER_PRIORITY | BIGINT | tablespace_rebalancer_priority - Prioridad de reequilibrador actual |

Tabla 264. Información devuelta por la función de tabla SNAP_GET_TBSP_PART (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------------|---------------|---|
| REBALANCER_START_TIME | TIMESTAMP | tablespace_rebalancer_start_time - Hora de inicio de reequilibrador |
| REBALANCER_RESTART_TIME | TIMESTAMP | tablespace_rebalancer_restart_time - Hora de reinicio de reequilibrador |
| REBALANCER_LAST_EXTENT_MOVED | BIGINT | tablespace_rebalancer_last_extent_moved - Última extensión movida por el reequilibrador |
| TBSP_NUM_RANGES | BIGINT | tablespace_num_ranges - Número de rangos de la correlación de espacio de tablas |
| TBSP_NUM_CONTAINERS | BIGINT | tablespace_num_containers - Número de contenedores de espacio de tablas |
| TBSP_INITIAL_SIZE | BIGINT | tablespace_initial_size - Tamaño inicial de espacio de tablas |
| TBSP_CURRENT_SIZE | BIGINT | tablespace_current_size - Tamaño actual de espacio de tablas |
| TBSP_MAX_SIZE | BIGINT | tablespace_max_size - Tamaño máximo de espacio de tablas |
| TBSP_INCREASE_SIZE | BIGINT | tablespace_increase_size - Aumento de tamaño en bytes |
| TBSP_INCREASE_SIZE_PERCENT | SMALLINT | tablespace_increase_size_percent - Aumento de tamaño en porcentaje |
| TBSP_LAST_RESIZE_TIME | TIMESTAMP | tablespace_last_resize_time - Hora del último redimensionamiento satisfactorio |
| TBSP_LAST_RESIZE_FAILED | SMALLINT | tablespace_last_resize_failed - Último intento de redimensionamiento fallido |
| DBPARTITIONNUM | SMALLINT | node_number - Número de nodo |

SNAPSHOT_AGENT

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la "Vista administrativa SNAPAGENT y función de tabla SNAP_GET_AGENT – Recuperar información de instantáneas de la aplicación del grupo de datos lógicos agent" en la página 515.

►►—SNAPSHOT_AGENT—(—nombrebd—,—númparticiónbd—)——►►

El esquema es SYSPROC.

La función SNAPSHOT_AGENT devuelve información sobre los agentes de una instantánea de aplicación.

nombredb

Argumento de entrada de tipo VARCHAR(255) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique el valor nulo para tomar la instantánea desde todas las bases de datos bajo la instancia de la base de datos.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

Si ambos parámetros se establecen en NULL, la instantánea sólo se tomará si el procedimiento almacenado SNAPSHOT_FILEW no ha creado anteriormente un archivo para el tipo de petición de API de instantánea correspondiente.

La función devuelve una tabla tal como se muestra más abajo.

Tabla 265. Información devuelta por la función de tabla SNAPSHOT_AGENT

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| AGENT_ID | BIGINT | agent_id - Descriptor de contexto de aplicación (ID de agente) |
| AGENT_PID | BIGINT | agent_pid - Unidad susceptible de envío de motor (EDU) |

SNAPSHOT_APPL

Devuelve información general de una instantánea de aplicación.

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la "Función de tabla SNAP_GET_APPL – Recuperar información de instantáneas del grupo de datos lógico appl" en la página 1051.

►►—SNAPSHOT_APPL—(—*nombredb*—, —*númparticiónbd*—)—◄◄

El esquema es SYSPROC.

nombredb

Argumento de entrada de tipo VARCHAR(255) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto

por el mandato LIST DATABASE DIRECTORY. Especifique el valor nulo para tomar la instantánea desde todas las bases de datos bajo la instancia de la base de datos.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

Si ambos parámetros se establecen en NULL, la instantánea sólo se tomará si el procedimiento almacenado SNAPSHOT_FILEW no ha creado anteriormente un archivo para el tipo de petición de API de instantánea correspondiente.

La función devuelve una tabla tal como se muestra más abajo.

Tabla 266. Información devuelta por la función de tabla SNAPSHOT_APPL

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| AGENT_ID | BIGINT | agent_id - Descriptor de contexto de aplicación (ID de agente) |
| UOW_LOG_SPACE_USED | BIGINT | uow_log_space_used - Unidad de espacio de registro de trabajo utilizado |
| ROWS_READ | BIGINT | rows_read - Filas leídas |
| ROWS_WRITTEN | BIGINT | rows_written - Filas grabadas |
| POOL_DATA_L_READS | BIGINT | pool_data_l_reads - Lecturas lógicas de datos de agrupación de almacenamientos intermedios |
| POOL_DATA_P_READS | BIGINT | pool_data_p_reads - Lecturas físicas de datos de agrupación de almacenamientos intermedios |
| POOL_DATA_WRITES | BIGINT | pool_data_writes - Grabaciones de datos de agrupación de almacenamientos intermedios |
| POOL_INDEX_L_READS | BIGINT | pool_index_l_reads - Lecturas lógicas de índice de agrupación de almacenamientos intermedios |
| POOL_INDEX_P_READS | BIGINT | pool_index_p_reads - Lecturas físicas de índice de agrupación de almacenamientos intermedios |
| POOL_INDEX_WRITES | BIGINT | pool_index_writes - Grabaciones de índice de agrupación de almacenamientos intermedios |
| POOL_READ_TIME | BIGINT | pool_read_time - Tiempo total de lectura física de agrupación de almacenamientos intermedios |

Tabla 266. Información devuelta por la función de tabla *SNAPSHOT_APPL* (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------|---------------|---|
| POOL_WRITE_TIME | BIGINT | pool_write_time - Tiempo total de grabación física de agrupación de almacenamientos intermedios |
| DIRECT_READS | BIGINT | direct_reads - Lecturas directas de la base de datos |
| DIRECT_WRITES | BIGINT | direct_writes - Grabaciones directas en la base de datos |
| DIRECT_READ_REQS | BIGINT | direct_read_reqs - Peticiones de lectura directa |
| DIRECT_WRITE_REQS | BIGINT | direct_write_reqs - Peticiones de grabación directa |
| DIRECT_READ_TIME | BIGINT | direct_read_time - Tiempo de lectura directa |
| DIRECT_WRITE_TIME | BIGINT | direct_write_time - Tiempo de grabación directa |
| POOL_DATA_TO_ESTORE | BIGINT | El elemento supervisor ESTORE pool_data_to_estore se ha dejado de mantener. Se devuelve un valor NULL para el elemento de supervisor que se ha dejado de mantener. |
| POOL_INDEX_TO_ESTORE | BIGINT | El elemento de supervisor ESTORE pool_index_to_estore se ha dejado de mantener. Se devuelve un valor NULL para el elemento de supervisor que se ha dejado de mantener. |
| POOL_INDEX_FROM_ESTORE | BIGINT | El elemento de supervisor ESTORE pool_index_from_estore se ha dejado de mantener. Se devuelve un valor NULL para el elemento de supervisor que se ha dejado de mantener. |
| POOL_DATA_FROM_ESTORE | BIGINT | El elemento de supervisor ESTORE pool_data_from_estore se ha dejado de mantener. Se devuelve un valor NULL para el elemento de supervisor que se ha dejado de mantener. |
| UNREAD_PREFETCH_PAGES | BIGINT | unread_prefetch_pages - Páginas de captación previa no leídas |
| LOCKS_HELD | BIGINT | locks_held - Bloqueos retenidos |
| LOCK_WAITS | BIGINT | lock_waits - Esperas de bloqueo |
| LOCK_WAIT_TIME | BIGINT | lock_wait_time - Tiempo esperado en bloqueos |
| LOCK_ESCALS | BIGINT | lock_escals - Número de escalamientos de bloqueo |
| X_LOCK_ESCALS | BIGINT | x_lock_escals - Escalas de bloqueo exclusivas |

Tabla 266. Información devuelta por la función de tabla *SNAPSHOT_APPL* (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------|---------------|---|
| DEADLOCKS | BIGINT | deadlocks - Puntos muertos detectados |
| TOTAL_SORTS | BIGINT | total_sorts - Clasificaciones totales |
| TOTAL_SORT_TIME | BIGINT | total_sort_time - Tiempo de clasificación total |
| SORT_OVERFLOWS | BIGINT | sort_overflows - Desbordamientos de clasificación |
| COMMIT_SQL_STMTS | BIGINT | commit_sql_stmts - Sentencias de confirmación intentadas |
| ROLLBACK_SQL_STMTS | BIGINT | rollback_sql_stmts - Sentencias de retrotracción intentadas |
| DYNAMIC_SQL_STMTS | BIGINT | dynamic_sql_stmts - Sentencias de SQL dinámico intentadas |
| STATIC_SQL_STMTS | BIGINT | static_sql_stmts - Sentencias de SQL estático intentadas |
| FAILED_SQL_STMTS | BIGINT | failed_sql_stmts - Operaciones de sentencia anómalas |
| SELECT_SQL_STMTS | BIGINT | select_sql_stmts - Sentencias de selección de SQL ejecutadas |
| DDL_SQL_STMTS | BIGINT | ddl_sql_stmts - Sentencias de lenguaje de definición de datos (DDL) de SQL |
| UID_SQL_STMTS | BIGINT | uid_sql_stmts - Sentencias UPDATE/INSERT/DELETE de SQL ejecutadas |
| INT_AUTO_REBINDS | BIGINT | int_auto_rebinds - Revinculaciones automáticas internas |
| INT_ROWS_DELETED | BIGINT | int_rows_deleted - Filas internas suprimidas |
| INT_ROWS_UPDATED | BIGINT | int_rows_updated - Filas internas actualizadas |
| INT_COMMITS | BIGINT | int_commits - Confirmaciones internas |
| INT_ROLLBACKS | BIGINT | int_rollback - Retrotracciones internas |
| INT_DEADLOCK_ROLLBACKS | BIGINT | int_deadlock_rollback - Retrotracciones internas debido a punto muerto |
| ROWS_DELETED | BIGINT | rows_deleted - Filas suprimidas |
| ROWS_INSERTED | BIGINT | rows_inserted - Filas insertadas |
| ROWS_UPDATED | BIGINT | rows_updated - Filas actualizadas |
| ROWS_SELECTED | BIGINT | rows_selected - Filas seleccionadas |
| BINDS_PRECOMPILES | BIGINT | binds_precompiles - Vinculaciones/precompilaciones intentadas |

Tabla 266. Información devuelta por la función de tabla *SNAPSHOT_APPL* (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------|---------------|---|
| OPEN_REM_CURS | BIGINT | open_rem_curs - Cursores remotos abiertos |
| OPEN_REM_CURS_BLK | BIGINT | open_rem_curs_blk - Cursores remotos abiertos con bloqueo |
| REJ_CURS_BLK | BIGINT | rej_curs_blk - Peticiones de cursor de bloqueo rechazadas |
| ACC_CURS_BLK | BIGINT | acc_curs_blk - Peticiones de cursor de bloqueo aceptadas |
| SQL_REQS_SINCE_COMMIT | BIGINT | sql_reqs_since_commit - Peticiones SQL desde la última petición |
| LOCK_TIMEOUTS | BIGINT | lock_timeouts - Número de tiempos de espera de bloqueo |
| INT_ROWS_INSERTED | BIGINT | int_rows_inserted - Filas internas insertadas |
| OPEN_LOC_CURS | BIGINT | open_loc_curs - Cursores locales abiertos |
| OPEN_LOC_CURS_BLK | BIGINT | open_loc_curs_blk - Cursores locales abiertos con bloqueo |
| PKG_CACHE_LOOKUPS | BIGINT | pkg_cache_lookups - Búsquedas de antememoria de paquete |
| PKG_CACHE_INSERTS | BIGINT | pkg_cache_inserts - Inserciones de antememoria de paquete |
| CAT_CACHE_LOOKUPS | BIGINT | cat_cache_lookups - Búsquedas de antememoria de catálogo |
| CAT_CACHE_INSERTS | BIGINT | cat_cache_inserts - Inserciones de antememoria de catálogo |
| CAT_CACHE_OVERFLOWS | BIGINT | cat_cache_overflows - Desbordamientos de antememoria de catálogo |
| CAT_CACHE_HEAP_FULL | BIGINT | cat_cache_overflows - Desbordamientos de antememoria de catálogo |
| NUM_AGENTS | BIGINT | num_agents - Número de agentes que trabajan en una sentencia |
| AGENTS_STOLEN | BIGINT | agents_stolen - Agentes robados |
| ASSOCIATED_AGENTS_TOP | BIGINT | associated_agents_top - Número máximo de agentes asociados |
| APPL_PRIORITY | BIGINT | appl_priority - Prioridad de agente de aplicación |
| APPL_PRIORITY_TYPE | BIGINT | appl_priority_type - Tipo de prioridad de aplicación |
| PREFETCH_WAIT_TIME | BIGINT | prefetch_wait_time - Tiempo esperado para captación previa |
| APPL_SECTION_LOOKUPS | BIGINT | appl_section_lookups - Búsqueda de sección |

Tabla 266. Información devuelta por la función de tabla *SNAPSHOT_APPL* (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------------|---------------|--|
| APPL_SECTION_INSERTS | BIGINT | appl_section_inserts - Inserciones de sección |
| LOCKS_WAITING | BIGINT | locks_waiting - Agentes esperando en bloqueos |
| TOTAL_HASH_JOINS | BIGINT | total_hash_joins - Total de uniones hash |
| TOTAL_HASH_LOOPS | BIGINT | total_hash_loops - Total de bucles hash |
| HASH_JOIN_OVERFLOWS | BIGINT | hash_join_overflows - Desbordamientos de uniones hash |
| HASH_JOIN_SMALL_OVERFLOWS | BIGINT | hash_join_small_overflows - Desbordamientos pequeños de uniones hash |
| APPL_IDLE_TIME | BIGINT | appl_idle_time - Tiempo de inactividad de aplicación |
| UOW_LOCK_WAIT_TIME | BIGINT | uow_lock_wait_time - Tiempo total que la unidad de trabajo ha esperado en bloqueos |
| UOW_COMP_STATUS | BIGINT | uow_comp_status - Estado de finalización de unidad de trabajo |
| AGENT_USR_CPU_TIME_S | BIGINT | agent_usr_cpu_time - Tiempo de CPU de usuario utilizado por el agente |
| AGENT_USR_CPU_TIME_MS | BIGINT | agent_usr_cpu_time - Tiempo de CPU de usuario utilizado por el agente |
| AGENT_SYS_CPU_TIME_S | BIGINT | agent_sys_cpu_time - Tiempo de CPU de sistema utilizado por el agente |
| AGENT_SYS_CPU_TIME_MS | BIGINT | agent_sys_cpu_time - Tiempo de CPU de sistema utilizado por el agente |
| APPL_CON_TIME | TIMESTAMP | appl_con_time - Indicación de fecha y hora de inicio de petición de conexión |
| CONN_COMPLETE_TIME | TIMESTAMP | conn_complete_time - Indicación de fecha y hora de finalización de petición de conexión |
| LAST_RESET | TIMESTAMP | last_reset - Indicación de fecha y hora de último restablecimiento |
| UOW_START_TIME | TIMESTAMP | uow_start_time - Indicación de fecha y hora de inicio de unidad de trabajo |
| UOW_STOP_TIME | TIMESTAMP | uow_stop_time - Indicación de fecha y hora de detención de unidad de trabajo |

Tabla 266. Información devuelta por la función de tabla *SNAPSHOT_APPL* (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|----------------------|---------------|--|
| PREV_UOW_STOP_TIME | TIMESTAMP | prev_uow_stop_time - Indicación de fecha y hora de finalización de unidad de trabajo anterior |
| UOW_ELAPSED_TIME_S | BIGINT | uow_elapsed_time - Tiempo transcurrido de unidad de trabajo más reciente |
| UOW_ELAPSED_TIME_MS | BIGINT | uow_elapsed_time - Tiempo transcurrido de unidad de trabajo más reciente |
| ELAPSED_EXEC_TIME_S | BIGINT | elapsed_exec_time - Tiempo transcurrido de ejecución de sentencia |
| ELAPSED_EXEC_TIME_MS | BIGINT | elapsed_exec_time - Tiempo transcurrido de ejecución de sentencia |
| INBOUND_COMM_ADDRESS | VARCHAR(32) | inbound_comm_address - Dirección de comunicaciones de entrada |

SNAPSHOT_APPL_INFO

Devuelve información general de una instantánea de aplicación.

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la "Función de tabla *SNAP_GET_APPL_INFO* - Recuperación de información de instantáneas del grupo de datos lógico *appl_info*" en la página 1059.

►►—SNAPSHOT_APPL_INFO—(—nombre-bd—,—númparticiónbd—)—————►►

El esquema es SYSPROC.

nombrebd

Argumento de entrada de tipo VARCHAR(255) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique el valor nulo para tomar la instantánea desde todas las bases de datos bajo la instancia de la base de datos.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

Si ambos parámetros se establecen en NULL, la instantánea sólo se tomará si el procedimiento almacenado SNAPSHOT_FILEW no ha creado anteriormente un archivo para el tipo de petición de API de instantánea correspondiente.

La función devuelve una tabla tal como se muestra más abajo.

Tabla 267. Información devuelta por la función de tabla SNAPSHOT_APPL_INFO

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| AGENT_ID | BIGINT | agent_id - Descriptor de contexto de aplicación (ID de agente) |
| APPL_STATUS | BIGINT | appl_status - Estado de aplicación |
| CODEPAGE_ID | BIGINT | codepage_id - ID de página de códigos utilizada por aplicación |
| NUM_ASSOC_AGENTS | BIGINT | num_assoc_agents - Número de agentes asociados |
| COORD_PARTITION_NUM | BIGINT | coord_node - Nodo de coordinación |
| AUTHORITY_LVL | BIGINT | authority_lvl - Nivel de autorización de usuario |
| CLIENT_PID | BIGINT | client_pid - ID de proceso de cliente |
| COORD_AGENT_PID | BIGINT | coord_agent_pid - Agente coordinador |
| STATUS_CHANGE_TIME | TIMESTAMP | status_change_time - Hora de cambio de estado de aplicación |
| CLIENT_PLATFORM | SMALLINT | client_platform - Plataforma operativa de cliente |
| CLIENT_PROTOCOL | SMALLINT | client_protocol - Protocolo de comunicaciones de cliente |
| COUNTRY_CODE | SMALLINT | territory_code - Código de territorio de base de datos |
| APPL_NAME | VARCHAR(256) | appl_name - Nombre de aplicación |
| APPL_ID | VARCHAR(128) | appl_id - ID de aplicación |
| SEQUENCE_NO | VARCHAR(4) | sequence_no - Número de secuencia |
| AUTH_ID | VARCHAR(128) | auth_id - ID de autorización |
| CLIENT_NNAME | VARCHAR(128) | El elemento de supervisor client_nname está obsoleto. El valor devuelto no es un valor válido. |
| CLIENT_PRDID | VARCHAR(128) | client_prdid - ID de producto/versión de cliente |

Tabla 267. Información devuelta por la función de tabla *SNAPSHOT_APPL_INFO* (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------|---------------|--|
| INPUT_DB_ALIAS | VARCHAR(128) | input_db_alias - Alias de base de datos de entrada |
| CLIENT_DB_ALIAS | VARCHAR(128) | client_db_alias - Alias de base de datos utilizado por la aplicación |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| DB_PATH | VARCHAR(1024) | db_path - Vía de acceso de base de datos |
| EXECUTION_ID | VARCHAR(128) | execution_id - ID de inicio de sesión de usuario |
| CORR_TOKEN | VARCHAR(128) | corr_token - Símbolo de correlación DRDA |
| TPMON_CLIENT_USERID | VARCHAR(256) | tpmon_client_userid - ID de usuario de cliente de supervisor de TP |
| TPMON_CLIENT_WKSTN | VARCHAR(256) | tpmon_client_wkstn - Nombre de estación de trabajo de cliente de supervisor de TP |
| TPMON_CLIENT_APP | VARCHAR(256) | tpmon_client_app - Nombre de aplicación de cliente de supervisor de TP |
| TPMON_ACC_STR | VARCHAR(200) | tpmon_acc_str - Serie de contabilidad de cliente de supervisor de TP |

SNAPSHOT_BP

Devuelve información de una instantánea de agrupación de almacenamientos intermedios.

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la "Función de tabla SNAP_GET_BP – Recuperación de la información de instantáneas del grupo de datos lógico bufferpool" en la página 1065.

►►—SNAPSHOT_BP—(—*nombrebd*—,—*númparticiónbd*—)—►►

El esquema es SYSPROC.

nombrebd

Argumento de entrada de tipo VARCHAR(255) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique el valor nulo para tomar la instantánea desde todas las bases de datos bajo la instancia de la base de datos.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

Si ambos parámetros se establecen en NULL, la instantánea sólo se tomará si el procedimiento almacenado SNAPSHOT_FILEW no ha creado anteriormente un archivo para el tipo de petición de API de instantánea correspondiente.

La función devuelve una tabla tal como se muestra más abajo.

Tabla 268. Información devuelta por la función de tabla SNAPSHOT_BP

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| POOL_DATA_L_READS | BIGINT | pool_data_l_reads - Lecturas lógicas de datos de agrupación de almacenamientos intermedios |
| POOL_DATA_P_READS | BIGINT | pool_data_p_reads - Lecturas físicas de datos de agrupación de almacenamientos intermedios |
| POOL_DATA_WRITES | BIGINT | pool_data_writes - Grabaciones de datos de agrupación de almacenamientos intermedios |
| POOL_INDEX_L_READS | BIGINT | pool_index_l_reads - Lecturas lógicas de índice de agrupación de almacenamientos intermedios |
| POOL_INDEX_P_READS | BIGINT | pool_index_p_reads - Lecturas físicas de índice de agrupación de almacenamientos intermedios |
| POOL_INDEX_WRITES | BIGINT | pool_index_writes - Grabaciones de índice de agrupación de almacenamientos intermedios |
| POOL_READ_TIME | BIGINT | pool_read_time - Tiempo total de lectura física de agrupación de almacenamientos intermedios |
| POOL_WRITE_TIME | BIGINT | pool_write_time - Tiempo total de grabación física de agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_READS | BIGINT | pool_async_data_reads - Lecturas de datos asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_WRITES | BIGINT | pool_async_data_writes - Grabaciones de datos asíncronas de agrupación de almacenamientos intermedios |

Tabla 268. Información devuelta por la función de tabla *SNAPSHOT_BP* (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------------|---------------|---|
| POOL_ASYNC_INDEX_WRITES | BIGINT | pool_async_index_writes - Grabaciones de índice asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_READ_TIME | BIGINT | pool_async_read_time - Tiempo de lectura asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_WRITE_TIME | BIGINT | pool_async_write_time - Tiempo de grabación asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_READ_REQS | BIGINT | pool_async_data_read_reqs - Peticiones de lectura asíncrona de agrupación de almacenamientos intermedios |
| DIRECT_READS | BIGINT | direct_reads - Lecturas directas de la base de datos |
| DIRECT_WRITES | BIGINT | direct_writes - Grabaciones directas en la base de datos |
| DIRECT_READ_REQS | BIGINT | direct_read_reqs - Peticiones de lectura directa |
| DIRECT_WRITE_REQS | BIGINT | direct_write_reqs - Peticiones de grabación directa |
| DIRECT_READ_TIME | BIGINT | direct_read_time - Tiempo de lectura directa |
| DIRECT_WRITE_TIME | BIGINT | direct_write_time - Tiempo de grabación directa |
| POOL_ASYNC_INDEX_READS | BIGINT | pool_async_index_reads - Lecturas de índice asíncronas de agrupación de almacenamientos intermedios |
| POOL_DATA_TO_ESTORE | BIGINT | El elemento supervisor ESTORE pool_data_to_estore se ha dejado de mantener. Se devuelve un valor NULL para el elemento de supervisor que se ha dejado de mantener. |
| POOL_INDEX_TO_ESTORE | BIGINT | El elemento de supervisor ESTORE pool_index_to_estore se ha dejado de mantener. Se devuelve un valor NULL para el elemento de supervisor que se ha dejado de mantener. |
| POOL_INDEX_FROM_ESTORE | BIGINT | El elemento de supervisor ESTORE pool_index_from_estore se ha dejado de mantener. Se devuelve un valor NULL para el elemento de supervisor que se ha dejado de mantener. |

Tabla 268. Información devuelta por la función de tabla *SNAPSHOT_BP* (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------|---------------|--|
| POOL_DATA_FROM_ESTORE | BIGINT | El elemento de supervisor ESTORE pool_data_from_estore se ha dejado de mantener. Se devuelve un valor NULL para el elemento de supervisor que se ha dejado de mantener. |
| UNREAD_PREFETCH_PAGES | BIGINT | unread_prefetch_pages - Páginas de captación previa no leídas |
| FILES_CLOSED | BIGINT | files_closed - Archivos de bases de datos cerrados |
| BP_NAME | VARCHAR(128) | bp_name - Nombre de agrupación de almacenamientos intermedios |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| DB_PATH | VARCHAR(1024) | db_path - Vía de acceso de base de datos |
| INPUT_DB_ALIAS | VARCHAR(128) | input_db_alias - Alias de base de datos de entrada |

SNAPSHOT_CONTAINER

Devuelve información de configuración de contenedor de una instantánea de espacio de tablas.

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la "Vista administrativa SNAPCONTAINER y función de tabla SNAP_GET_CONTAINER_V91 – Recuperar información acerca de la instantánea de grupo de datos lógicos tablespace_container" en la página 549

►►—SNAPSHOT_CONTAINER—(—*nombrebd*—,—*númparticiónbd*—)—◄◄

El esquema es SYSPROC.

nombrebd

Argumento de entrada de tipo VARCHAR(255) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique el valor nulo para tomar la instantánea desde la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

Si ambos parámetros se establecen en NULL, la instantánea sólo se tomará si el procedimiento almacenado SNAPSHOT_FILEW no ha creado anteriormente un archivo para el tipo de petición de API de instantánea correspondiente.

La función devuelve una tabla tal como se muestra más abajo.

Tabla 269. Información devuelta por la función de tabla SNAPSHOT_CONTAINER

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| TABLESPACE_ID | BIGINT | tablespace_id - Identificación de espacio de tablas |
| TABLESPACE_NAME | VARCHAR(128) | tablespace_name - Nombre de espacio de tablas |
| CONTAINER_ID | BIGINT | container_id - Identificación de contenedor |
| CONTAINER_NAME | VARCHAR(256) | container_name - Nombre de contenedor |
| CONTAINER_TYPE | SMALLINT | container_type - Tipo de contenedor |
| TOTAL_PAGES | BIGINT | container_total_pages - Total de páginas en contenedor |
| USABLE_PAGES | BIGINT | container_usable_pages - Páginas utilizables en contenedor |
| ACCESSIBLE | BIGINT | container_accessible - Accesibilidad de contenedor |
| STRIPE_SET | BIGINT | container_stripe_set - Conjunto de bandas |

SNAPSHOT_DATABASE

Devuelve información de una instantánea de base de datos.

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la "Función de tabla SNAP_GET_DB_V91 - Recuperar información de instantánea desde el grupo lógico dbase" en la página 1083

►►SNAPSHOT_DATABASE—(—nombrebd—,—númparticiónbd—)◄◄

El esquema es SYSPROC.

nombrebd

Argumento de entrada de tipo VARCHAR(255) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique el valor nulo para tomar la instantánea desde todas las bases de datos bajo la instancia de la base de datos.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

Si ambos parámetros se establecen en NULL, la instantánea sólo se tomará si el procedimiento almacenado SNAPSHOT_FILEW no ha creado anteriormente un archivo para el tipo de petición de API de instantánea correspondiente.

La función devuelve una tabla tal como se muestra más abajo.

Tabla 270. Información devuelta por la función de tabla SNAPSHOT_DATABASE

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| SEC_LOG_USED_TOP | BIGINT | sec_log_used_top - Espacio máximo utilizado de anotaciones cronológicas secundarias |
| TOT_LOG_USED_TOP | BIGINT | tot_log_used_top - Espacio máximo utilizado de anotaciones cronológicas totales |
| TOTAL_LOG_USED | BIGINT | total_log_used - Espacio utilizado de anotaciones cronológicas totales |
| TOTAL_LOG_AVAILABLE | BIGINT | total_log_available - Anotaciones cronológicas totales disponibles |
| ROWS_READ | BIGINT | rows_read - Filas leídas |
| POOL_DATA_L_READS | BIGINT | pool_data_l_reads - Lecturas lógicas de datos de agrupación de almacenamientos intermedios |
| POOL_DATA_P_READS | BIGINT | pool_data_p_reads - Lecturas físicas de datos de agrupación de almacenamientos intermedios |
| POOL_DATA_WRITES | BIGINT | pool_data_writes - Grabaciones de datos de agrupación de almacenamientos intermedios |
| POOL_INDEX_L_READS | BIGINT | pool_index_l_reads - Lecturas lógicas de índice de agrupación de almacenamientos intermedios |
| POOL_INDEX_P_READS | BIGINT | pool_index_p_reads - Lecturas físicas de índice de agrupación de almacenamientos intermedios |
| POOL_INDEX_WRITES | BIGINT | pool_index_writes - Grabaciones de índice de agrupación de almacenamientos intermedios |
| POOL_READ_TIME | BIGINT | pool_read_time - Tiempo total de lectura física de agrupación de almacenamientos intermedios |

Tabla 270. Información devuelta por la función de tabla *SNAPSHOT_DATABASE* (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|---|
| POOL_WRITE_TIME | BIGINT | pool_write_time - Tiempo total de grabación física de agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_READS | BIGINT | pool_async_index_reads - Lecturas de índice asíncronas de agrupación de almacenamientos intermedios |
| POOL_DATA_TO_ESTORE | BIGINT | El elemento supervisor ESTORE pool_data_to_estore se ha dejado de mantener. Se devuelve un valor NULL para el elemento de supervisor que se ha dejado de mantener. |
| POOL_INDEX_TO_ESTORE | BIGINT | El elemento de supervisor ESTORE pool_index_to_estore se ha dejado de mantener. Se devuelve un valor NULL para el elemento de supervisor que se ha dejado de mantener. |
| POOL_INDEX_FROM_ESTORE | BIGINT | El elemento de supervisor ESTORE pool_index_from_estore se ha dejado de mantener. Se devuelve un valor NULL para el elemento de supervisor que se ha dejado de mantener. |
| POOL_DATA_FROM_ESTORE | BIGINT | El elemento de supervisor ESTORE pool_data_from_estore se ha dejado de mantener. Se devuelve un valor NULL para el elemento de supervisor que se ha dejado de mantener. |
| POOL_ASYNC_DATA_READS | BIGINT | pool_async_data_reads - Lecturas de datos asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_WRITES | BIGINT | pool_async_data_writes - Grabaciones de datos asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_WRITES | BIGINT | pool_async_index_writes - Grabaciones de índice asíncronas de agrupación de almacenamientos intermedios |
| POOL_ASYNC_READ_TIME | BIGINT | pool_async_read_time - Tiempo de lectura asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_WRITE_TIME | BIGINT | pool_async_write_time - Tiempo de grabación asíncrona de agrupación de almacenamientos intermedios |

Tabla 270. Información devuelta por la función de tabla
 SNAPSHOT_DATABASE (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------------|---------------|--|
| POOL_ASYNC_DATA_READ_REQS | BIGINT | pool_async_data_read_reqs - Peticiones de lectura asíncrona de agrupación de almacenamientos intermedios |
| DIRECT_READS | BIGINT | direct_reads - Lecturas directas de la base de datos |
| DIRECT_WRITES | BIGINT | direct_writes - Grabaciones directas en la base de datos |
| DIRECT_READ_REQS | BIGINT | direct_read_reqs - Peticiones de lectura directa |
| DIRECT_WRITE_REQS | BIGINT | direct_write_reqs - Peticiones de grabación directa |
| DIRECT_READ_TIME | BIGINT | direct_read_time - Tiempo de lectura directa |
| DIRECT_WRITE_TIME | BIGINT | direct_write_time - Tiempo de grabación directa |
| UNREAD_PREFETCH_PAGES | BIGINT | unread_prefetch_pages - Páginas de captación previa no leídas |
| FILES_CLOSED | BIGINT | files_closed - Archivos de bases de datos cerrados |
| POOL_LSN_GAP_CLNS | BIGINT | pool_lsn_gap_clns - Limpiadores de espacio de anotaciones cronológicas de agrupación de almacenamientos intermedios activados |
| POOL_DRTY_PG_STEAL_CLNS | BIGINT | pool_drty_pg_steal_clns - Limpiadores de páginas víctima de agrupación de almacenamientos intermedios activados |
| POOL_DRTY_PG_THRSH_CLNS | BIGINT | pool_drty_pg_thrsh_clns - Limpiadores de umbral de agrupación de almacenamientos intermedios activados |
| LOCKS_HELD | BIGINT | locks_held - Bloqueos retenidos |
| LOCK_WAITS | BIGINT | lock_waits - Esperas de bloqueo |
| LOCK_WAIT_TIME | BIGINT | lock_wait_time - Tiempo esperado en bloqueos |
| LOCK_LIST_IN_USE | BIGINT | lock_list_in_use - Memoria total de lista de bloqueos en uso |
| DEADLOCKS | BIGINT | deadlocks - Puntos muertos detectados |
| LOCK_ESCALS | BIGINT | lock_escals - Número de escalamientos de bloqueo |
| X_LOCK_ESCALS | BIGINT | x_lock_escals - Escalas de bloqueo exclusivas |
| LOCKS_WAITING | BIGINT | locks_waiting - Agentes esperando en bloqueos |

Tabla 270. Información devuelta por la función de tabla `SNAPSHOT_DATABASE` (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------------------|---------------|---|
| <code>SORT_HEAP_ALLOCATED</code> | BIGINT | sort_heap_allocated - Total de pila de clasificación asignada |
| <code>TOTAL_SORTS</code> | BIGINT | total_sorts - Clasificaciones totales |
| <code>TOTAL_SORT_TIME</code> | BIGINT | total_sort_time - Tiempo de clasificación total |
| <code>SORT_OVERFLOWS</code> | BIGINT | sort_overflows - Desbordamientos de clasificación |
| <code>ACTIVE_SORTS</code> | BIGINT | active_sorts - Clasificaciones activas |
| <code>COMMIT_SQL_STMTS</code> | BIGINT | commit_sql_stmts - Sentencias de confirmación intentadas |
| <code>ROLLBACK_SQL_STMTS</code> | BIGINT | rollback_sql_stmts - Sentencias de retroacción intentadas |
| <code>DYNAMIC_SQL_STMTS</code> | BIGINT | dynamic_sql_stmts - Sentencias de SQL dinámico intentadas |
| <code>STATIC_SQL_STMTS</code> | BIGINT | static_sql_stmts - Sentencias de SQL estático intentadas |
| <code>FAILED_SQL_STMTS</code> | BIGINT | failed_sql_stmts - Operaciones de sentencia anómalas |
| <code>SELECT_SQL_STMTS</code> | BIGINT | select_sql_stmts - Sentencias de selección de SQL ejecutadas |
| <code>DDL_SQL_STMTS</code> | BIGINT | ddl_sql_stmts - Sentencias de lenguaje de definición de datos (DDL) de SQL |
| <code>UID_SQL_STMTS</code> | BIGINT | uid_sql_stmts - Sentencias UPDATE/INSERT/DELETE de SQL ejecutadas |
| <code>INT_AUTO_REBINDS</code> | BIGINT | int_auto_rebinds - Revinculaciones automáticas internas |
| <code>INT_ROWS_DELETED</code> | BIGINT | int_rows_deleted - Filas internas suprimidas |
| <code>INT_ROWS_UPDATED</code> | BIGINT | int_rows_updated - Filas internas actualizadas |
| <code>INT_COMMITS</code> | BIGINT | int_commits - Confirmaciones internas |
| <code>INT_ROLLBACKS</code> | BIGINT | int_rollback - Retroacciones internas |
| <code>INT_DEADLOCK_ROLLBACKS</code> | BIGINT | int_deadlock_rollback - Retroacciones internas debido a punto muerto |
| <code>ROWS_DELETED</code> | BIGINT | rows_deleted - Filas suprimidas |
| <code>ROWS_INSERTED</code> | BIGINT | rows_inserted - Filas insertadas |
| <code>ROWS_UPDATED</code> | BIGINT | rows_updated - Filas actualizadas |
| <code>ROWS_SELECTED</code> | BIGINT | rows_selected - Filas seleccionadas |

Tabla 270. Información devuelta por la función de tabla *SNAPSHOT_DATABASE* (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------|---------------|--|
| BINDS_PRECOMPILES | BIGINT | binds_precompiles - Vinculaciones/precompilaciones intentadas |
| TOTAL_CONS | BIGINT | total_cons - Conexiones desde la activación de base de datos |
| APPLS_CUR_CONS | BIGINT | appls_cur_cons - Aplicaciones conectadas actualmente |
| APPLS_IN_DB2 | BIGINT | appls_in_db2 - Aplicaciones que se ejecutan en la base de datos actualmente |
| SEC_LOGS_ALLOCATED | BIGINT | sec_logs_allocated - Anotaciones cronológicas secundarias asignadas actualmente |
| DB_STATUS | BIGINT | db_status - Estado de base de datos |
| LOCK_TIMEOUTS | BIGINT | lock_timeouts - Número de tiempos de espera de bloqueo |
| CONNECTIONS_TOP | BIGINT | connections_top - Número máximo de conexiones simultáneas |
| DB_HEAP_TOP | BIGINT | db_heap_top - Pila de base de datos máxima asignada |
| INT_ROWS_INSERTED | BIGINT | int_rows_inserted - Filas internas insertadas |
| LOG_READS | BIGINT | log_reads - Número de páginas de anotaciones cronológicas leídas |
| LOG_WRITES | BIGINT | log_writes - Número de páginas de anotaciones cronológicas grabadas |
| PKG_CACHE_LOOKUPS | BIGINT | pkg_cache_lookups - Búsquedas de antememoria de paquete |
| PKG_CACHE_INSERTS | BIGINT | pkg_cache_inserts - Inserciones de antememoria de paquete |
| CAT_CACHE_LOOKUPS | BIGINT | cat_cache_lookups - Búsquedas de antememoria de catálogo |
| CAT_CACHE_INSERTS | BIGINT | cat_cache_inserts - Inserciones de antememoria de catálogo |
| CAT_CACHE_OVERFLOWS | BIGINT | cat_cache_overflows - Desbordamientos de antememoria de catálogo |
| CAT_CACHE_HEAP_FULL | BIGINT | cat_cache_overflows - Desbordamientos de antememoria de catálogo |
| CATALOG_PARTITION | SMALLINT | catalog_node - Número de nodo de catálogo |
| TOTAL_SEC_CONS | BIGINT | total_sec_cons - Conexiones secundarias |

Tabla 270. Información devuelta por la función de tabla *SNAPSHOT_DATABASE* (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------------|---------------|---|
| NUM_ASSOC_AGENTS | BIGINT | num_assoc_agents - Número de agentes asociados |
| AGENTS_TOP | BIGINT | agents_top - Número de agentes creados |
| COORD_AGENTS_TOP | BIGINT | coord_agents_top - Número máximo de agentes de coordinación |
| PREFETCH_WAIT_TIME | BIGINT | prefetch_wait_time - Tiempo esperado para captación previa |
| APPL_SECTION_LOOKUPS | BIGINT | appl_section_lookups - Búsqueda de sección |
| APPL_SECTION_INSERTS | BIGINT | appl_section_inserts - Inserciones de sección |
| TOTAL_HASH_JOINS | BIGINT | total_hash_joins - Total de uniones hash |
| TOTAL_HASH_LOOPS | BIGINT | total_hash_loops - Total de bucles hash |
| HASH_JOIN_OVERFLOWS | BIGINT | hash_join_overflows - Desbordamientos de uniones hash |
| HASH_JOIN_SMALL_OVERFLOWS | BIGINT | hash_join_small_overflows - Desbordamientos pequeños de uniones hash |
| PKG_CACHE_NUM_OVERFLOWS | BIGINT | pkg_cache_num_overflows - Desbordamientos de antememoria de paquete |
| PKG_CACHE_SIZE_TOP | BIGINT | pkg_cache_size_top - Marca de límite superior de antememoria de paquete |
| DB_CONN_TIME | TIMESTAMP | db_conn_time - Indicación de fecha y hora de activación de base de datos |
| SQLM_ELM_LAST_RESET | TIMESTAMP | last_reset - Indicación de fecha y hora de último restablecimiento |
| SQLM_ELM_LAST_BACKUP | TIMESTAMP | last_backup - Indicación de fecha y hora de última copia de seguridad |
| APPL_CON_TIME | TIMESTAMP | appl_con_time - Indicación de fecha y hora de inicio de petición de conexión |
| DB_LOCATION | INTEGER | db_location - Ubicación de base de datos |
| SERVER_PLATFORM | INTEGER | server_platform - Sistema operativo de servidor |
| APPL_ID_OLDEST_XACT | BIGINT | appl_id_oldest_xact - Aplicación con la transacción más antigua |
| CATALOG_PARTITION_NAME | VARCHAR(128) | catalog_node_name - Nombre de red de nodo de catálogo |

Tabla 270. Información devuelta por la función de tabla *SNAPSHOT_DATABASE* (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|---|
| INPUT_DB_ALIAS | VARCHAR(128) | input_db_alias - Alias de base de datos de entrada |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| DB_PATH | VARCHAR(1024) | db_path - Vía de acceso de base de datos |

SNAPSHOT_DBM

Devuelve información de una instantánea del gestor de bases de datos DB2.

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la "Función de tabla *SNAP_GET_DBM* – Recuperación de la información de instantáneas del grupo de datos lógicos dbm" en la página 1079.

►—SNAPSHOT_DBM—(—númparticiónbd—)——►

El esquema es SYSPROC.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

Si se especifica el valor nulo, la instantánea sólo se tomará si anteriormente no se ha creado un archivo mediante el procedimiento almacenado *SNAPSHOT_FILEW* para el tipo de petición de API de instantánea correspondiente.

La función devuelve una tabla tal como se muestra más abajo.

Tabla 271. Información devuelta por la función de tabla *SNAPSHOT_DBM*

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| SORT_HEAP_ALLOCATED | BIGINT | sort_heap_allocated - Total de pila de clasificación asignada |
| POST_THRESHOLD_SORTS | BIGINT | post_threshold_sorts - Clasificaciones de umbral posteriores |
| PIPED_SORTS_REQUESTED | BIGINT | pipед_sorts_requested - Clasificaciones mediante conexiones solicitadas |

Tabla 271. Información devuelta por la función de tabla *SNAPSHOT_DBM* (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------------|---------------|---|
| PIPED_SORTS_ACCEPTED | BIGINT | piped_sorts_accepted - Clasificaciones mediante conexiones aceptadas |
| REM_CONS_IN | BIGINT | rem_cons_in - Conexiones remotas con gestor de bases de datos |
| REM_CONS_IN_EXEC | BIGINT | rem_cons_in_exec - Conexiones remotas que se ejecutan en el elemento de supervisor de gestor de bases de datos |
| LOCAL_CONS | BIGINT | local_cons - Conexiones locales |
| LOCAL_CONS_IN_EXEC | BIGINT | local_cons_in_exec - Conexiones locales que se ejecutan en el elemento de supervisor de gestor de bases de datos |
| CON_LOCAL_DBASES | BIGINT | con_local_dbases - Bases de datos locales con conexiones actuales |
| AGENTS_REGISTERED | BIGINT | agents_registered - Agentes registrados |
| AGENTS_WAITING_ON_TOKEN | BIGINT | agents_waiting_on_token - Agentes en espera de una señal |
| DB2_STATUS | BIGINT | db_status - Estado de base de datos |
| AGENTS_REGISTERED_TOP | BIGINT | agents_registered_top - Número máximo de agentes registrados |
| AGENTS_WAITING_TOP | BIGINT | agents_waiting_top - Número máximo de agentes en espera |
| COMM_PRIVATE_MEM | BIGINT | comm_private_mem - Memoria privada confirmada |
| IDLE_AGENTS | BIGINT | idle_agents - Número de agentes desocupados |
| AGENTS_FROM_POOL | BIGINT | agents_from_pool - Agentes asignados de la agrupación |
| AGENTS_CREATED_EMPTY_POOL | BIGINT | agents_created_empty_pool - Agentes creados porque la agrupación de agentes está vacía |
| COORD_AGENTS_TOP | BIGINT | coord_agents_top - Número máximo de agentes de coordinación |
| MAX_AGENT_OVERFLOWES | BIGINT | max_agent_overflows - Número máximo de desbordamientos de agentes |
| AGENTS_STOLEN | BIGINT | agents_stolen - Agentes robados |
| GW_TOTAL_CONS | BIGINT | gw_total_cons - Número total de conexiones intentadas para DB2 Connect |
| GW_CUR_CONS | BIGINT | gw_cur_cons - Número actual de conexiones para DB2 Connect |

Tabla 271. Información devuelta por la función de tabla *SNAPSHOT_DBM* (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------------|---------------|--|
| GW_CONS_WAIT_HOST | BIGINT | gw_cons_wait_host - Número de conexiones en espera de respuesta del sistema principal |
| GW_CONS_WAIT_CLIENT | BIGINT | gw_cons_wait_client - Número de conexiones en espera para que el cliente envíe petición |
| POST_THRESHOLD_HASH_JOINS | BIGINT | post_threshold_hash_joins - Umbral de uniones hash |
| INACTIVE_GW_AGENTS | BIGINT | idle_agents - Número de agentes desocupados |
| NUM_GW_CONN_SWITCHES | BIGINT | num_gw_conn_switches - conmutadores de conexión |
| DB2START_TIME | TIMESTAMP | db2start_time - Indicación de fecha y hora de inicio de gestor de base de datos |
| LAST_RESET | TIMESTAMP | last_reset - Indicación de fecha y hora de último restablecimiento |

SNAPSHOT_DYN_SQL

Devuelve información de una instantánea de SQL dinámico. Sustituye la función *SQLCACHE_SNAPSHOT*, que sigue estando disponible por razones de compatibilidad.

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la "Función de tabla *SNAP_GET_DYN_SQL_V91* - Recuperar información de la instantánea del grupo de datos lógicos *dynsql*" en la página 1095

►►—SNAPSHOT_DYN_SQL—(—*nombrebd*—,—*númparticiónbd*—)—◄◄

El esquema es *SYSPROC*.

nombrebd

Argumento de entrada de tipo *VARCHAR(255)* que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato *LIST DATABASE DIRECTORY*. Especifique el valor nulo para tomar la instantánea desde la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada de tipo *INTEGER* que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

Si ambos parámetros se establecen en NULL, la instantánea sólo se tomará si el procedimiento almacenado `SNAPSHOT_FILEW` no ha creado anteriormente un archivo para el tipo de petición de API de instantánea correspondiente.

La función devuelve una tabla tal como se muestra más abajo.

Tabla 272. Información devuelta por la función de tabla `SNAPSHOT_DYN_SQL`

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------------------|-------------------------------------|--|
| <code>SNAPSHOT_TIMESTAMP</code> | <code>TIMESTAMP</code> | La fecha y la hora en que se tomó la instantánea. |
| <code>ROWS_READ</code> | <code>BIGINT</code> | rows_read - Filas leídas |
| <code>ROWS_WRITTEN</code> | <code>BIGINT</code> | rows_written - Filas grabadas |
| <code>NUM_EXECUTIONS</code> | <code>BIGINT</code> | num_executions - Ejecuciones de sentencia |
| <code>NUM_COMPILATIONS</code> | <code>BIGINT</code> | num_compilations - Compilaciones de sentencia |
| <code>PREP_TIME_WORST</code> | <code>BIGINT</code> | prep_time_worst - Peor tiempo de preparación de sentencia |
| <code>PREP_TIME_BEST</code> | <code>BIGINT</code> | prep_time_best - Mejor tiempo de preparación de sentencia |
| <code>INT_ROWS_DELETED</code> | <code>BIGINT</code> | int_rows_deleted - Filas internas suprimidas |
| <code>INT_ROWS_INSERTED</code> | <code>BIGINT</code> | int_rows_inserted - Filas internas insertadas |
| <code>INT_ROWS_UPDATED</code> | <code>BIGINT</code> | int_rows_updated - Filas internas actualizadas |
| <code>STMT_SORTS</code> | <code>BIGINT</code> | stmt_sorts - Clasificaciones de sentencias |
| <code>TOTAL_EXEC_TIME</code> | <code>BIGINT</code> | total_exec_time - Tiempo de ejecución de sentencia transcurrido |
| <code>TOTAL_SYS_CPU_TIME</code> | <code>BIGINT</code> | total_sys_cpu_time - Total de CPU de sistema para una sentencia |
| <code>TOTAL_USR_CPU_TIME</code> | <code>BIGINT</code> | total_usr_cpu_time - Total de CPU de usuario para una sentencia |
| <code>STMT_TEXT</code> | <code>CLOB(16M)</code> ¹ | stmt_text - Texto de sentencia de SQL |

¹ `STMT_TEXT` está definido como `CLOB(16M)` para permitir sólo futuras ampliaciones. La salida real del texto de la sentencia se trunca en los 64 K.

SNAPSHOT_FCM

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la “Vista administrativa SNAPFCM y función de tabla SNAP_GET_FCM – Recuperar la información de instantáneas del grupo de datos lógicos fcm” en la página 589.

►►—SNAPSHOT_FCM—(—dbpartitionnum—)—————►►

El esquema es SYSPROC.

La función SNAPSHOT_FCM devuelve la información sobre el nivel del gestor de bases de datos respecto al gestor de comunicaciones rápidas (FCM).

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

La función devuelve una tabla tal como se muestra más abajo.

Tabla 273. Información devuelta por la función de tabla SNAPSHOT_FCM

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| BUFF_FREE | BIGINT | buff_free - Almacenamientos intermedios FCM libres actualmente |
| BUFF_FREE_BOTTOM | BIGINT | buff_free_bottom - Almacenamientos intermedios FCM mínimos libres |
| MA_FREE | BIGINT | El elemento de supervisor ma_free se ha dejado de mantener. El sistema devolverá un valor nulo para el elemento del supervisor que se ha dejado de mantener. |
| MA_FREE_BOTTOM | BIGINT | El elemento de supervisor ma_free_bottom se ha dejado de mantener. El sistema devolverá un valor nulo para el elemento del supervisor que se ha dejado de mantener. |

Tabla 273. Información devuelta por la función de tabla *SNAPSHOT_FCM* (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|--|
| CE_FREE | BIGINT | El elemento ce_free monitor se ha dejado de mantener. El sistema devolverá un valor nulo para el elemento del supervisor que se ha dejado de mantener. |
| CE_FREE_BOTTOM | BIGINT | El elemento de supervisor ce_free_bottom se ha dejado de mantener. El sistema devolverá un valor nulo para el elemento del supervisor que se ha dejado de mantener. |
| RB_FREE | BIGINT | El elemento rb_free monitor se ha dejado de mantener. El sistema devolverá un valor nulo para el elemento del supervisor que se ha dejado de mantener. |
| RB_FREE_BOTTOM | BIGINT | El elemento de supervisor rb_free_bottom se ha dejado de mantener. El sistema devolverá un valor nulo para el elemento del supervisor que se ha dejado de mantener. |
| PARTITION_NUMBER | SMALLINT | node_number - Número de nodo |

SNAPSHOT_FCMNODE

Devuelve información de una instantánea del gestor de comunicaciones rápidas del gestor de bases de datos.

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la "Vista administrativa *SNAPFCM_PART* y función de tabla *SNAP_GET_FCM_PART* – Recuperar la información de instantáneas del grupo de datos lógicos *fcm_node*" en la página 591.

►►—SNAPSHOT_FCMNODE—(—*númparticiónbd*—)——►►

El esquema es SYSPROC.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

Si se especifica el valor nulo, la instantánea sólo se tomará si anteriormente no se ha creado un archivo mediante el procedimiento almacenado `SNAPSHOT_FILEW` para el tipo de petición de API de instantánea correspondiente.

La función devuelve una tabla tal como se muestra más abajo.

Tabla 274. Información devuelta por la función de tabla `SNAPSHOT_FCMNODE`

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------------------|------------------------|--|
| <code>SNAPSHOT_TIMESTAMP</code> | <code>TIMESTAMP</code> | La fecha y la hora en que se tomó la instantánea. |
| <code>CONNECTION_STATUS</code> | <code>BIGINT</code> | connection_status - Estado de conexión |
| <code>TOTAL_BUFFERS_SENT</code> | <code>BIGINT</code> | total_buffers_sent - Total de almacenamientos intermedios FCM enviados |
| <code>TOTAL_BUFFERS_RCVD</code> | <code>BIGINT</code> | total_buffers_rcvd - Total de almacenamientos intermedios FCM recibidos |
| <code>PARTITION_NUMBER</code> | <code>SMALLINT</code> | node_number - Número de nodo |

SNAPSHOT_FILEW

Nota: Este procedimiento ha quedado obsoleto y se ha sustituido por el "Procedimiento `SNAP_WRITE_FILE`" en la página 666.

►► `SNAPSHOT_FILEW` *(—tipo-petición—, —nombre-bd—, —númparticiónbd—)* ◀◀

El esquema es `SYSPROC`.

El procedimiento `SNAPSHOT_FILEW` graba datos de instantánea del sistema en un archivo ubicado en el subdirectorio `tmp` del directorio de la instancia. Para ejecutar el procedimiento `SNAPSHOT_FILEW`, el usuario debe tener autorización `SYSADM`, `SYSCTRL` o `SYSMAINT`. Los usuarios que no tengan autorización `SYSADM`, `SYSCTRL` o `SYSMAINT` podrán leer la instantánea guardada pasando valores nulos como entrada a funciones de la instantánea.

tipo-petición

Argumento de entrada de tipo `SMALLINT` que especifica un tipo de petición de instantánea válido, tal como está definido en `sqlmon.h`.

nombrebd

Argumento de entrada de tipo `VARCHAR(128)` que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada en el momento en que se llama a este procedimiento. Especifique el valor nulo para tomar la instantánea desde la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada de tipo `SMALLINT` que especifica un número de partición de base de datos válido. Especifique `-1` para la partición de base de datos actual o `-2` para todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

Ejemplo: Tomar una instantánea de la información del gestor de bases de datos especificando un tipo de petición de 1 (que corresponde a SQLMA_DB2) y adoptando los valores por omisión de la base de datos conectada actualmente y la partición de base de datos actual.

```
CALL SNAPSHOT_FILEW (1, CAST (NULL AS VARCHAR(128)), CAST (NULL AS SMALLINT))
```

Esto dará como resultado que se graben datos de una instantánea en el directorio de instancias /tmp/SQLMA_DB2.dat del sistema operativo UNIX o en el directorio de instancias \tmp\SQLMA_DB2.dat de un sistema operativo Windows.

SNAPSHOT_LOCK

Devuelve información de una instantánea de bloqueo.

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la "Vista administrativa SNAPLOCK y función de tabla SNAP_GET_LOCK – Recuperar información de instantáneas del grupo de datos lógicos lock" en la página 599.

►►—SNAPSHOT_LOCK—(—*nombredb*—, —*númparticiónbd*—)—————►►

El esquema es SYSPROC.

nombredb

Argumento de entrada de tipo VARCHAR(255) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique el valor nulo para tomar la instantánea desde la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

Si ambos parámetros se establecen en NULL, la instantánea sólo se tomará si el procedimiento almacenado SNAPSHOT_FILEW no ha creado anteriormente un archivo para el tipo de petición de API de instantánea correspondiente.

La función devuelve una tabla tal como se muestra más abajo.

Tabla 275. Información devuelta por la función de tabla SNAPSHOT_LOCK

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| AGENT_ID | BIGINT | agent_id - Descriptor de contexto de aplicación (ID de agente) |

Tabla 275. Información devuelta por la función de tabla `SNAPSHOT_LOCK` (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|--|
| TABLE_FILE_ID | BIGINT | table_file_id - Identificación de archivo de tabla |
| LOCK_OBJECT_TYPE | BIGINT | lock_object_type - Tipo de objeto de bloqueo esperado |
| LOCK_MODE | BIGINT | lock_mode - Modalidad de bloqueo |
| LOCK_STATUS | BIGINT | lock_status - Estado de bloqueo |
| LOCK_OBJECT_NAME | BIGINT | lock_object_name - Nombre de objeto de bloqueo |
| PARTITION_NUMBER | SMALLINT | node_number - Número de nodo |
| LOCK_ESCALATION | SMALLINT | lock_escalation - Escalamiento de bloqueos |
| TABLE_NAME | VARCHAR(128) | table_name - Nombre de tabla |
| TABLE_SCHEMA | VARCHAR(128) | table_schema - Nombre de esquema de tabla |
| TABLESPACE_NAME | VARCHAR(128) | tablespace_name - Nombre de espacio de tablas |

SNAPSHOT_LOCKWAIT

Devuelve información de esperas de bloqueo de una instantánea de aplicación.

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la “Vista administrativa `SNAPLOCKWAIT` y función de tabla `SNAP_GET_LOCKWAIT` – Recuperar información de instantáneas del grupo de datos lógicos lockwait” en la página 604.

►► `SNAPSHOT_LOCKWAIT` (—*nombrebd*—, —*númparticiónbd*—) ◀◀

El esquema es `SYSPROC`.

nombrebd

Argumento de entrada de tipo `VARCHAR(255)` que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato `LIST DATABASE DIRECTORY`. Especifique el valor nulo para tomar la instantánea desde todas las bases de datos bajo la instancia de la base de datos.

númparticiónbd

Argumento de entrada de tipo `INTEGER` que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

Si ambos parámetros se establecen en NULL, la instantánea sólo se tomará si el procedimiento almacenado SNAPSHOT_FILEW no ha creado anteriormente un archivo para el tipo de petición de API de instantánea correspondiente.

La función devuelve una tabla tal como se muestra más abajo.

Tabla 276. Información devuelta por la función de tabla SNAPSHOT_LOCKWAIT

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|----------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| AGENT_ID | BIGINT | agent_id - Descriptor de contexto de aplicación (ID de agente) |
| SUBSECTION_NUMBER | BIGINT | ss_number - Número de subsección |
| LOCK_MODE | BIGINT | lock_mode - Modalidad de bloqueo |
| LOCK_OBJECT_TYPE | BIGINT | lock_object_type - Tipo de objeto de bloqueo esperado |
| AGENT_ID_HOLDING_LK | BIGINT | agent_id_holding_lock - ID de agente que mantiene el bloqueo |
| LOCK_WAIT_START_TIME | TIMESTAMP | lock_wait_start_time - Indicación de fecha y hora de inicio de espera de bloqueo |
| LOCK_MODE_REQUESTED | BIGINT | lock_mode_requested - Modalidad de bloqueo solicitada |
| PARTITION_NUMBER | SMALLINT | node_number - Número de nodo |
| LOCK_ESCALATION | SMALLINT | lock_escalation - Escalamiento de bloqueos |
| TABLE_NAME | VARCHAR(128) | table_name - Nombre de tabla |
| TABLE_SCHEMA | VARCHAR(128) | table_schema - Nombre de esquema de tabla |
| TABLESPACE_NAME | VARCHAR(128) | tablespace_name - Nombre de espacio de tablas |
| APPL_ID_HOLDING_LK | VARCHAR(128) | appl_id_holding_lk - ID de aplicación que mantiene el bloqueo |

SNAPSHOT QUIESCERS

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la "Vista administrativa SNAPTbsp_QUIESCER y función de tabla SNAP_GET_Tbsp_QUIESCER – Recuperar información acerca de la instantánea de espacio de tablas de inmovilizador" en la página 650.

El esquema es SYSPROC.

La función SNAPSHOT QUIESCERS devuelve información sobre inmovilizadores desde una instantánea de espacio de tablas.

nombrebd

Argumento de entrada de tipo VARCHAR(255) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique el valor nulo para tomar la instantánea desde la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

La función devuelve una tabla tal como se muestra más abajo.

Tabla 277. Información devuelta por la función de tabla SNAPSHOT QUIESCERS

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| TABLESPACE_NAME | VARCHAR(128) | tablespace_name - Nombre de espacio de tablas |
| QUIESCER_TBS_ID | BIGINT | quiescer_ts_id - Identificación de espacio de tablas de inmovilizador |
| QUIESCER_OBJ_ID | BIGINT | quiescer_obj_id - Identificación de objeto de inmovilizador |
| QUIESCER_AUTH_ID | BIGINT | quiescer_auth_id - Identificación de autorización de usuario de inmovilizador |
| QUIESCER_AGENT_ID | BIGINT | quiescer_agent_id - Identificación de agente de inmovilizador |
| QUIESCER_STATE | BIGINT | quiescer_state - Estado de inmovilizador |

SNAPSHOT_RANGES

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la “Vista administrativa SNAPTbsp_RANGE y función de tabla SNAP_GET_TBSP_RANGE – Recuperar información acerca de la instantánea de rango” en la página 654.

►►—SNAPSHOT_RANGES—(—nombre-bd—, —númparticiónbd—)—————►►

El esquema es SYSPROC.

La función SNAPSHOT_RANGES devuelve información de una instantánea de rango.

nombrebd

Argumento de entrada de tipo VARCHAR(255) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique el valor nulo para tomar la instantánea desde la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

La función devuelve una tabla tal como se muestra más abajo.

Tabla 278. Información devuelta por la función de tabla SNAPSHOT_RANGES

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| TABLESPACE_ID | BIGINT | tablespace_id - Identificación de espacio de tablas |
| TABLESPACE_NAME | VARCHAR(128) | tablespace_name - Nombre de espacio de tablas |
| RANGE_NUMBER | BIGINT | range_number - Número de rango |
| RANGE_STRIPE_SET_NUMBER | BIGINT | range_stripe_set_number - Número de conjunto de bandas |
| RANGE_OFFSET | BIGINT | range_offset - Desplazamiento de rango |
| RANGE_MAX_PAGE | BIGINT | range_max_page_number - Página máxima en rango |
| RANGE_MAX_EXTENT | BIGINT | range_max_extent - Extensión máxima en rango |
| RANGE_START_STRIPE | BIGINT | range_start_stripe - Banda inicial |
| RANGE_END_STRIPE | BIGINT | range_end_stripe - Banda final |

Tabla 278. Información devuelta por la función de tabla *SNAPSHOT_RANGES* (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------|---------------|---|
| RANGE_ADJUSTMENT | BIGINT | range_adjustment - Ajuste de rango |
| RANGE_NUM_CONTAINER | BIGINT | range_num_containers - Número de contenedores en rango |
| RANGE_CONTAINER_ID | BIGINT | range_container_id - Contenedor de rango |

SNAPSHOT_STATEMENT

Devuelve información sobre las sentencias de una instantánea de aplicación.

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la "Vista administrativa SNAPSTMT y función de tabla SNAP_GET_STMT – Recuperar información de instantáneas de sentencias" en la página 611.

►—SNAPSHOT_STATEMENT—(—*nombredb*—,—*númparticiónbd*—)—►

El esquema es SYSPROC.

nombredb

Argumento de entrada de tipo VARCHAR(255) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique el valor nulo para tomar la instantánea desde todas las bases de datos bajo la instancia de la base de datos.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

Si ambos parámetros se establecen en NULL, la instantánea sólo se tomará si el procedimiento almacenado SNAPSHOT_FILEW no ha creado anteriormente un archivo para el tipo de petición de API de instantánea correspondiente.

La función devuelve una tabla tal como se muestra más abajo.

Tabla 279. Información devuelta por la función de tabla *SNAPSHOT_STATEMENT*

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| AGENT_ID | BIGINT | agent_id - Descriptor de contexto de aplicación (ID de agente) |

Tabla 279. Información devuelta por la función de tabla *SNAPSHOT_STATEMENT* (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|----------------------|---------------|--|
| ROWS_READ | BIGINT | rows_read - Filas leídas |
| ROWS_WRITTEN | BIGINT | rows_written - Filas grabadas |
| NUM_AGENTS | BIGINT | num_agents - Número de agentes que trabajan en una sentencia |
| AGENTS_TOP | BIGINT | agents_top - Número de agentes creados |
| STMT_TYPE | BIGINT | stmt_type - Tipo de sentencia |
| STMT_OPERATION | BIGINT | stmt_operation/operation - Operación de sentencia |
| SECTION_NUMBER | BIGINT | section_number - Número de sección |
| QUERY_COST_ESTIMATE | BIGINT | query_cost_estimate - Estimación de coste de consulta |
| QUERY_CARD_ESTIMATE | BIGINT | query_card_estimate - Estimación de consulta de número de filas |
| DEGREE_PARALLELISM | BIGINT | degree_parallelism - Grado de paralelismo |
| STMT_SORTS | BIGINT | stmt_sorts - Clasificaciones de sentencias |
| TOTAL_SORT_TIME | BIGINT | total_sort_time - Tiempo de clasificación total |
| SORT_OVERFLOWS | BIGINT | sort_overflows - Desbordamientos de clasificación |
| INT_ROWS_DELETED | BIGINT | int_rows_deleted - Filas internas suprimidas |
| INT_ROWS_UPDATED | BIGINT | int_rows_updated - Filas internas actualizadas |
| INT_ROWS_INSERTED | BIGINT | int_rows_inserted - Filas internas insertadas |
| FETCH_COUNT | BIGINT | fetch_count - Número de captaciones satisfactorias |
| STMT_START | TIMESTAMP | stmt_start - Indicación de fecha y hora de inicio de operación de sentencia |
| STMT_STOP | TIMESTAMP | stmt_stop - Indicación de fecha y hora de detención de operación de sentencia |
| STMT_USR_CPU_TIME_S | BIGINT | stmt_usr_cpu_time - Tiempo de CPU de usuario utilizado por sentencia |
| STMT_USR_CPU_TIME_MS | BIGINT | stmt_usr_cpu_time - Tiempo de CPU de usuario utilizado por sentencia |
| STMT_SYS_CPU_TIME_S | BIGINT | stmt_sys_cpu_time - Tiempo de CPU de sistema utilizado por sentencia |

Tabla 279. Información devuelta por la función de tabla *SNAPSHOT_STATEMENT* (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------|------------------------|---|
| STMT_SYS_CPU_TIME_MS | BIGINT | stmt_sys_cpu_time - Tiempo de CPU de sistema utilizado por sentencia |
| STMT_ELAPSED_TIME_S | BIGINT | stmt_elapsed_time - Tiempo transcurrido de sentencia más reciente |
| STMT_ELAPSED_TIME_MS | BIGINT | stmt_elapsed_time - Tiempo transcurrido de sentencia más reciente |
| BLOCKING_CURSOR | SMALLINT | blocking_cursor - Cursor de bloqueo |
| STMT_PARTITION_NUMBER | SMALLINT | stmt_node_number - Nodo de sentencia |
| CURSOR_NAME | VARCHAR(128) | cursor_name - Nombre de cursor |
| CREATOR | VARCHAR(128) | creator - Creador de aplicación |
| PACKAGE_NAME | VARCHAR(128) | package_name - Nombre de paquete |
| STMT_TEXT | CLOB(16M) ¹ | stmt_text - Texto de sentencia de SQL |

¹ STMT_TEXT está definido como CLOB(16M) para permitir sólo futuras ampliaciones. La salida real del texto de la sentencia se trunca en los 64 K.

SNAPSHOT_SUBSECT

Devuelve información sobre las subsecciones de los planes de acceso de una instantánea de aplicación.

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la "Vista administrativa SNAPSUBSECTION y función de tabla SNAP_GET_SUBSECTION – Recuperar información de instantáneas del grupo de supervisor lógico de la subsección" en la página 621.

►►—SNAPSHOT_SUBSECT—(—*nombredb*—,—*númparticiónbd*—)—◄◄

El esquema es SYSPROC.

nombredb

Argumento de entrada de tipo VARCHAR(255) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique el valor nulo para tomar la instantánea desde todas las bases de datos bajo la instancia de la base de datos.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos

actual o -2 para todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

Si ambos parámetros se establecen en NULL, la instantánea sólo se tomará si el procedimiento almacenado SNAPSHOT_FILEW no ha creado anteriormente un archivo para el tipo de petición de API de instantánea correspondiente.

La función devuelve una tabla tal como se muestra más abajo.

Tabla 280. Información devuelta por la función de tabla SNAPSHOT_SUBSECT

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|------------------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| STMT_TEXT | CLOB(16M) ¹ | stmt_text - Texto de sentencia de SQL |
| SS_EXEC_TIME | BIGINT | ss_exec_time - Tiempo transcurrido de ejecución de subsección |
| TQ_TOT_SEND_SPILLS | BIGINT | tq_tot_send_spills - Número total de almacenamientos intermedios de cola de tabla desbordados |
| TQ_CUR_SEND_SPILLS | BIGINT | tq_cur_send_spills - Número actual de almacenamientos intermedios de cola de tabla desbordados |
| TQ_MAX_SEND_SPILLS | BIGINT | tq_max_send_spills - Número máximo de desbordamientos de almacenamientos intermedios de cola de tabla |
| TQ_ROWS_READ | BIGINT | tq_rows_read - Número de filas leídas de las colas de tabla |
| TQ_ROWS_WRITTEN | BIGINT | tq_rows_written - Número de filas grabadas en las colas de tabla |
| ROWS_READ | BIGINT | rows_read - Filas leídas |
| ROWS_WRITTEN | BIGINT | rows_written - Filas grabadas |
| SS_USR_CPU_TIME | BIGINT | ss_usr_cpu_time - Tiempo de CPU de usuario utilizado por subsección |
| SS_SYS_CPU_TIME | BIGINT | ss_sys_cpu_time - Tiempo de CPU de sistema utilizado por subsección |
| SS_NUMBER | INTEGER | ss_number - Número de subsección |
| SS_STATUS | INTEGER | ss_status - Estado de subsección |
| SS_PARTITION_NUMBER | SMALLINT | ss_node_number - Número de nodo de subsección |
| TQ_PARTITION_WAITED_FOR | SMALLINT | tq_node_waited_for - Nodo esperado en una cola de tabla |

Tabla 280. Información devuelta por la función de tabla *SNAPSHOT_SUBSECT* (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--|---------------|--|
| TQ_WAIT_FOR_ANY | INTEGER | tq_wait_for_any - En espera de cualquier nodo a enviar en una cola de tabla |
| TQ_ID_WAITING_ON | INTEGER | tq_id_waiting_on - Nodo en que se ha esperado en una cola de tabla |
| ¹ STMT_TEXT está definido como CLOB(16M) para permitir sólo futuras ampliaciones. La salida real del texto de la sentencia se trunca en los 64 K. | | |

SNAPSHOT_SWITCHES

Devuelve información sobre el estado de conmutador de instantánea de base de datos.

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la “Vista administrativa SNAPSHOT_SWITCHES y función de tabla SNAP_GET_SWITCHES - Recuperar información de estado de conmutador de instantáneas de base de datos” en la página 625.

►►—SNAPSHOT_SWITCHES—(—númparticiónbd—)—————►►

El esquema es SYSPROC.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

La función devuelve una tabla tal como se muestra más abajo.

Tabla 281. Información devuelta por la función de tabla *SNAPSHOT_SWITCHES*

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| UOW_SW_STATE | SMALLINT | Estado del conmutador de registro del supervisor de unidad de trabajo (0 ó 1). |
| UOW_SW_TIME | TIMESTAMP | Si el conmutador de registro del supervisor de unidad de trabajo está activado, la fecha y la hora a la que se activó este conmutador. |

Tabla 281. Información devuelta por la función de tabla *SNAPSHOT_SWITCHES* (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| STATEMENT_SW_STATE | SMALLINT | Estado del conmutador de registro del supervisor de sentencia de SQL (0 ó 1). |
| STATEMENT_SW_TIME | TIMESTAMP | Si el conmutador de registro del supervisor de sentencia de SQL está activado, la fecha y la hora a la que se activó este conmutador. |
| TABLE_SW_STATE | SMALLINT | Estado del conmutador de registro del supervisor de actividad de tabla (0 ó 1). |
| TABLE_SW_TIME | TIMESTAMP | Si el conmutador de registro del supervisor de actividad de tabla está activado, la fecha y la hora a la que se activó este conmutador. |
| BUFFPOOL_SW_STATE | SMALLINT | Estado del conmutador de registro del supervisor de actividad de agrupación de almacenamientos intermedios (0 ó 1). |
| BUFFPOOL_SW_TIME | TIMESTAMP | Si el conmutador de registro del supervisor de actividad de agrupación de almacenamientos intermedios está activado, la fecha y la hora a la que se activó este conmutador. |
| LOCK_SW_STATE | SMALLINT | Estado del conmutador de registro del supervisor de bloqueo (0 ó 1). |
| LOCK_SW_TIME | TIMESTAMP | Si el conmutador de registro del supervisor de bloqueo está activado, la fecha y la hora a la que se activó este conmutador. |
| SORT_SW_STATE | SMALLINT | Estado del conmutador de registro del supervisor de clasificación (0 ó 1). |
| SORT_SW_TIME | TIMESTAMP | Si el conmutador de registro del supervisor de clasificación está activado, la fecha y la hora a la que se activó este conmutador. |
| PARTITION_NUMBER | SMALLINT | node_number - Número de nodo |

SNAPSHOT_TABLE

Devuelve información de actividad de una instantánea de tabla.

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la "Vista administrativa SNAPTAB y función de tabla SNAP_GET_TAB_V91 – Recuperar información de instantáneas del grupo de datos lógicos table" en la página 628

►—SNAPSHOT_TABLE—(—*nombrebd*—,—*númparticiónbd*—)—————►

El esquema es SYSPROC.

nombrebd

Argumento de entrada de tipo VARCHAR(255) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique el valor nulo para tomar la instantánea desde la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

Si ambos parámetros se establecen en NULL, la instantánea sólo se tomará si el procedimiento almacenado SNAPSHOT_FILEW no ha creado anteriormente un archivo para el tipo de petición de API de instantánea correspondiente.

La función devuelve una tabla tal como se muestra más abajo.

Tabla 282. Información devuelta por la función de tabla SNAPSHOT_TABLE

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| ROWS_WRITTEN | BIGINT | rows_written - Filas grabadas |
| ROWS_READ | BIGINT | rows_read - Filas leídas |
| OVERFLOW_ACCESSES | BIGINT | overflow_accesses - Accesos a registros desbordados |
| TABLE_FILE_ID | BIGINT | table_file_id - Identificación de archivo de tabla |
| TABLE_TYPE | BIGINT | table_type - Tipo de tabla |
| PAGE_REORGS | BIGINT | page_reorgs - Reorganizaciones de página |
| TABLE_NAME | VARCHAR(128) | table_name - Nombre de tabla |
| TABLE_SCHEMA | VARCHAR(128) | table_schema - Nombre de esquema de tabla |

SNAPSHOT_TBREORG

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la “Vista administrativa SNAPTAB_REORG y función de tabla SNAP_GET_TAB_REORG - Recuperar información sobre instantáneas de reorganización de tabla” en la página 632.

►►—SNAPSHOT_TBREORG—(—*nombre-bd*—,—*númparticiónbd*—)—————►►

El esquema es SYSPROC.

La función SNAPSHOT_TBREORG devuelve información sobre la reorganización de tablas en forma de conjunto de resultados. Si no se ha reorganizado ninguna tabla, se devolverán 0 filas. Para obtener información de instantáneas en tiempo real, el usuario debe tener autorización SYSADM, SYSCTRL o SYSMANT.

nombrebd

Argumento de entrada de tipo VARCHAR(255) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique el valor nulo para tomar la instantánea desde la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

Si ambos parámetros se establecen en NULL, la instantánea sólo se tomará si el procedimiento almacenado SNAPSHOT_FILEW no ha creado anteriormente un archivo para el tipo de petición de API de instantánea correspondiente.

La función devuelve una tabla tal como se muestra más abajo.

Tabla 283. Información devuelta por la función de tabla SNAPSHOT_TBREORG

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|---|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| TABLE_NAME | VARCHAR(128) | table_name - Nombre de tabla |
| TABLE_SCHEMA | VARCHAR(128) | table_schema - Nombre de esquema de tabla |
| PAGE_REORGS | BIGINT | page_reorgs - Reorganizaciones de página |
| REORG_PHASE | BIGINT | reorg_phase - Fase de reorganización de tabla |
| REORG_MAX_PHASE | INTEGER | reorg_max_phase - Fase de reorganización de tabla máxima |

Tabla 283. Información devuelta por la función de tabla *SNAPSHOT_TBREORG* (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-----------------------|---------------|---|
| REORG_CURRENT_COUNTER | BIGINT | reorg_current_counter - Proceso de reorganización de tabla |
| REORG_MAX_COUNTER | BIGINT | reorg_max_counter - Cantidad total de reorganización de tabla |
| REORG_TYPE | INTEGER | reorg_type - Atributos de reorganización de tabla |
| REORG_STATUS | BIGINT | reorg_status - Estado de reorganización de tabla |
| REORG_COMPLETION | INTEGER | reorg_completion - Distintivo de finalización de reorganización de tabla |
| REORG_START | TIMESTAMP | reorg_start - Hora de inicio de reorganización de tabla |
| REORG_END | TIMESTAMP | reorg_end - Hora de finalización de reorganización de tabla |
| REORG_PHASE_START | TIMESTAMP | reorg_phase_start - Hora de inicio de fase de reorganización de tabla |
| REORG_INDEX_ID | BIGINT | reorg_index_id - Índice utilizado para reorganizar la tabla |
| REORG_TBSPC_ID | BIGINT | reorg_tbspc_id - Espacio de tablas donde se reorganiza la tabla |
| PARTITION_NUMBER | SMALLINT | node_number - Número de nodo |

SNAPSHOT_TBS

Devuelve información de actividad de una instantánea de espacio de tablas.

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la "Vista administrativa SNAPTbsp y función de tabla SNAP_GET_TBSP_V91 – Recuperar información de instantáneas del grupo de datos lógicos de espacio de tablas" en la página 639

►►—SNAPSHOT_TBS—(—*nombrebd*—,—*númparticiónbd*—)——►►

El esquema es SYSPROC.

nombrebd

Argumento de entrada de tipo VARCHAR(255) que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato LIST DATABASE DIRECTORY. Especifique el valor nulo para tomar la instantánea desde la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos activas. Una partición

de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

Si ambos parámetros se establecen en NULL, la instantánea sólo se tomará si el procedimiento almacenado SNAPSHOT_FILEW no ha creado anteriormente un archivo para el tipo de petición de API de instantánea correspondiente.

La función devuelve una tabla tal como se muestra más abajo.

Tabla 284. Información devuelta por la función de tabla SNAPSHOT_TBS

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| POOL_DATA_L_READS | BIGINT | pool_data_l_reads - Lecturas lógicas de datos de agrupación de almacenamientos intermedios |
| POOL_DATA_P_READS | BIGINT | pool_data_p_reads - Lecturas físicas de datos de agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_READS | BIGINT | pool_async_data_reads - Lecturas de datos asíncronas de agrupación de almacenamientos intermedios |
| POOL_DATA_WRITES | BIGINT | pool_data_writes - Grabaciones de datos de agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_WRITES | BIGINT | pool_async_data_writes - Grabaciones de datos asíncronas de agrupación de almacenamientos intermedios |
| POOL_INDEX_L_READS | BIGINT | pool_index_l_reads - Lecturas lógicas de índice de agrupación de almacenamientos intermedios |
| POOL_INDEX_P_READS | BIGINT | pool_index_p_reads - Lecturas físicas de índice de agrupación de almacenamientos intermedios |
| POOL_INDEX_WRITES | BIGINT | pool_index_writes - Grabaciones de índice de agrupación de almacenamientos intermedios |
| POOL_ASYNC_INDEX_WRITES | BIGINT | pool_async_index_writes - Grabaciones de índice asíncronas de agrupación de almacenamientos intermedios |
| POOL_READ_TIME | BIGINT | pool_read_time - Tiempo total de lectura física de agrupación de almacenamientos intermedios |
| POOL_WRITE_TIME | BIGINT | pool_write_time - Tiempo total de grabación física de agrupación de almacenamientos intermedios |
| POOL_ASYNC_READ_TIME | BIGINT | pool_async_read_time - Tiempo de lectura asíncrona de agrupación de almacenamientos intermedios |

Tabla 284. Información devuelta por la función de tabla *SNAPSHOT_TBS* (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------------|---------------|---|
| POOL_ASYNC_WRITE_TIME | BIGINT | pool_async_write_time - Tiempo de grabación asíncrona de agrupación de almacenamientos intermedios |
| POOL_ASYNC_DATA_READ_REQS | BIGINT | pool_async_data_read_reqs - Peticiones de lectura asíncrona de agrupación de almacenamientos intermedios |
| DIRECT_READS | BIGINT | direct_reads - Lecturas directas de la base de datos |
| DIRECT_WRITES | BIGINT | direct_writes - Grabaciones directas en la base de datos |
| DIRECT_READ_REQS | BIGINT | direct_read_reqs - Peticiones de lectura directa |
| DIRECT_WRITE_REQS | BIGINT | direct_write_reqs - Peticiones de grabación directa |
| DIRECT_READ_TIME | BIGINT | direct_read_time - Tiempo de lectura directa |
| DIRECT_WRITE_TIME | BIGINT | direct_write_time - Tiempo de grabación directa |
| UNREAD_PREFETCH_PAGES | BIGINT | unread_prefetch_pages - Páginas de captación previa no leídas |
| POOL_ASYNC_INDEX_READS | BIGINT | pool_async_index_reads - Lecturas de índice asíncronas de agrupación de almacenamientos intermedios |
| POOL_DATA_TO_ESTORE | BIGINT | El elemento supervisor ESTORE pool_data_to_estore se ha dejado de mantener. Se devuelve un valor NULL para el elemento de supervisor que se ha dejado de mantener. |
| POOL_INDEX_TO_ESTORE | BIGINT | El elemento de supervisor ESTORE pool_index_to_estore se ha dejado de mantener. Se devuelve un valor NULL para el elemento de supervisor que se ha dejado de mantener. |
| POOL_INDEX_FROM_ESTORE | BIGINT | El elemento de supervisor ESTORE pool_index_from_estore se ha dejado de mantener. Se devuelve un valor NULL para el elemento de supervisor que se ha dejado de mantener. |
| POOL_DATA_FROM_ESTORE | BIGINT | El elemento de supervisor ESTORE pool_data_from_estore se ha dejado de mantener. Se devuelve un valor NULL para el elemento de supervisor que se ha dejado de mantener. |

Tabla 284. Información devuelta por la función de tabla *SNAPSHOT_TBS* (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|---|
| FILES_CLOSED | BIGINT | files_closed - Archivos de bases de datos cerrados |
| TABLESPACE_NAME | VARCHAR(128) | tablespace_name - Nombre de espacio de tablas |

SNAPSHOT_TBS_CFG

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la "Vista administrativa *SNAPTbsp_Part* y función de tabla *SNAP_GET_TBSP_PART_V91* – Recuperar información de instantáneas del grupo de datos lógicos *tablespace_nodeinfo*" en la página 797

►► *SNAPSHOT_TBS_CFG* (—*nombredb*—, —*númparticiónbd*—) ◀◀

El esquema es *SYSPROC*.

La función *SNAPSHOT_TBS_CFG* devuelve información sobre la configuración de una instantánea de espacio de tablas.

nombredb

Argumento de entrada de tipo *VARCHAR(255)* que especifica un nombre de base de datos válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique un nombre de base de datos que tenga un tipo de entrada de directorio de "Indirect" o "Home", devuelto por el mandato *LIST DATABASE DIRECTORY*. Especifique el valor nulo para tomar la instantánea desde la base de datos conectada actualmente.

númparticiónbd

Argumento de entrada de tipo *INTEGER* que especifica un número válido de partición de base de datos. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos activas. Una partición de base de datos activa es una partición donde la base de datos está disponible para conexión y uso por parte de las aplicaciones.

Si se especifica el valor nulo, se establece -1 de forma implícita.

Si ambos parámetros se establecen en *NULL*, la instantánea sólo se tomará si el procedimiento almacenado *SNAPSHOT_FILEW* no ha creado anteriormente un archivo para el tipo de petición de API de instantánea correspondiente.

La función devuelve una tabla tal como se muestra más abajo.

Tabla 285. Información devuelta por la función de tabla *SNAPSHOT_TBS_CFG*

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|---------------|--|
| SNAPSHOT_TIMESTAMP | TIMESTAMP | La fecha y la hora en que se tomó la instantánea. |
| TABLESPACE_ID | BIGINT | tablespace_id - Identificación de espacio de tablas |

Tabla 285. Información devuelta por la función de tabla
 SNAPSHOT_TBS_CFG (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|---------------------|---------------|--|
| TABLESPACE_NAME | VARCHAR (128) | tablespace_name - Nombre de espacio de tablas |
| TABLESPACE_TYPE | SMALLINT | tablespace_type - Tipo de espacio de tablas |
| TABLESPACE_STATE | BIGINT | tablespace_state - Estado de espacio de tablas |
| NUM QUIESCERS | BIGINT | tablespace_num_quiescers - Número de inmovilizadores |
| STATE_CHANGE_OBJ_ID | BIGINT | tablespace_state_change_object_id - Identificación de objeto de cambio de estado |
| STATE_CHANGE_TBS_ID | BIGINT | tablespace_state_change_ts_id - Identificación de espacio de tablas de cambio de estado |
| MIN_RECOVERY_TIME | TIMESTAMP | tablespace_min_recovery_time - Tiempo mínimo de recuperación para la recuperación en avance |
| TBS_CONTENTS_TYPE | SMALLINT | tablespace_content_type - Tipo de contenido de espacio de tablas |
| BUFFERPOOL_ID | BIGINT | tablespace_cur_pool_id - Agrupación de almacenamiento intermedio en uso actualmente |
| NEXT_BUFFERPOOL_ID | BIGINT | tablespace_next_pool_id - Agrupación de almacenamiento intermedio que se utilizará en el siguiente arranque |
| PAGE_SIZE | BIGINT | tablespace_page_size - Tamaño de página de espacio de tablas |
| EXTENT_SIZE | BIGINT | tablespace_extent_size - Tamaño de extensión de espacio de tablas |
| PREFETCH_SIZE | BIGINT | tablespace_prefetch_size - Tamaño de captación previa de espacio de tablas |
| TOTAL_PAGES | BIGINT | tablespace_total_pages - Total de páginas en espacio de tablas |
| USABLE_PAGES | BIGINT | tablespace_usable_pages - Páginas utilizables de espacio de tablas |
| USED_PAGES | BIGINT | tablespace_used_pages - Páginas utilizadas en espacio de tablas |
| FREE_PAGES | BIGINT | tablespace_free_pages - Páginas libres en espacio de tablas |
| PENDING_FREE_PAGES | BIGINT | tablespace_pending_free_pages - Páginas libres pendientes en espacio de tablas |
| HIGH_WATER_MARK | BIGINT | pool_watermark - Marca de límite de agrupación de memoria |
| REBALANCER_MODE | BIGINT | tablespace_rebalancer_mode - Modalidad de reequilibrador |

Tabla 285. Información devuelta por la función de tabla *SNAPSHOT_TBS_CFG* (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|------------------------------|---------------|--|
| REBALANCER_EXTENTS_REMAINING | BIGINT | tablespace_rebalancer_extents_remaining - Número total de extensiones que el reequilibrador debe procesar |
| REBALANCER_EXTENTS_PROCESSED | BIGINT | tablespace_rebalancer_extents_processed - Número de extensiones que el reequilibrador ha procesado |
| REBALANCER_PRIORITY | BIGINT | tablespace_rebalancer_priority - Prioridad de reequilibrador actual |
| REBALANCER_START_TIME | TIMESTAMP | tablespace_rebalancer_start_time - Hora de inicio de reequilibrador |
| REBALANCER_RESTART_TIME | TIMESTAMP | tablespace_rebalancer_restart_time - Hora de inicio de reequilibrador |
| LAST_EXTENT_MOVED | BIGINT | tablespace_rebalancer_last_extent_moved - Última extensión movida por el reequilibrador |
| NUM_RANGES | BIGINT | tablespace_num_ranges - Número de rangos de la correlación de espacio de tablas |
| NUM_CONTAINERS | BIGINT | tablespace_num_containers - Número de contenedores de espacio de tablas |

SQLCACHE_SNAPSHOT

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la "Función de tabla *SNAP_GET_DYN_SQL_V91* - Recuperar información de la instantánea del grupo de datos lógicos *dynsql*" en la página 1095

►►—SQLCACHE_SNAPSHOT—(—)—————►

El esquema es SYSFUN.

La función *SQLCACHE_SNAPSHOT* devuelve los resultados de una instantánea de la antememoria de sentencias de SQL dinámico de DB2.

La función no toma ningún argumento. Devuelve una tabla, como se muestra a continuación.

Tabla 286. Información devuelta por la función de tabla *SQLCACHE_SNAPSHOT*

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|-------------------|---------------|---|
| NUM_EXECUTIONS | INTEGER | num_executions - Ejecuciones de sentencia |
| NUM_COMPILATIONS | INTEGER | num_compilations - Compilaciones de sentencia |

Tabla 286. Información devuelta por la función de tabla `SQLCACHE_SNAPSHOT` (continuación)

| Nombre de columna | Tipo de datos | Descripción o elemento del supervisor correspondiente |
|--------------------|------------------------|---|
| PREP_TIME_WORST | INTEGER | prep_time_worst - Peor tiempo de preparación de sentencia |
| PREP_TIME_BEST | INTEGER | prep_time_best - Mejor tiempo de preparación de sentencia |
| INT_ROWS_DELETED | INTEGER | int_rows_deleted - Filas internas suprimidas |
| INT_ROWS_INSERTED | INTEGER | int_rows_inserted - Filas internas insertadas |
| ROWS_READ | INTEGER | rows_read - Filas leídas |
| INT_ROWS_UPDATED | INTEGER | int_rows_updated - Filas internas actualizadas |
| ROWS_WRITE | INTEGER | rows_written - Filas grabadas |
| STMT_SORTS | INTEGER | stmt_sorts - Clasificaciones de sentencias |
| TOTAL_EXEC_TIME_S | INTEGER | total_exec_time - Tiempo transcurrido de ejecución de sentencia |
| TOTAL_EXEC_TIME_MS | INTEGER | total_exec_time - Tiempo transcurrido de ejecución de sentencia |
| TOT_U_CPU_TIME_S | INTEGER | total_usr_cpu_time - Total de CPU de usuario para una sentencia |
| TOT_U_CPU_TIME_MS | INTEGER | total_usr_cpu_time - Total de CPU de usuario para una sentencia |
| TOT_S_CPU_TIME_S | INTEGER | total_sys_cpu_time - Total de CPU de sistema para una sentencia |
| TOT_S_CPU_TIME_MS | INTEGER | total_sys_cpu_time - Total de CPU de sistema para una sentencia |
| DB_NAME | VARCHAR(128) | db_name - Nombre de base de datos |
| STMT_TEXT | CLOB(16M) ¹ | stmt_text - Texto de sentencia de SQL |

¹ STMT_TEXT está definido como CLOB(16M) para permitir sólo futuras ampliaciones. La salida real del texto de la sentencia se trunca en los 64 K.

RSYSINSTALLROUTINES

Nota: Este procedimiento ha quedado obsoleto. El procedimiento se utilizaba para crear nuevos procedimientos y funciones en DB2 UDB para Linux, UNIX y Windows Versión 8.

►►—SYSINSTALLROUTINES—(—)——————►►

El esquema es SYSPROC.

WLM_GET_ACTIVITY_DETAILS - Devolver información detallada sobre una actividad específica

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la “Función de tabla MON_GET_ACTIVITY_DETAILS - Obtener detalles completos sobre la actividad” en la página 370.

Esta función devuelve estadísticas básicas de una o varias subclases de servicio.

Esta función devuelve información detallada sobre una actividad específica identificada por su descriptor de contexto de aplicación, su ID de unidad de trabajo y su ID de actividad. Esta información incluye detalles sobre cualquier umbral que haya violado la actividad.

Sintaxis

►►—WLM_GET_ACTIVITY_DETAILS—(—*descriptorcontexto_aplicación*—,—*id_uow*—,—
►—*id_actividad*—,—*númparticiónbd*—)——————►►

El esquema es SYSPROC.

Parámetros de la función de tabla

descriptorcontexto_aplicación

Es un argumento de entrada de tipo BIGINT que especifica un descriptor de contexto de aplicación válido. Si el argumento es nulo, esta función no devuelve ninguna fila. Si el argumento es nulo, se devuelve un error SQL171N.

id_uow

Argumento de entrada de tipo INTEGER que especifica un identificador de unidad de trabajo válido exclusivo dentro de la aplicación. Si el argumento es nulo, esta función no devuelve ninguna fila. Si el argumento es nulo, se devuelve un error SQL171N.

id_actividad

Argumento de entrada de tipo INTEGER que especifica un ID de actividad válido exclusivo dentro de la unidad de trabajo. Si el argumento es nulo, esta función no devuelve ninguna fila. Si el argumento es nulo, se devuelve un error SQL171N.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número de partición válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique -1 para la partición de base

de datos actual o -2 para todas las particiones de base de datos. Si se especifica un valor nulo, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE sobre la función WLM_GET_ACTIVITY_DETAILS.

Ejemplo

Se puede obtener información detallada sobre una actividad individual mediante la función de tabla WLM_GET_ACTIVITY_DETAILS. Esta función de tabla devuelve información sobre la actividad como pares nombre-valor para cada partición. Este ejemplo se limita a mostrar sólo un subconjunto de once miembros de los pares nombre-valor para cada partición para una actividad identificada mediante el descriptor de contexto de aplicación 1, el ID de unidad de trabajo 1 y el ID de actividad 5. Para ver una lista completa de pares nombre-valor, consulte la Tabla 288 en la página 1158 y la Tabla 289 en la página 1161.

```
SELECT SUBSTR(CHAR(DBPARTITIONNUM),1,4) AS PART,
       SUBSTR(NAME, 1, 20) AS NAME,
       SUBSTR(VALUE, 1, 30) AS VALUE
FROM TABLE(WLM_GET_ACTIVITY_DETAILS(1, 1, 5, -2)) AS ACTDETAIL
WHERE NAME IN ('APPLICATION_HANDLE',
              'COORD_PARTITION_NUM',
              'LOCAL_START_TIME',
              'UOW_ID',
              'ACTIVITY_ID',
              'PARENT_UOW_ID',
              'PARENT_ACTIVITY_ID',
              'ACTIVITY_TYPE',
              'NESTING_LEVEL',
              'INVOCATION_ID',
              'ROUTINE_ID')
ORDER BY PART
```

El ejemplo siguiente muestra los datos de salida de la consulta.

| PART | NAME | VALUE |
|------|---------------------|----------------------------|
| 0 | APPLICATION_HANDLE | 1 |
| 0 | COORD_PARTITION_NUM | 0 |
| 0 | LOCAL_START_TIME | 2005-11-25-18.52.49.343000 |
| 0 | UOW_ID | 1 |
| 0 | ACTIVITY_ID | 5 |
| 0 | PARENT_UOW_ID | 1 |
| 0 | PARENT_ACTIVITY_ID | 3 |
| 0 | ACTIVITY_TYPE | READ_DML |
| 0 | NESTING_LEVEL | 0 |
| 0 | INVOCATION_ID | 1 |
| 0 | ROUTINE_ID | 0 |
| 1 | APPLICATION_HANDLE | 1 |
| 1 | COORD_PARTITION_NUM | 0 |
| 1 | LOCAL_START_TIME | 2005-11-25-18.52.49.598000 |
| 1 | UOW_ID | 1 |
| 1 | ACTIVITY_ID | 5 |
| 1 | PARENT_UOW_ID | |
| 1 | PARENT_ACTIVITY_ID | |
| 1 | ACTIVITY_TYPE | READ_DML |
| 1 | NESTING_LEVEL | 0 |
| 1 | INVOCATION_ID | 1 |
| 1 | ROUTINE_ID | 0 |

Nota sobre uso

El valor QUEUED de ACTIVITY_STATE significa que la actividad de coordinación ha realizado un RPC sobre la partición de catálogo para obtener tíquets de umbral y aún no se ha recibido una respuesta. Este estado puede indicar que WLM ha colocado la actividad en cola o, durante breves periodos de tiempo, puede indicar que la actividad está en proceso de obtener sus tíquets. Para obtener una imagen más precisa de si la actividad se está colocando realmente en cola o no, se puede determinar qué agente está trabajando en la actividad (mediante la función de tabla WLM_GET_SERVICE_CLASS_AGENTS) y buscar si el event_object del agente de la partición de catálogo tiene el valor WLM_QUEUE.

Información devuelta

Tabla 287. Información devuelta para WLM_GET_ACTIVITY_DETAILS

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| DBPARTITIONNUM | SMALLINT | Número de partición de la que se ha recopilado este registro. |
| NAME | VARCHAR(256) | Nombre del elemento. Consulte la Tabla 288 y Tabla 289 en la página 1161 para ver los valores posibles. |
| VALUE | VARCHAR(1024) | Valores de elemento. Consulte la Tabla 288 y Tabla 289 en la página 1161 para ver los valores posibles. |

Tabla 288. Elementos devueltos

| Nombre de elemento | Descripción |
|--------------------|--|
| ACTIVITY_ID | Identificador exclusivo de actividad dentro de una aplicación. |
| ACTIVITY_STATE | Los valores posibles incluyen: <ul style="list-style-type: none">• CANCEL_PENDING• EXECUTING• IDLE• INITIALIZING• QP_CANCEL_PENDING• QP_QUEUED• QUEUED• TERMINATING• DESCONOCIDO |
| ACTIVITY_TYPE | Los valores posibles incluyen: <ul style="list-style-type: none">• CALL• DDL• LOAD• OTHER• READ_DML• WRITE_DML |

Tabla 288. Elementos devueltos (continuación)

| Nombre de elemento | Descripción |
|-----------------------------|--|
| APPLICATION_HANDLE | ID exclusivo a nivel del sistema para la aplicación. En una base de datos de una sola partición, este identificador consiste en un recuento de 16 bits. En una base de datos de varias particiones, este identificador consiste en el número de partición de coordinación concatenado con un recuento de 16 bits. Además, este identificador será el mismo en cada partición en la que la aplicación puede realizar una conexión secundaria. |
| COORD_PARTITION_NUM | La partición de coordinación de la actividad. |
| DATABASE_WORK_ACTION_SET_ID | Si esta actividad se ha correlacionado con un conjunto de acciones de trabajo que se ha aplicado a la base de datos, esta columna contiene el ID del conjunto de acciones de trabajo. Esta columna contiene 0 si la actividad aún no se ha correlacionado con un conjunto de acciones de trabajo que se ha aplicado a la base de datos. |
| DATABASE_WORK_CLASS_ID | Si esta actividad se ha correlacionado con un conjunto de acciones de trabajo que se ha aplicado a la base de datos, esta columna contiene el ID de la clase de trabajo de esta actividad. Esta columna contiene 0 si la actividad aún no se ha correlacionado con un conjunto de acciones de trabajo que se ha aplicado a la base de datos. |
| EFFECTIVE_ISOLATION | Nivel de aislamiento efectivo correspondiente a esta actividad. |
| EFFECTIVE_LOCK_TIMEOUT | El valor de tiempo de espera excedido de bloqueo efectivo para esta actividad. |
| EFFECTIVE_QUERY_DEGREE | El valor efectivo del grado de consulta correspondiente a esta actividad. |
| ENTRY_TIME | La hora a la que esta actividad ha llegado al sistema. |
| INVOCATION_ID | Distingue una determinada invocación de esta actividad de otras del mismo nivel de anidamiento. Devuelve cero si la actividad no está anidada. |
| LAST_REFERENCE_TIME | Cada vez que se produce una petición en esta actividad, este campo se actualiza. |
| LOCAL_START_TIME | La hora a la que esta actividad ha empezado a funcionar en la partición. Es una hora local. Es campo puede ser una serie vacía si la actividad ha entrado en el sistema pero está en cola y aún no se ha empezado a ejecutar. |
| NESTING_LEVEL | Representa el nivel de anidamiento de esta actividad. El nivel de anidamiento es la profundidad a la que se anida esta actividad dentro de la actividad principal. |
| PACKAGE_NAME | Si la actividad es una sentencia SQL, representa el nombre de su paquete. |
| PACKAGE_SCHEMA | Si la actividad es una sentencia SQL, representa el nombre de esquema de su paquete. |
| PACKAGE_VERSION_ID | Si la actividad es una sentencia SQL, representa la versión de su paquete. |
| PARENT_ACTIVITY_ID | Identificador exclusiva de actividad dentro de una unidad de trabajo correspondiente a la actividad padre de la actividad cuyo ID es ACTIVITY_ID. Devuelve una serie vacía si la actividad no tiene actividad padre. |

Tabla 288. Elementos devueltos (continuación)

| Nombre de elemento | Descripción |
|----------------------------------|---|
| PARENT_UOW_ID | Identificador exclusivo de unidad de trabajo dentro de una aplicación. Se refiere a la unidad de trabajo original en la que se ha iniciado esta actividad padre. Devuelve una serie vacía si la actividad no tiene actividad padre o si está en una partición remota. |
| QP_QUERY_ID | ID de consulta que Query Patroller ha asignado a esta actividad si la actividad es una consulta. El ID de consulta 0 indica que Query Patroller no ha asignado ningún ID de consulta a esta actividad. |
| QUERY_COST_ESTIMATE | Coste estimado, en timerons, para una consulta, determinado por el compilador de SQL. |
| ROUTINE_ID | Identificador exclusivo de la rutina. Devuelve cero si la actividad no forma parte de una rutina. |
| ROWS_FETCHED | Número de filas leídas de la tabla. Sólo notifica los valores correspondientes a la partición de base para la que se graba este registro. En sistemas DPF, estos valores pueden no reflejar el total correcto de toda la actividad. Cuando el conmutador del supervisor de sentencias no está activado, este elemento no se recopila y se graba el valor -1. |
| ROWS_MODIFIED | Número de filas insertadas, actualizadas o suprimidas. Sólo notifica los valores correspondientes a la partición de base para la que se graba este registro. En sistemas DPF, estos valores pueden no reflejar el total correcto de toda la actividad. Cuando el conmutador del supervisor de sentencias no está activado, este elemento no se recopila y se graba el valor -1. |
| SECTION_NUMBER | Si la actividad es una sentencia SQL, representa su número de sección. |
| SERVICE_CLASS_ID | Identificador exclusivo de la clase de servicio a la que pertenece esta actividad. |
| SERVICE_CLASS_WORK_ACTION_SET_ID | Si esta actividad se ha correlacionado con un conjunto de acciones de trabajo que se ha aplicado a una clase de servicio, esta columna contiene el ID del conjunto de acciones de trabajo. Esta columna contiene 0 si la actividad aún no se ha correlacionado con un conjunto de acciones de trabajo que se ha aplicado a una clase de servicio. |
| SERVICE_CLASS_WORK_CLASS_ID | Si esta actividad se ha correlacionado con un conjunto de acciones de trabajo que se ha aplicado a una clase de servicio, esta columna contiene el ID de la clase de trabajo de esta actividad. Esta columna contiene 0 si la actividad aún no se ha correlacionado con un conjunto de acciones de trabajo que se ha aplicado a una clase de servicio. |
| STMT_PKG_CACHE_ID | Identificador de antememoria del paquete de la sentencia. |
| STMT_TEXT | Si la actividad es SQL dinámico o es SQL estático para el que el texto de la sentencia está disponible, este campo contiene los 1024 primeros caracteres del texto de la sentencia. De lo contrario es una serie vacía. |

Tabla 288. Elementos devueltos (continuación)

| Nombre de elemento | Descripción |
|--------------------|---|
| SYSTEM_CPU_TIME | Tiempo total de CPU del sistema (en segundos y microsegundos) utilizado por el proceso del agente del gestor de bases de datos, la unidad de trabajo o la sentencia. Cuando el conmutador del supervisor de sentencias o el conmutador de indicación de fecha y hora no están activados, este elemento no se recopila y se graba el valor -1. |
| UOW_ID | Identificador exclusivo de unidad de trabajo dentro de una aplicación. Hace referencia a la unidad de trabajo original en la que se ha iniciado esta actividad. |
| USER_CPU_TIME | Tiempo total de CPU del usuario (en segundos y microsegundos) utilizado por el proceso del agente del gestor de bases de datos, la unidad de trabajo o la sentencia. Cuando el conmutador del supervisor de sentencias o el conmutador de indicación de fecha y hora no están activados, este elemento no se recopila y se graba el valor -1. |
| UTILITY_ID | Si la actividad es un programa de utilidad, es el ID del programa de utilidad. De lo contrario, este campo es 0. |

Importante: La función de tabla WLM_GET_ACTIVITY_DETAILS muestra únicamente los umbrales que se están aplicando actualmente a una actividad.

Se devuelven los elementos siguientes únicamente si los umbrales correspondientes se aplican a la actividad.

Tabla 289. Elementos devueltos si se aplica

| Nombre de elemento | Descripción |
|---|--|
| ACTIVITYTOTALTIME_THRESHOLD_ID | El ID del umbral ACTIVITYTOTALTIME que se aplicó a la actividad. |
| ACTIVITYTOTALTIME_THRESHOLD_VALUE | Una indicación de fecha y hora que se calcula añadiendo la duración de umbral ACTIVITYTOTALTIME al tiempo de entrada de actividad. Si la actividad aún se está ejecutando cuando se alcanza esta indicación de fecha y hora, se violará el umbral. |
| ACTIVITYTOTALTIME_THRESHOLD_VIOLATED | 'Yes' indica que la actividad ha violado el umbral ACTIVITYTOTALTIME. 'No' indica que la actividad todavía no ha violado el umbral. |
| CONCURRENTDBCOORDACTIVITIES_DB_THRESHOLD_ID | El ID del umbral CONCURRENTDBCOORDACTIVITIES_DB que se aplicó a la actividad. |
| CONCURRENTDBCOORDACTIVITIES_DB_THRESHOLD_QUEUED | 'Yes' indica que el umbral CONCURRENTDBCOORDACTIVITIES_DB puso a la actividad en cola. 'No' indica que la actividad no se ha colocado en cola. |
| CONCURRENTDBCOORDACTIVITIES_DB_THRESHOLD_VALUE | El límite superior del umbral CONCURRENTDBCOORDACTIVITIES_DB que se aplicó a la actividad. |

Tabla 289. Elementos devueltos si se aplica (continuación)

| Nombre de elemento | Descripción |
|--|--|
| CONCURRENTDBCOORDACTIVITIES_DB_THRESHOLD_VIOLATED | 'Yes' indica que la actividad violó el umbral CONCURRENTDBCOORDACTIVITIES_DB. 'No' indica que la actividad todavía no ha violado el umbral. |
| CONCURRENTDBCOORDACTIVITIES_SUBCLASS_THRESHOLD_ID | El ID del umbral CONCURRENTDBCOORDACTIVITIES_SUBCLASS que se aplicó a la actividad. |
| CONCURRENTDBCOORDACTIVITIES_SUBCLASS_THRESHOLD_QUEUED | 'Yes' indica que el umbral CONCURRENTDBCOORDACTIVITIES_SUBCLASS puso a la actividad en cola. 'No' indica que la actividad no se ha colocado en cola. |
| CONCURRENTDBCOORDACTIVITIES_SUBCLASS_THRESHOLD_VALUE | El límite superior del umbral CONCURRENTDBCOORDACTIVITIES_SUBCLASS que se aplicó a la actividad. |
| CONCURRENTDBCOORDACTIVITIES_SUBCLASS_THRESHOLD_VIOLATED | 'Yes' indica que la actividad violó el umbral CONCURRENTDBCOORDACTIVITIES_SUBCLASS. 'No' indica que la actividad todavía no ha violado el umbral. |
| CONCURRENTDBCOORDACTIVITIES_SUPERCLASS_THRESHOLD_ID | El ID del umbral CONCURRENTDBCOORDACTIVITIES_SUPERCLASS que se aplicó a la actividad. |
| CONCURRENTDBCOORDACTIVITIES_SUPERCLASS_THRESHOLD_QUEUED | 'Yes' indica que el umbral CONCURRENTDBCOORDACTIVITIES_SUPERCLASS puso a la actividad en cola. 'No' indica que la actividad no se ha colocado en cola. |
| CONCURRENTDBCOORDACTIVITIES_SUPERCLASS_THRESHOLD_VALUE | El límite superior del umbral CONCURRENTDBCOORDACTIVITIES_SUPERCLASS que se aplicó a la actividad. |
| CONCURRENTDBCOORDACTIVITIES_SUPERCLASS_THRESHOLD_VIOLATED | 'Sí' indica que la actividad ha violado el umbral CONCURRENTDBCOORDACTIVITIES_SUPERCLASS. 'No' indica que la actividad todavía no ha violado el umbral. |
| CONCURRENTDBCOORDACTIVITIES_WORK_ACTION_SET_THRESHOLD_ID | El ID del umbral CONCURRENTDBCOORDACTIVITIES_WORK_ACTION_SET que se aplicó a la actividad. |
| CONCURRENTDBCOORDACTIVITIES_WORK_ACTION_SET_THRESHOLD_QUEUED | 'Yes' indica que el umbral CONCURRENTDBCOORDACTIVITIES_WORK_ACTION_SET puso a la actividad en cola. 'No' indica que la actividad no se ha colocado en cola. |
| CONCURRENTDBCOORDACTIVITIES_WORK_ACTION_SET_THRESHOLD_VALUE | El límite superior del umbral CONCURRENTDBCOORDACTIVITIES_WORK_ACTION_SET que se aplicó a la actividad. |
| CONCURRENTDBCOORDACTIVITIES_WORK_ACTION_SET_THRESHOLD_VIOLATED | 'Sí' indica que la actividad ha violado el umbral CONCURRENTDBCOORDACTIVITIES_WORK_ACTION_SET. 'No' indica que la actividad todavía no ha violado el umbral. |

Tabla 289. Elementos devueltos si se aplica (continuación)

| Nombre de elemento | Descripción |
|---|--|
| CONCURRENTWORKLOADACTIVITIES_THRESHOLD_ID | El ID del umbral CONCURRENTWORKLOADACTIVITIES que se aplicó a la actividad. |
| CONCURRENTWORKLOADACTIVITIES_THRESHOLD_VALUE | El límite superior del umbral CONCURRENTWORKLOADACTIVITIES que se aplicó a la actividad. |
| CONCURRENTWORKLOADACTIVITIES_THRESHOLD_VIOLATED | 'Yes' indica que la actividad ha violado el umbral CONCURRENTWORKLOADACTIVITIES. 'No' indica que la actividad todavía no ha violado el umbral. |
| ESTIMATEDSQLCOST_THRESHOLD_ID | El ID del umbral ESTIMATEDSQLCOST que se aplicó a la actividad. |
| ESTIMATEDSQLCOST_THRESHOLD_VALUE | El límite superior del umbral ESTIMATEDSQLCOST que se aplicó a la actividad. |
| ESTIMATEDSQLCOST_THRESHOLD_VIOLATED | '1' indica que la actividad ha violado el umbral ESTIMATEDSQLCOST. '0' indica que la actividad todavía no ha violado el umbral. |
| SQLROWSRETURNED_THRESHOLD_ID | El ID del umbral SQLROWSRETURNED que se aplicó a la actividad. |
| SQLROWSRETURNED_THRESHOLD_VALUE | El límite superior del umbral SQLROWSRETURNED que se aplicó a la actividad. |
| SQLROWSRETURNED_THRESHOLD_VIOLATED | 'Yes' indica que la actividad ha violado el umbral SQLROWSRETURNED. 'No' indica que la actividad todavía no ha violado el umbral. |
| SQLTEMPSPACE_THRESHOLD_ID | El ID del umbral SQLTEMPSPACE que se aplicó a la actividad. |
| SQLTEMPSPACE_THRESHOLD_VALUE | El límite superior del umbral SQLTEMPSPACE que se aplicó a la actividad. |
| SQLTEMPSPACE_THRESHOLD_VIOLATED | 'Yes' indica que la actividad ha violado el umbral SQLTEMPSPACE. 'No' indica que la actividad todavía no ha violado el umbral. |

WLM_GET_SERVICE_CLASS_AGENTS - Listar agentes en ejecución en una clase de servicio

La función WLM_GET_SERVICE_CLASS_AGENTS devuelve la lista de agentes, procesos en modalidad delimitada (procesos db2fmp) y entidades del sistema de la partición especificada que se están ejecutando en una clase de servicio dada o en nombre de una aplicación dada. Las entidades del sistema son procesos y hebras sin agente, como limpiadores de páginas y captadores previos.

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la "Función de tabla WLM_GET_SERVICE_CLASS_AGENTS_V97 - Listar los agentes que se ejecutan en una clase de servicio" en la página 883.

Sintaxis

►►—WLM_GET_SERVICE_CLASS_AGENTS—(—nombre_superclase_servicio—,—————►

El esquema es SYSPROC.

Parámetros de la función de tabla

nombre_superclase_servicio

Argumento de entrada de tipo VARCHAR(128) que especifica el nombre de una superclase de servicio de la base de datos actualmente conectada. Si el argumento es nulo o una serie vacía, se recuperan datos correspondientes a todas las superclases de la base de datos que coinciden con los valores de los otros parámetros.

nombre_subclase_servicio

Argumento de entrada de tipo VARCHAR(128) que hace referencia a una determinada subclase dentro de una superclase. Si el argumento es nulo o una serie vacía, se recuperan datos correspondientes a todas las subclases de la base de datos que coinciden con los valores de los otros parámetros.

descriptorcontexto_aplicación

Argumento de entrada de tipo BIGINT que especifica el descriptor de contexto de aplicación para el que se debe devolver información del agente. Si el argumento es nulo, se recuperan datos correspondientes a todas las aplicaciones de la base de datos que coinciden con los valores de los otros parámetros. Si se especifica 0, sólo se devuelven entidades del sistema.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número de partición válido en la misma instancia que la base de datos conectada actualmente. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos. Si se especifica un valor nulo, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE sobre la función WLM_GET_SERVICE_CLASS_AGENTS

Ejemplo

La consulta siguiente devuelve una lista de agentes asociados al descriptor de contexto de aplicación 1 para todas las particiones de la base de datos. El descriptor de contexto de aplicación se podría haber determinado mediante el mandato LIST APPLICATIONS o mediante la función de tabla WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES.

```
SELECT SUBSTR(CHAR(APPLICATION_HANDLE),1,7) AS APPHANDLE,
       SUBSTR(CHAR(DBPARTITIONNUM),1,4) AS PART,
       SUBSTR(CHAR(AGENT_TID),1,9) AS AGENT_TID,
       SUBSTR(AGENT_TYPE,1,11) AS AGENTTYPE,
       SUBSTR(AGENT_STATE,1,10) AS AGENTSTATE,
       SUBSTR(REQUEST_TYPE,1,12) AS REQTYPE,
       SUBSTR(CHAR(UOW_ID),1,6) AS UOW_ID,
       SUBSTR(CHAR(ACTIVITY_ID),1,6) AS ACT_ID
FROM TABLE(WLM_GET_SERVICE_CLASS_AGENTS(CAST(NULL AS VARCHAR(128)),
      CAST(NULL AS VARCHAR(128)), 1, -2)) AS SCDETAILS
ORDER BY APPHANDLE, PART, AGENT_TID
```

Un ejemplo de la salida es el siguiente:

| APPHANDLE | PART | AGENT_TID | AGENTTYPE | AGENTSTATE | REQTYPE | UOW_ID | ACT_ID |
|-----------|------|-----------|-------------|------------|--------------|--------|--------|
| 1 | 0 | 3 | COORDINATOR | ACTIVE | FETCH | 1 | 5 |
| 1 | 0 | 4 | SUBAGENT | ACTIVE | SUBSECTION:1 | 1 | 5 |
| 1 | 1 | 2 | SUBAGENT | ACTIVE | SUBSECTION:2 | 1 | 5 |

La salida muestra un agente de coordinación y un subagente en la partición 0, así como un subagente en la partición 1 que opera en nombre de una actividad con el ID de UOW 1 y el ID de actividad 5. El agente de coordinación indica que la petición es una petición de captación.

Nota sobre uso

Los parámetros tienen el efecto de sumarse (operación AND). Es decir, si se especifican parámetros de entrada conflictivos como una superclase de servicio SUP_A y una subclase de servicio SUB_B de modo que SUB_B no es una subclase de SUP_A, no se devuelve ninguna fila.

Información devuelta

Tabla 290. Información devuelta por WLM_GET_SERVICE_CLASS_AGENTS

| Nombre de columna | Tipo de datos | Descripción |
|-------------------------|---------------|---|
| SERVICE_SUPERCLASS_NAME | VARCHAR(128) | Nombre de la superclase de servicio de la que se ha recopilado este registro. |
| SERVICE_SUBCLASS_NAME | VARCHAR(128) | Nombre de la subclase de servicio de la que se ha recopilado este registro. |
| APPLICATION_HANDLE | BIGINT | ID exclusivo a nivel del sistema para la aplicación. En una base de datos de una sola partición, este identificador consiste en un recuento de 16 bits. En una base de datos de varias particiones, este identificador consiste en el número de partición de coordinación concatenado con un recuento de 16 bits. Además, este identificador es el mismo en cada partición en la que la aplicación realiza una conexión secundaria. |
| DBPARTITIONNUM | SMALLINT | Número de partición de la que se ha recopilado este registro. |
| ENTITY | VARCHAR(32) | Uno de los valores siguientes: <ul style="list-style-type: none"> • db2agent, que indica que el tipo de entidad es un agente • db2fmp (<i>pid</i>), que indica que la entidad es un proceso en modalidad delimitada, donde <i>pid</i> es el ID del proceso en modalidad delimitada • El nombre de una entidad del sistema |
| WORKLOAD_NAME | VARCHAR(128) | Nombre de la carga de trabajo de la que se ha recopilado este registro. |
| WORKLOAD_OCCURRENCE_ID | INTEGER | ID de la aparición de carga de trabajo. Este ID no identifica la aparición de carga de trabajo de forma exclusiva a no ser que esté combinado con el número de partición de base de datos de coordinación y con el nombre de la carga de trabajo. |
| UOW_ID | INTEGER | ID único de la unidad de trabajo en la que se ha iniciado esta actividad. |
| ACTIVITY_ID | INTEGER | ID de actividad exclusivo dentro de una unidad de trabajo. |
| PARENT_UOW_ID | INTEGER | ID único de la unidad de trabajo en la que se ha iniciado la actividad padre de la actividad. Si la actividad no tiene actividad padre, el valor de la columna es nulo. |

Tabla 290. Información devuelta por WLM_GET_SERVICE_CLASS_AGENTS (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|--------------------|---------------|--|
| PARENT_ACTIVITY_ID | INTEGER | ID exclusivo de actividad dentro de una unidad de trabajo correspondiente a la actividad padre de la actividad cuyo ID es activity_id. Si la actividad no tiene actividad padre, el valor de la columna es nulo. |
| AGENT_TID | BIGINT | ID de hebra del agente o de la entidad del sistema. Si este ID no está disponible, el valor de la columna es nulo. |
| AGENT_TYPE | VARCHAR(32) | Tipo de agente. Los valores posibles son los siguientes: <ul style="list-style-type: none"> • COORDINATOR • OTHER • PDBSUBAGENT • SMPSUBAGENT Si el tipo de agente es COORDINATOR, el ID de agente puede variar en entorno de concentrador. |
| SMP_COORDINATOR | INTEGER | Indica si el agente es de coordinación SMP. En caso afirmativo, el valor es 1; en caso contrario, el valor es 0. |
| AGENT_SUBTYPE | VARCHAR(32) | Subtipo de agente. Los valores posibles son los siguientes: <ul style="list-style-type: none"> • DSS • OTHER • RPC • SMP |
| AGENT_STATE | VARCHAR(32) | Estado de agente. Los valores posibles son los siguientes: <ul style="list-style-type: none"> • ACTIVE • ASSOCIATED |
| EVENT_TYPE | VARCHAR(32) | El tipo del último suceso que ha procesado este agente. Los valores posibles son los siguientes: <ul style="list-style-type: none"> • ACQUIRE • PROCESS • WAIT |
| EVENT_OBJECT | VARCHAR(32) | El objeto del último suceso que ha procesado este agente. Los valores posibles son los siguientes: <ul style="list-style-type: none"> • COMPRESSION_DICTIONARY_BUILD • IMPLICIT_REBIND • INDEX_RECREATE • LOCK • LOCK_ESCALATION • QP_QUEUE • REMOTE_REQUEST • REQUEST • ROUTINE • WLM_QUEUE |
| EVENT_STATE | VARCHAR(32) | El estado del último suceso que ha procesado este agente. Los valores posibles son los siguientes: <ul style="list-style-type: none"> • EXECUTING • IDLE |

Tabla 290. Información devuelta por WLM_GET_SERVICE_CLASS_AGENTS (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| REQUEST_ID | VARCHAR(64) | ID de petición. Este valor es único solamente si se combina con el valor de <i>descriptorcontexto_aplicación</i> . Puede utilizar esta combinación para distinguir entre una petición que tarda mucho tiempo y varias peticiones; por ejemplo, para distinguir varias captaciones de una captación larga. |
| REQUEST_TYPE | VARCHAR(32) | <p>Tipo de petición. Los valores posibles son los siguientes:</p> <ul style="list-style-type: none"> • Para agentes de coordinación: <ul style="list-style-type: none"> – CLOSE – COMMIT – COMPILE – DESCRIBE – EXCSQLSET – EXECIMMD – EXECUTE – FETCH – INTERNAL <i>número</i>, donde <i>número</i> es el valor de la constante interna – OPEN – PREPARE – REBIND – REDISTRIBUTE – REORG – ROLLBACK – RUNSTATS • Para subagentes con un AGENT_SUBTYPE de DSS o SMP <ul style="list-style-type: none"> – Si el número de subsección es distinto de cero, el número de subsección está en el formato <i>SUBSECCIÓN:número subsección</i>; de lo contrario, devuelve NULL. |

Tabla 290. Información devuelta por WLM_GET_SERVICE_CLASS_AGENTS (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-----------------------------|---------------|---|
| REQUEST_TYPE (continuación) | VARCHAR(32) | <ul style="list-style-type: none"> • Para subagentes con un AGENT_SUBTYPE de RPC <ul style="list-style-type: none"> - ABP - CATALOG - INTERNAL - REORG - RUNSTATS - WLM • Para subagentes con un AGENT_SUBTYPE de OTHER" <ul style="list-style-type: none"> - ABP - APP_RBSVPT - APP_RELSVPT - BACKUP - CLOSE - EXTERNAL_RBSVPT - EVMON - FORCE - FORCE_ALL - INTERNAL <i>número</i>, donde <i>número</i> es el valor de la constante interna - INTERRUPT - NOOP (si no hay petición) - QP - REDISTRIBUTE - STMT_RBSVPT - STOP_USING - UPDATE_DBM_CFG - WLM |
| NESTING_LEVEL | INTEGER | Nivel de anidamiento de la actividad cuyo ID es activity_id. El nivel de anidamiento es la profundidad a la que se anida esta actividad dentro de la actividad principal. |
| INVOCATION_ID | INTEGER | ID que distingue una determinada invocación de una actividad de otras del mismo nivel de anidamiento. |
| ROUTINE_ID | INTEGER | ID único de una rutina. El valor de la columna es nulo si la columna no forma parte de una rutina. |

WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES - Lista de apariciones de carga de trabajo

Devuelve la lista de todas las apariciones de carga de trabajo ejecutadas en una clase de servicio dada en una partición determinada.

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la "WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES_V97 - Lista de apariciones de carga de trabajo" en la página 890.

Una ocurrencia de carga de trabajo es una conexión de base de datos específica cuyos atributos coinciden con la definición de una carga de trabajo y por lo tanto está asociada a la carga de trabajo o asignada a la misma.

Sintaxis

```
►—WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES—(—nombre_superclase_servicio—, —►
►—nombre_subclase_servicio—, —númparticiónbd—) —►
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombre_superclase_servicio

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de superclase de servicio válido de la base de datos actualmente conectada. Si el argumento es nulo o una serie vacía, se recuperan datos correspondientes a todas las superclases de la base de datos para las que coinciden los otros parámetros.

nombre_subclase_servicio

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre desubclase de servicio válido en la base de datos actualmente conectada. Si el argumento es nulo o una serie vacía, se recuperan datos correspondientes a todas las subclases de la base de datos para las que coinciden los otros parámetros.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número de partición válido en la misma instancia que la base de datos conectada actualmente. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos. Si se especifica el valor nulo, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE sobre la función

WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES

Ejemplo

Si un administrador desea ver qué apariciones de carga de trabajo se están ejecutando en el sistema como un conjunto, puede llamar a la función WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES con un valor nulo o una serie vacía para *nombre_superclase_servicio* y *nombre_subclase_servicio* y -2 para *númparticiónbd*.

```
SELECT SUBSTR(SERVICE_SUPERCLASS_NAME,1,19) AS SUPERCLASS_NAME,
       SUBSTR(SERVICE_SUBCLASS_NAME,1,18) AS SUBCLASS_NAME,
       SUBSTR(CHAR(DBPARTITIONNUM),1,4) AS PART,
       SUBSTR(CHAR(COORD_PARTITION_NUM),1,4) AS COORDPART,
       SUBSTR(CHAR(APPLICATION_HANDLE),1,7) AS APPHNDL,
       SUBSTR(WORKLOAD_NAME,1,22) AS WORKLOAD_NAME,
       SUBSTR(CHAR(WORKLOAD_OCCURRENCE_ID),1,6) AS WLO_ID
FROM TABLE(WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES
            (CAST(NULL AS VARCHAR(128)), CAST(NULL AS VARCHAR(128)), -2))
AS SCINFO
ORDER BY SUPERCLASS_NAME, SUBCLASS_NAME, PART, APPHNDL,
       WORKLOAD_NAME, WLO_ID
```

Suponiendo que el sistema tiene cuatro particiones de base de datos y que está ejecutando dos cargas de trabajo en este momento, la consulta anterior generaría un resultado parecido al siguiente:

```

SUPERCLASS_NAME    SUBCLASS_NAME    PART COORDPART ...
-----
SYSDEFAULTMAINTENAN  SYSDEFAULTSUBCLASS  0    0    ...
SYSDEFAULTSYSTEMCLA  SYSDEFAULTSUBCLASS  0    0    ...
SYSDEFAULTUSERCLASS  SYSDEFAULTSUBCLASS  0    0    ...
SYSDEFAULTUSERCLASS  SYSDEFAULTSUBCLASS  0    0    ...
SYSDEFAULTUSERCLASS  SYSDEFAULTSUBCLASS  1    0    ...
SYSDEFAULTUSERCLASS  SYSDEFAULTSUBCLASS  1    0    ...
SYSDEFAULTUSERCLASS  SYSDEFAULTSUBCLASS  2    0    ...
SYSDEFAULTUSERCLASS  SYSDEFAULTSUBCLASS  2    0    ...
SYSDEFAULTUSERCLASS  SYSDEFAULTSUBCLASS  3    0    ...
SYSDEFAULTUSERCLASS  SYSDEFAULTSUBCLASS  3    0    ...

```

Datos de salida de esta consulta (continuación).

```

... APPHNDL WORKLOAD_NAME    WLO_ID
... -----
... -          -              -
... -          -              -
... 1      SYSDEFAULTUSERWORKLOAD 1
... 2      SYSDEFAULTUSERWORKLOAD 2
... 1      SYSDEFAULTUSERWORKLOAD 1
... 2      SYSDEFAULTUSERWORKLOAD 2
... 1      SYSDEFAULTUSERWORKLOAD 1
... 2      SYSDEFAULTUSERWORKLOAD 2
... 1      SYSDEFAULTUSERWORKLOAD 1
... 2      SYSDEFAULTUSERWORKLOAD 2

```

Nota sobre uso

Los parámetros tienen el efecto de sumarse (operación AND). Es decir, si se especifican parámetros de entrada conflictivos como una superclase de servicio SUP_A y una subclase de servicio SUB_B de modo que SUB_B no es una subclase de SUP_A, no se devuelve ninguna fila.

Nota: Las estadísticas notificadas para la aparición de carga de trabajo (por ejemplo coord_act_completed_total) se restablecen al principio de cada unidad de trabajo cuando se combinan con las estadísticas correspondientes de carga de trabajo.

Información devuelta

Tabla 291. Información devuelta para WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES

| Nombre de columna | Tipo de datos | Descripción |
|-------------------------|---------------|---|
| SERVICE_SUPERCLASS_NAME | VARCHAR(128) | Nombre de la superclase de servicio de la que se ha recopilado este registro. |
| SERVICE_SUBCLASS_NAME | VARCHAR(128) | Nombre de la subclase de servicio de la que se ha recopilado este registro. |
| DBPARTITIONNUM | SMALLINT | Número de la partición de la que se ha recopilado este registro. |
| COORD_PARTITION_NUM | SMALLINT | Número de partición de la partición de coordinación de la aparición de carga de trabajo dada. |

Tabla 291. Información devuelta para WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|---------------------------|---------------|--|
| APPLICATION_HANDLE | BIGINT | ID exclusivo a nivel del sistema para la aplicación. En una base de datos de una sola partición, este identificador consiste en un recuento de 16 bits. En una base de datos de varias particiones, este identificador consiste en el número de partición de coordinación concatenado con un recuento de 16 bits. Además, este identificador será el mismo en cada partición en la que la aplicación puede realizar una conexión secundaria. |
| WORKLOAD_NAME | VARCHAR(128) | Nombre de la carga de trabajo de la que se ha recopilado este registro. |
| WORKLOAD_OCCURRENCE_ID | INTEGER | ID de la aparición de carga de trabajo. No identifica la aparición de carga de trabajo de forma exclusiva a no ser que esté combinado con el número de partición de base de datos de coordinación y con el nombre de la carga de trabajo. |
| WORKLOAD_OCCURRENCE_STATE | VARCHAR(32) | Los valores posibles incluyen: <ul style="list-style-type: none"> • DECOUPLED - La aparición de la carga de trabajo no tiene ningún agente de coordinación asignado (caso de concentrador). • DISCONNECTPEND - La aparición de la carga de trabajo se está desconectando de la base de datos • FORCED - Se ha forzado la aparición de la carga de trabajo. • INTERRUPTED - Se ha interrumpido la aparición de la carga de trabajo. • QUEUED - Query Patroller o un umbral de colas de gestión de carga de trabajo ha puesto en cola el agente coordinador de apariciones de carga de trabajo. En un entorno de base de datos particionada, este estado puede indicar que el agente coordinador ha realizado un RPC en la partición de catálogo para obtener certificados de umbral y aún no ha recibido una respuesta. • TRANSIENT - La aparición de la carga de trabajo aún no se ha correlacionado con una superclase de servicio. • UOWEXEC - La aparición de la carga de trabajo está procesando una petición. • UOWWAIT - La aparición de la carga de trabajo está a la espera de una solicitud procedente del cliente. |
| UOW_ID | INTEGER | Identificador exclusivo de unidad de trabajo dentro de una aplicación. |
| SYSTEM_AUTH_ID | VARCHAR(128) | ID de autorización del sistema bajo el que la aparición de la carga de trabajo se ha inyectado en el sistema. |

Tabla 291. Información devuelta para WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|---------------------------|---------------|---|
| SESSION_AUTH_ID | VARCHAR(128) | ID de autorización de la sesión bajo el que la aparición de la carga de trabajo se ha inyectado en el sistema. |
| APPLICATION_NAME | VARCHAR(128) | El nombre de la aplicación que ha creado esta aparición de la carga de trabajo. |
| CLIENT_WRKSTNNAME | VARCHAR(255) | Valor actual del registro especial CLIENT_WRKSTNNAME correspondiente a esta aparición de la carga de trabajo. |
| CLIENT_ACCTNG | VARCHAR(255) | Valor actual del registro especial CLIENT_ACCTNG correspondiente a esta aparición de la carga de trabajo. |
| CLIENT_USER | VARCHAR(255) | El valor actual del registro especial CLIENT_USERID correspondiente a esta aparición de la carga de trabajo. |
| CLIENT_APPLNAME | VARCHAR(255) | Valor actual del registro especial CLIENT_APPLNAME correspondiente a esta aparición de la carga de trabajo. |
| COORD_ACT_COMPLETED_TOTAL | INTEGER | Número de actividades de coordinación a cualquier nivel de anidamiento completadas hasta el momento en la unidad de trabajo actual de esta aparición de la carga de trabajo. Esta estadística se actualiza cada vez que una actividad de esta aparición de la carga de trabajo se completa y se restablece al principio de cada unidad de trabajo. |
| COORD_ACT_ABORTED_TOTAL | INTEGER | Número de actividades de coordinación terminadas anormalmente hasta el momento en la unidad de trabajo actual de esta aparición de la carga de trabajo. Esta estadística se actualiza cada vez que una actividad de esta aparición de la carga de trabajo termina anormalmente y se restablece al principio de cada unidad de trabajo. |
| COORD_ACT_REJECTED_TOTAL | INTEGER | Número de actividades de coordinación rechazadas hasta el momento en la unidad de trabajo actual de esta aparición de la carga de trabajo. Una actividad se cuenta como rechazada cuando se impide su ejecución mediante una acción que impide el trabajo de ejecución o mediante un umbral de predicción. Esta estadística se actualiza cada vez que una actividad de esta aparición de la carga de trabajo se rechaza y se restablece al principio de cada unidad de trabajo. |
| CONCURRENT_ACT_TOP | INTEGER | Número más alto de actividades simultáneas en cualquier nivel de anidamiento que se están ejecutando (lo que incluye las desocupadas y las en espera) y en cola que se ha alcanzado para esta aparición de la carga de trabajo en la unidad de trabajo actual. Esta estadística se restablece al principio de cada unidad de trabajo. |

WLM_GET_SERVICE_SUBCLASS_STATS - Devolver estadísticas de subclases de servicio

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la "Función de tabla WLM_GET_SERVICE_SUBCLASS_STATS_V97 - Devolver estadísticas de subclases de servicio" en la página 895.

Esta función devuelve estadísticas básicas de una o varias subclases de servicio.

Sintaxis

```
►►—WLM_GET_SERVICE_SUBCLASS_STATS—(—nombre_superclase_servicio—,—————►  
►—nombre_subclase_servicio—, —númparticiónbd—)—————►◄
```

El esquema es SYSPROC.

Parámetros de la función de tabla

nombre_superclase_servicio

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de superclase de servicio válido que está en la misma base de datos a la que está conectada al llamar a esta función. Si el argumento es nulo o una serie vacía, se recuperan datos correspondientes a todas las superclases de la base de datos.

nombre_subclase_servicio

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de subclase de servicio válido que está en la misma base de datos a la que está conectada al llamar a esta función. Si el argumento es nulo o una serie vacía, se recuperan datos correspondientes a todas las subclases de la base de datos.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número de partición válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos. Si se especifica el valor nulo, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE sobre la función WLM_GET_SERVICE_SUBCLASS_STATS.

Ejemplos

Ejemplo 1: Puesto que cada actividad se tiene que correlacionar con una clase de servicio de DB2 antes de que se ejecute, regularmente se puede supervisar el estado global del sistema utilizando las funciones de tabla de estadísticas de clase de servicio y consultando todas las clases de servicio de todas las particiones. Tenga en cuenta que pasar un valor nulo para un argumento indica que el resultado no debe restringirse por ese argumento, excepto el argumento final, *dbpartitionnum*, donde un valor de -2 indica que se devuelven datos de todas las particiones de base de datos. Este ejemplo devuelve estadísticas de clase de servicio, como la duración media de la actividad y la desviación estándar en segundos:

```

SELECT SUBSTR(SERVICE_SUPERCLASS_NAME,1,19) AS SUPERCLASS_NAME,
       SUBSTR(SERVICE_SUBCLASS_NAME,1,18) AS SUBCLASS_NAME,
       SUBSTR(CHAR(DBPARTITIONNUM),1,4) AS PART,
       CAST(COORD_ACT_LIFETIME_AVG / 1000 AS DECIMAL(9,3))
       AS AVGLIFETIME,
       CAST(COORD_ACT_LIFETIME_STDDEV / 1000 AS DECIMAL(9,3))
       AS STDDEVLIFETIME,
       SUBSTR(CAST(LAST_RESET AS VARCHAR(30)),1,16) AS LAST_RESET
FROM TABLE(WLM_GET_SERVICE_SUBCLASS_STATS(CAST(NULL AS VARCHAR(128)),
      CAST(NULL AS VARCHAR(128)), -2)) AS SCSTATS
ORDER BY SUPERCLASS_NAME, SUBCLASS_NAME, PART

```

El ejemplo siguiente muestra los datos de salida de la consulta.

| SUPERCLASS_NAME | SUBCLASS_NAME | PART | ... |
|---------------------|--------------------|------|-----|
| SYSDEFAULTUSERCLASS | SYSDEFAULTSUBCLASS | 0 | ... |
| SYSDEFAULTUSERCLASS | SYSDEFAULTSUBCLASS | 1 | ... |
| SYSDEFAULTUSERCLASS | SYSDEFAULTSUBCLASS | 2 | ... |
| SYSDEFAULTUSERCLASS | SYSDEFAULTSUBCLASS | 3 | ... |

Datos de salida de esta consulta (continuación).

| ... | AVGLIFETIME | STDDEVLIFETIME | LAST_RESET |
|-----|-------------|----------------|------------------|
| ... | 691.242 | 34.322 | 2006-07-24-11.44 |
| ... | 644.740 | 22.124 | 2006-07-24-11.44 |
| ... | 612.431 | 43.347 | 2006-07-24-11.44 |
| ... | 593.451 | 28.329 | 2006-07-24-11.44 |

Ejemplo 2: La misma función de tabla también da el valor más alto correspondiente a simultaneidad promedio de actividades de coordinación que se ejecutan en la clase de servicio de cada partición.

```

SELECT SUBSTR(SERVICE_SUPERCLASS_NAME,1,19) AS SUPERCLASS_NAME,
       SUBSTR(SERVICE_SUBCLASS_NAME,1,18) AS SUBCLASS_NAME,
       SUBSTR(CHAR(DBPARTITIONNUM),1,4) AS PART,
       CONCURRENT_ACT_TOP AS ACTTOP,
       CONCURRENT_WLO_TOP AS CONNTOP
FROM TABLE(WLM_GET_SERVICE_SUBCLASS_STATS(CAST(NULL AS VARCHAR(128)),
      CAST(NULL AS VARCHAR(128)), -2)) AS SCSTATS
ORDER BY SUPERCLASS_NAME, SUBCLASS_NAME, PART

```

El ejemplo siguiente muestra los datos de salida de la consulta.

| SUPERCLASS_NAME | SUBCLASS_NAME | PART | ACTTOP | CONNTOP |
|---------------------|--------------------|------|--------|---------|
| SYSDEFAULTUSERCLASS | SYSDEFAULTSUBCLASS | 0 | 10 | 7 |
| SYSDEFAULTUSERCLASS | SYSDEFAULTSUBCLASS | 1 | 0 | 0 |
| SYSDEFAULTUSERCLASS | SYSDEFAULTSUBCLASS | 2 | 0 | 0 |
| SYSDEFAULTUSERCLASS | SYSDEFAULTSUBCLASS | 3 | 0 | 0 |

La salida de esta función de tabla le ofrece una buena vista a nivel alto de la "carga" de cada partición para una determinada base de datos. La salida proviene de la comprobación de los tiempos promedios de ejecución y los números de actividades. Cualquier variación significativa de los valores de nivel alto que devuelven estas funciones de tabla puede indicar un cambio en la carga del sistema.

Notas de uso

Algunas estadísticas sólo se devolverán si los valores COLLECT AGGREGATE ACTIVITY DATA y COLLECT AGGREGATE REQUEST DATA de la subclase de servicio correspondiente están establecidos en un valor que no sea "NONE".

La función de tabla WLM_GET_SERVICE_SUBCLASS_STATS devuelve una fila de datos por subclase de servicio y por partición. No se realiza ninguna agregación entre clases de servicio (de una partición) ni entre particiones (correspondientes a una o varias clases de servicio). Sin embargo, se puede lograr la agregación mediante consultas SQL, tal como se muestra en los ejemplos anteriores.

Los parámetros tienen el efecto de sumarse lógicamente como cláusulas "AND". Es decir, si se especifican parámetros de entrada conflictivos como un nombre de superclase llamado SUPA y una subclase llamada SUBB de modo que SUBB no es una subclase de SUPA, no se devolverá ninguna fila.

Información devuelta

Tabla 292. Información devuelta para WLM_GET_SERVICE_SUBCLASS_STATS

| Nombre de columna | Tipo de datos | Descripción |
|---------------------------|---------------|---|
| SERVICE_SUPERCLASS_NAME | VARCHAR(128) | Nombre de la superclase de servicio de la que se ha recopilado este registro. |
| SERVICE_SUBCLASS_NAME | VARCHAR(128) | Nombre de la subclase de servicio de la que se ha recopilado este registro. |
| DBPARTITIONNUM | SMALLINT | Número de partición de la que se ha recopilado este registro. |
| LAST_RESET | TIMESTAMP | <p>Hora en que se restauraron las estadísticas por última vez. Se pueden producir cuatro sucesos que causarán un restablecimiento de estadísticas, que actualizará esta indicación de fecha y hora:</p> <ul style="list-style-type: none"> • Se llama al procedimiento WLM_COLLECT_STATS. • El proceso periódico de colección y restauración controlado por el parámetro de configuración WLM_COLLECT_INT provoca una colección y una restauración. • Se reactiva la base de datos. • La subclase de servicio para la que se notifican estadísticas se ha modificado y el cambio se ha confirmado. <p>La indicación de fecha y hora de LAST_RESET está en el horario local.</p> |
| COORD_ACT_COMPLETED_TOTAL | BIGINT | <p>Número total de actividades de coordinación que han enviado los usuarios desde el último restablecimiento que se han completado satisfactoriamente. Esta cuenta se actualiza a medida que se completa cada actividad.</p> <p>Si correlaciona una actividad con una subclase de servicio diferente, esta actividad contará únicamente para establecer el total de la subclase en la que se completa.</p> |

Tabla 292. Información devuelta para WLM_GET_SERVICE_SUBCLASS_STATS (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|--------------------------|---------------|--|
| COORD_ACT_ABORTED_TOTAL | BIGINT | <p>Número total de actividades de coordinación que han enviado los usuarios desde el último restablecimiento que se han completado con errores. Esta cuenta se actualiza a medida que termina anormalmente cada actividad.</p> <p>Si correlaciona una actividad con una subclase de servicio diferente, esta actividad contará únicamente para establecer el total de la subclase en la que se aborta.</p> |
| COORD_ACT_REJECTED_TOTAL | BIGINT | <p>Número total de actividades de coordinación que han enviado los usuarios desde el último restablecimiento que se han rechazado antes de su ejecución en lugar de permitirse su ejecución. Una actividad se cuenta como rechazada cuando se impide su ejecución mediante una acción que impide el trabajo de ejecución o mediante un umbral de predicción. Esta cuenta se actualiza a medida que se rechaza cada actividad.</p> |
| CONCURRENT_ACT_TOP | INTEGER | <p>Número más alto de actividades simultáneas en cualquier nivel de anidamiento que se están ejecutando (lo que incluye las desocupadas y las en espera) y en cola que se ha alcanzado para esta subclase de servicio.</p> |
| COORD_ACT_LIFETIME_TOP | BIGINT | <p>Indicador de nivel alto correspondiente a la duración de actividad de coordinación, contado a todos los niveles de anidamiento. Es nulo si COLLECT AGGREGATE ACTIVITY DATA de la clase de servicio es NONE. Las unidades son milisegundos.</p> <p>Para utilizar esta estadística eficazmente cuando la clase de servicio incluye subclases que están correlacionadas, es necesario agregar la marca de límite superior COORD_ACT_LIFETIME_TOP de la subclase de servicio con la del resto de subclases afectadas por el mismo umbral o umbrales de correlación. Debe agregar estos valores ya que una actividad puede completarse después de haberse correlacionado con una subclase de servicio distinta, pero el tiempo que dedica la actividad en otras subclases de servicio antes de correlacionarse sólo cuenta para establecer el total de la clase de servicio en la que se completa.</p> |

Tabla 292. Información devuelta para WLM_GET_SERVICE_SUBCLASS_STATS (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|---------------------------|---------------|--|
| COORD_ACT_LIFETIME_AVG | DOUBLE | <p>Media aritmética de duración para actividades de coordinación a nivel de anidamiento 0 asociadas a esta subclase de servicio desde el último restablecimiento. Si se ha desbordado el promedio del que se hace un seguimiento interno, se devuelve el valor -2. Es nulo si COLLECT AGGREGATE ACTIVITY DATA de la clase de servicio es NONE. Las unidades son milisegundos.</p> <p>El valor de COORD_ACT_LIFETIME_AVG de una subclase de servicio no se ve afectado por las actividades que pasan a través de la subclase pero que están correlacionadas con una subclase distinta antes de completarse.</p> |
| COORD_ACT_LIFETIME_STDDEV | DOUBLE | <p>Desviación estándar de duración para actividades de coordinación a nivel de anidamiento 0 asociadas a esta subclase de servicio desde el último restablecimiento. Es nulo si COLLECT AGGREGATE ACTIVITY DATA de la clase de servicio es NONE. Las unidades son milisegundos. Esta desviación estándar se calcula a partir del histograma de duración de las actividades de coordinación y puede resultar imprecisa si el histograma no se ha dimensionado correctamente para dar cabida a los datos. Se devuelve el valor -1 si alguno de los valores cae dentro de la bandeja del último histograma.</p> <p>El valor de COORD_ACT_LIFETIME_STDDEV de una subclase de servicio no se ve afectado por las actividades que pasan a través de la subclase de servicio pero que están correlacionadas con una subclase distinta antes de completarse.</p> |
| COORD_ACT_EXEC_TIME_AVG | DOUBLE | <p>Media aritmética de tiempos de ejecución para actividades de coordinación a nivel de anidamiento 0 asociadas a esta subclase de servicio desde el último restablecimiento. Si se ha desbordado el promedio del que se hace un seguimiento interno, se devuelve el valor -2. Es nulo si COLLECT AGGREGATE ACTIVITY DATA de la clase de servicio es NONE. Las unidades son milisegundos.</p> <p>El promedio del tiempo de ejecución de una subclase de servicio no se ve afectado por las actividades que pasan a través de la subclase pero que están correlacionadas con una subclase distinta antes de completarse.</p> |

Tabla 292. Información devuelta para WLM_GET_SERVICE_SUBCLASS_STATS (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-----------------------------|---------------|---|
| COORD_ACT_EXEC_TIME_STDDEV | DOUBLE | <p>Desviación estándar de tiempos de ejecución para actividades de coordinación a nivel de anidamiento 0 asociadas a esta subclase de servicio desde el último restablecimiento. Las unidades son milisegundos. Esta desviación estándar se calcula a partir del histograma de tiempos de ejecución de las actividades de coordinación y puede resultar imprecisa si el histograma no se ha dimensionado correctamente para dar cabida a los datos. Se devuelve el valor -1 si alguno de los valores cae dentro de la bandeja del último histograma.</p> <p>La desviación estándar del tiempo de ejecución de una subclase de servicio no se ve afectada por las actividades que pasan a través de la subclase pero que están correlacionadas con una subclase distinta antes de completarse.</p> |
| COORD_ACT_QUEUE_TIME_AVG | DOUBLE | <p>Media aritmética de tiempo en cola para actividades de coordinación a nivel de anidamiento 0 asociadas a esta subclase de servicio desde el último restablecimiento. Si se ha desbordado el promedio del que se hace un seguimiento interno, se devuelve el valor -2. Es nulo si COLLECT AGGREGATE ACTIVITY DATA de la clase de servicio es NONE. Las unidades son milisegundos.</p> <p>El promedio del tiempo de cola sólo se cuenta para establecer el total de la subclase de servicio en la que se ha puesto en cola la actividad.</p> |
| COORD_ACT_QUEUE_TIME_STDDEV | DOUBLE | <p>Desviación estándar de tiempo en cola para actividades de coordinación a nivel de anidamiento 0 asociadas a esta subclase de servicio desde el último restablecimiento. Es nulo si COLLECT AGGREGATE ACTIVITY DATA de la clase de servicio es NONE. Las unidades son milisegundos. Esta desviación estándar se calcula a partir del histograma de tiempos en cola de las actividades de coordinación y puede resultar imprecisa si el histograma no se ha dimensionado correctamente para dar cabida a los datos. Se devuelve el valor -1 si alguno de los valores cae dentro de la bandeja del último histograma.</p> <p>La desviación estándar del tiempo de cola sólo se cuenta para establecer el total de la subclase de servicio en la que se ha puesto en cola la actividad.</p> |
| NUM_REQUESTS_ACTIVE | BIGINT | Número de solicitudes que se están ejecutando en la subclase de servicio en el momento de ejecutar esta función de tabla. |

Tabla 292. Información devuelta para WLM_GET_SERVICE_SUBCLASS_STATS (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|--------------------------|---------------|--|
| NUM_REQUESTS_TOTAL | BIGINT | <p>Número de solicitudes que se han acabado de ejecutar en esta subclase de servicio desde la última restauración. Esto se aplica a cualquier solicitud, independientemente de su pertenencia a una actividad. Si COLLECT AGGREGATE REQUEST DATA en esta subclase de servicio está establecido en NONE, el valor de esta columna es NULL.</p> <p>El NUM_REQUESTS_TOTAL de una subclase de servicio no se ve afectado por peticiones que pasan a través de la subclase de servicio pero que no se completan en la misma.</p> |
| REQUEST_EXEC_TIME_AVG | DOUBLE | <p>Media aritmética de tiempos de ejecución para solicitudes asociadas a esta subclase de servicio desde el último restablecimiento. Las unidades son milisegundos. Si se ha desbordado el promedio del que se hace un seguimiento interno, se devuelve el valor -2. Si COLLECT AGGREGATE REQUEST DATA en esta clase de servicio está establecido en NONE, el valor de esta columna es NULL.</p> <p>El promedio del tiempo de ejecución de una subclase de servicio no se ve afectada por peticiones que pasan a través de la subclase pero que no se completan en la misma.</p> |
| REQUEST_EXEC_TIME_STDDEV | DOUBLE | <p>Desviación estándar de tiempos de ejecución para solicitudes asociadas a esta subclase de servicio desde el último restablecimiento. Las unidades son milisegundos. Si COLLECT AGGREGATE REQUEST DATA en esta clase de servicio está establecido en NONE, el valor de esta columna es NULL. Esta desviación estándar se calcula a partir del histograma de tiempos de ejecución de solicitudes y puede resultar imprecisa si el histograma no se ha dimensionado correctamente para dar cabida a los datos. Se devuelve el valor -1 si alguno de los valores cae dentro de la bandeja del último histograma.</p> <p>La desviación estándar del tiempo de ejecución de una subclase de servicio no se ve afectada por peticiones que pasan a través de la subclase pero que no se completan en la misma.</p> |

Tabla 292. Información devuelta para WLM_GET_SERVICE_SUBCLASS_STATS (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------------|---------------|---|
| REQUEST_EXEC_TIME_TOTAL | BIGINT | <p>Suma de tiempos de ejecución para solicitudes asociadas a esta subclase de servicio desde el último restablecimiento. Las unidades son milisegundos. Si COLLECT AGGREGATE REQUEST DATA en esta clase de servicio está establecido en NONE, el valor de esta columna es NULL. Este total se calcula a partir del histograma de tiempos de ejecución de solicitudes y puede resultar impreciso si el histograma no se ha dimensionado correctamente para dar cabida a los datos. Se devuelve el valor -1 si alguno de los valores cae dentro de la bandeja del último histograma.</p> <p>El tiempo de ejecución total de una subclase de servicio no se ve afectado por peticiones que pasan a través de la subclase pero que no se completan en la misma.</p> |

WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES - Devolver una lista de actividades

La función WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES devuelve la lista de todas las actividades enviadas una la aplicación especificada en una partición especificada y que aún no se han completado.

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la "WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES_V97 - Devolver una lista de actividades" en la página 908.

Sintaxis

```

▶—WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES—(—descriptorcontexto_aplicación—, —————▶
▶—númparticiónbd—)—————▶

```

El esquema es SYSPROC.

Parámetros de la función de tabla

descriptorcontexto_aplicación

Argumento de entrada de tipo BIGINT que especifica un descriptor de contexto de aplicación para el que se debe devolver una lista de actividades. Si el argumento es nulo, se recuperan datos correspondientes a todas las aplicaciones de la base de datos.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número de partición válido en la misma instancia que la base de datos conectada actualmente. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos. Si se especifica el valor nulo, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE sobre la función
WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES.

Ejemplo

Después de identificar el descriptor de contexto de aplicación, se pueden buscar todas las actividades que se ejecutan actualmente en dicha aplicación. Por ejemplo, supongamos que un administrador desea tener una lista de las actividades de una aplicación cuyo descriptor de contexto de aplicación, determinado mediante el mandato LIST APPLICATIONS, es 1. El administrador ejecuta la consulta siguiente:

```
SELECT SUBSTR(CHAR(COORD_PARTITION_NUM),1,5) AS COORD,
       SUBSTR(CHAR(DBPARTITIONNUM),1,4) AS PART,
       SUBSTR(CHAR(UOW_ID),1,5) AS UOWID,
       SUBSTR(CHAR(ACTIVITY_ID),1,5) AS ACTID,
       SUBSTR(CHAR(PARENT_UOW_ID),1,8) AS PARUOWID,
       SUBSTR(CHAR(PARENT_ACTIVITY_ID),1,8) AS PARACTID,
       ACTIVITY_TYPE AS ACTTYPE,
       SUBSTR(CHAR(NESTING_LEVEL),1,7) AS NESTING
FROM TABLE(WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES(1, -2)) AS WLOACTS
ORDER BY PART, UOWID, ACTID
```

Un ejemplo de resultado de la consulta es el siguiente:

| COORD | PART | UOWID | ACTID | PARUOWID | PARACTID | ACTTYPE | NESTING |
|-------|------|-------|-------|----------|----------|----------|---------|
| 0 | 0 | 2 | 3 | - | - | CALL | 0 |
| 0 | 0 | 2 | 5 | 2 | 3 | READ_DML | 1 |

Información devuelta

Tabla 293. Información devuelta por WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES

| Nombre de columna | Tipo de datos | Descripción |
|---------------------|---------------|--|
| APPLICATION_HANDLE | BIGINT | ID exclusivo a nivel del sistema para la aplicación. En una base de datos de una sola partición, este ID consiste en un recuento de 16 bits. En una base de datos de varias particiones, este ID consiste en el número de partición de coordinación concatenado con un recuento de 16 bits. Además, este ID es el mismo en cada partición en la que la aplicación realiza una conexión secundaria. |
| DBPARTITIONNUM | SMALLINT | Número de partición de la que se ha recopilado este registro. |
| COORD_PARTITION_NUM | SMALLINT | La partición de coordinación de la actividad. |
| LOCAL_START_TIME | TIMESTAMP | La hora a la que esta actividad ha empezado a funcionar en la partición. El valor de la columna es nulo si la actividad ha entrado en el sistema pero está en cola y aún no se ha empezado a ejecutar. |
| UOW_ID | INTEGER | ID único de la unidad de trabajo en la que se ha iniciado la actividad. |
| ACTIVITY_ID | INTEGER | ID de actividad exclusivo dentro de una unidad de trabajo. |

Tabla 293. Información devuelta por *WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES* (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|--------------------|---------------|---|
| PARENT_UOW_ID | INTEGER | ID único de la unidad de trabajo en la que se ha iniciado la actividad padre de la actividad. Si la actividad no tiene actividad padre o si está en una partición remota, el valor de la columna es nulo. |
| PARENT_ACTIVITY_ID | INTEGER | ID exclusivo de actividad dentro de una unidad de trabajo correspondiente a la actividad padre de la actividad cuyo ID es el valor de la columna <i>ACTIVITY_ID</i> . Si la actividad no tiene actividad padre o si está en una partición remota, el valor de la columna es nulo. |
| ACTIVITY_STATE | VARCHAR(32) | <p>Estado de la actividad. Los valores posibles son los siguientes:</p> <p><i>CANCEL_PENDING</i> La actividad se ha cancelado porque no había ningún agente trabajando de forma activa en una petición correspondiente a la actividad. La siguiente vez que se envíe una petición como parte de la actividad, la actividad se cancelará y se generará un error SQL4725N.</p> <p><i>EXECUTING</i> Los agentes están trabajando de forma activa en una petición correspondiente a la actividad.</p> <p><i>IDLE</i> No hay ningún agente procesando de forma activa una petición correspondiente a la actividad.</p> <p><i>INITIALIZING</i> La actividad se ha enviado pero aún no se ha empezado a ejecutar. Durante el estado de inicialización, se aplican umbrales de predicción a la actividad para determinar si se permitirá que se ejecute la actividad.</p> |

Tabla 293. Información devuelta por `WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES` (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-------------------------------|---------------|--|
| ACTIVITY_STATE (continuación) | VARCHAR(32) | <p>Estado de la actividad. Los valores posibles son los siguientes:</p> <p><i>QP_CANCEL_PENDING</i> Este estado es el mismo que el estado <code>CANCEL_PENDING</code>, excepto que la actividad ha sido cancelada por Query Patroller en lugar de mediante el procedimiento <code>WLM_CANCEL_ACTIVITY</code>.</p> <p><i>QP_QUEUED</i> Query Patroller ha colocado la actividad en cola.</p> <p><i>QUEUED</i> Un umbral de colas de gestión de carga de trabajo pone en cola la actividad. En un entorno de base de datos particionada, este estado podría significar que el agente coordinador ha realizado un RPC en la partición de catálogo para obtener certificados de umbral y aún no ha recibido una respuesta. Este estado puede indicar que un umbral de colas de gestión de carga de trabajo ha colocado en cola la actividad o, si no ha transcurrido mucho tiempo, puede indicar que la actividad está en proceso de obtener sus tiquets. Para obtener una imagen más precisa de si la actividad se está colocando realmente en cola, determine qué agente está trabajando en la actividad y busque si el valor <code>EVENT_OBJECT</code> del agente de la partición de catálogo es <code>WLM_QUEUE</code>.</p> <p><i>TERMINATING</i> Ha finalizado la ejecución de la actividad y se está eliminando del sistema.</p> |

Tabla 293. Información devuelta por WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|-----------------------------|---------------|--|
| ACTIVITY_TYPE | VARCHAR(32) | Tipo de actividad. Los valores posibles son los siguientes: <ul style="list-style-type: none"> • CALL • DDL • LOAD • OTHER • READ_DML • WRITE_DML <p>Consulte “Identificar tipos de trabajo con clases de trabajo” en <i>Workload Manager Guide and Reference</i> para ver una descripción de los distintos tipos de sentencias de SQL asociados a cada tipo de actividad.</p> |
| NESTING_LEVEL | INTEGER | Profundidad a la que se anida esta actividad dentro de la actividad principal. |
| INVOCATION_ID | INTEGER | ID de invocación, que distingue una determinada invocación de esta actividad de otras del mismo nivel de anidamiento. |
| ROUTINE_ID | INTEGER | ID único de la rutina. |
| UTILITY_ID | INTEGER | Uno de los valores siguientes: <ul style="list-style-type: none"> • Si la actividad es un programa de utilidad, el valor es el ID del programa de utilidad. • Si la actividad no es un programa de utilidad, el valor es nulo. |
| SERVICE_CLASS_ID | INTEGER | ID exclusivo de la clase de servicio a la que pertenece esta actividad. |
| DATABASE_WORK_ACTION_SET_ID | INTEGER | Uno de los valores siguientes: <ul style="list-style-type: none"> • Si esta actividad se ha clasificado en una clase de trabajo de ámbito de base de datos, el valor es el ID del conjunto de la clase de trabajo del que esta clase de trabajo es miembro. • Si esta actividad no se ha clasificado en una clase de trabajo de ámbito de base de datos, el valor es nulo. |
| DATABASE_WORK_CLASS_ID | INTEGER | Uno de los valores siguientes: <ul style="list-style-type: none"> • Si esta actividad se ha clasificado en una clase de trabajo de ámbito de base de datos, el valor es el ID del conjunto de la clase de trabajo. • Si esta actividad no se ha clasificado en una clase de trabajo de ámbito de base de datos, el valor es nulo. |

Tabla 293. Información devuelta por WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|----------------------------------|---------------|---|
| SERVICE_CLASS_WORK_ACTION_SET_ID | INTEGER | Uno de los valores siguientes: <ul style="list-style-type: none"> • Si esta actividad se ha clasificado en una clase de trabajo de ámbito de clase de servicio, el valor es el ID del conjunto de la acción de trabajo asociado al conjunto de la clase de trabajo al que pertenece esta clase de trabajo. • Si esta actividad no se ha clasificado en una clase de trabajo de ámbito de clase de servicio, el valor es nulo. |
| SERVICE_CLASS_WORK_CLASS_ID | INTEGER | Uno de los valores siguientes: <ul style="list-style-type: none"> • Si esta actividad se ha clasificado en una clase de trabajo de ámbito de clase de servicio, el valor es el ID de la clase de trabajo asociado a esta actividad. • Si esta actividad no se ha clasificado en una clase de trabajo de ámbito de clase de servicio, el valor es nulo. |

WLM_GET_WORKLOAD_STATS - Devolver estadísticas de carga de trabajo

Nota: Esta función de tabla ha quedado obsoleta y se ha sustituido por la "Función de tabla WLM_GET_WORKLOAD_STATS_V97 - Devolver estadísticas de carga de trabajo" en la página 913.

Esta función devuelve estadísticas de carga de trabajo correspondientes a cada combinación de nombre de carga de trabajo y número de partición de base de datos.

Sintaxis

►►—WLM_GET_WORKLOAD_STATS—(—nombre_cargatrabajo—,—númparticiónbd—)————◄◄

El esquema es SYSPROC.

Parámetros de la función de tabla

nombre_cargatrabajo

Argumento de entrada de tipo VARCHAR(128) que especifica una determinada carga de trabajo para la que se tienen que devolver estadísticas. Si el argumento es NULL o una serie vacía, se devuelven estadísticas correspondientes a todas las cargas de trabajo.

númparticiónbd

Argumento de entrada de tipo INTEGER que especifica un número de partición válido en la misma instancia que la base de datos conectada actualmente al llamar a esta función. Especifique -1 para la partición de base de datos actual o -2 para todas las particiones de base de datos. Si se especifica un valor nulo, se establece -1 de forma implícita.

Autorización

Privilegio EXECUTE sobre la función WLM_GET_WORKLOAD_STATS.

Ejemplo

Es posible que un administrador desee revisar las estadísticas correspondientes a cargas de trabajo. Puede hacerlo utilizando la siguiente consulta:

```
SELECT SUBSTR(WORKLOAD_NAME,1,22) AS WL_DEF_NAME,  
       SUBSTR(CHAR(DBPARTITIONNUM),1,4) AS PART,  
       CONCURRENT_WLO_TOP AS WLO_TOP,  
       CONCURRENT_WLO_ACT_TOP AS WLO_ACT_TOP  
FROM TABLE(WLM_GET_WORKLOAD_STATS(CAST(NULL AS VARCHAR(128)), -2))  
AS WLSTATS  
ORDER BY WL_DEF_NAME, PART
```

A continuación se muestra un ejemplo de salida de esta consulta.

| WL_DEF_NAME | PART | WLO_TOP | WLO_ACT_TOP |
|------------------------|------|---------|-------------|
| MYUSERWORKLOAD | 0 | 2 | 8 |
| MYUSERWORKLOAD | 1 | 0 | 0 |
| SYSDEFAULTUSERWORKLOAD | 0 | 1 | 1 |
| SYSDEFAULTUSERWORKLOAD | 1 | 0 | 0 |

Aquí se ve que en la partición 0, el número más alto de apariciones simultáneas de la carga de trabajo MYUSERWORKLOAD ha sido 2 y que el número más alto de actividades simultáneas en cualquiera de estas apariciones de carga de trabajo ha sido 8.

Nota sobre uso

Esta función devuelve una fila para cada combinación de nombre de carga de trabajo y número de partición de base de datos. No se pueden agregar las cargas de trabajo, las particiones y las clases de servicio. Sin embargo, se puede lograr la agregación mediante consultas de SQL.

Información devuelta

Tabla 294. Información devuelta por WLM_GET_WORKLOAD_STATS

| Nombre de columna | Tipo de datos | Descripción |
|-------------------|---------------|---|
| WORKLOAD_NAME | VARCHAR(128) | Nombre de la carga de trabajo de la que se ha recopilado este registro. |
| DBPARTITIONNUM | SMALLINT | Número de la partición de la que se ha recopilado este registro. |

Tabla 294. Información devuelta por WLM_GET_WORKLOAD_STATS (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|---------------------------|---------------|---|
| LAST_RESET | TIMESTAMP | <p>Hora en que se restauraron las estadísticas por última vez. Se pueden producir cuatro sucesos que causarán un restablecimiento de estadísticas, que actualizará esta indicación de fecha y hora:</p> <ul style="list-style-type: none"> • Se llama al procedimiento WLM_COLLECT_STATS. • El proceso periódico de colección y restauración controlado por el parámetro de configuración WLM_COLLECT_INT provoca una colección y una restauración. • Se reactiva la base de datos. • La carga de trabajo para la que se notifican estadísticas se ha modificado y el cambio se ha confirmado. <p>La indicación de fecha y hora de LAST_RESET está en el horario local.</p> |
| CONCURRENT_WLO_TOP | INTEGER | Número más alto de apariciones simultáneas de una determinada carga de trabajo de esta partición desde la última restauración. |
| CONCURRENT_WLO_ACT_TOP | INTEGER | Número más alto de actividades simultáneas (incluidas las de coordinación y las anidadas) en ejecución (lo que incluye las desocupadas y las en espera) y en cola que se ha alcanzado en cualquier aparición de esta carga de trabajo desde el último restablecimiento. Lo actualiza cada aparición de carga de trabajo al final de su unidad de trabajo. |
| COORD_ACT_COMPLETED_TOTAL | BIGINT | Número total de actividades de coordinación a cualquier nivel de anidamiento asignadas a cualquier aparición de esta carga de trabajo que se han completado desde el último restablecimiento. Lo actualiza cada aparición de carga de trabajo al final de su unidad de trabajo. |
| COORD_ACT_ABORTED_TOTAL | BIGINT | Número total de actividades de coordinación a cualquier nivel de anidamiento asignadas a cualquier aparición de esta carga de trabajo que han terminado anormalmente antes de finalizar desde el último restablecimiento. Lo actualiza cada aparición de carga de trabajo al final de su unidad de trabajo. |
| COORD_ACT_REJECTED_TOTAL | BIGINT | Número total de actividades de coordinación a cualquier nivel de anidamiento asignadas a cualquier aparición de esta carga de trabajo que se han rechazado antes de su ejecución desde el último restablecimiento. Lo actualiza cada aparición de carga de trabajo al final de su unidad de trabajo. Una actividad se cuenta como rechazada cuando se impide su ejecución mediante una acción que impide el trabajo de ejecución o mediante un umbral de predicción. Tenga en cuenta que, a diferencia de la columna del mismo nombre de la función WLM_GET_SERVICE_SUBCLASS_STATS, también se cuentan los rechazos producidos antes de que una actividad se pueda asignar a una clase de servicio. Un ejemplo de este tipo de rechazo se produce cuando una actividad viola el umbral ConcurrentWorkloadOccurrences. |

Tabla 294. Información devuelta por WLM_GET_WORKLOAD_STATS (continuación)

| Nombre de columna | Tipo de datos | Descripción |
|---------------------|---------------|--|
| WLO_COMPLETED_TOTAL | BIGINT | El número de apariciones de carga de trabajo que hay que completar desde la última restauración. |

Apéndice A. Visión general de la información técnica de DB2

La información técnica de DB2 está disponible a través de las herramientas y los métodos siguientes:

- Centro de información de DB2
 - Temas (Tareas, concepto y temas de consulta)
 - Ayuda para herramientas de DB2
 - Programas de ejemplo
 - Guías de aprendizaje
- Manuales de DB2
 - Archivos PDF (descargables)
 - Archivos PDF (desde el DVD con PDF de DB2)
 - Manuales en copia impresa
- Ayuda de línea de mandatos
 - Ayuda de mandatos
 - Ayuda de mensajes

Nota: Los temas del Centro de información de DB2 se actualizan con más frecuencia que los manuales en PDF o impresos. Para obtener la información más actualizada, instale las actualizaciones de la documentación cuando estén disponibles, o consulte el Centro de información de DB2 en ibm.com.

Puede acceder a información técnica adicional de DB2 como, por ejemplo, notas técnicas, documentos técnicos y publicaciones IBM Redbooks en línea, en el sitio ibm.com. Acceda al sitio de la biblioteca de software de gestión de información de DB2 en <http://www.ibm.com/software/data/sw-library/>.

Comentarios sobre la documentación

Agradecemos los comentarios sobre la documentación de DB2. Si tiene sugerencias sobre cómo podemos mejorar la documentación de DB2, envíe un correo electrónico a db2docs@ca.ibm.com. El personal encargado de la documentación de DB2 lee todos los comentarios de los usuarios, pero no puede responderlos directamente. Proporcione ejemplos específicos siempre que sea posible de manera que podamos comprender mejor sus problemas. Si realiza comentarios sobre un tema o archivo de ayuda determinado, incluya el título del tema y el URL.

No utilice esta dirección de correo electrónico para contactar con el Soporte al cliente de DB2. Si tiene un problema técnico de DB2 que no está tratado por la documentación, consulte al centro local de servicio técnico de IBM para obtener ayuda.

Biblioteca técnica de DB2 en copia impresa o en formato PDF

Las tablas siguientes describen la biblioteca de DB2 que está disponible en el Centro de publicaciones de IBM en www.ibm.com/shop/publications/order. Los manuales de DB2 Versión 9.7 en inglés y las versiones traducidas en formato PDF se pueden descargar del sitio web www.ibm.com/support/docview.wss?rs=71&uid=swg2700947.

Aunque las tablas identifican los manuales en copia impresa disponibles, puede que dichos manuales no estén disponibles en su país o región.

El número de documento se incrementa cada vez que se actualiza un manual. Asegúrese de que lee la versión más reciente de los manuales, tal como aparece a continuación:

Nota: El Centro de información de DB2 se actualiza con más frecuencia que los manuales en PDF o impresos.

Tabla 295. Información técnica de DB2

| Nombre | Número de documento | Copia impresa disponible | Última actualización |
|---|----------------------------|---------------------------------|-----------------------------|
| <i>Consulta de las API administrativas</i> | SC11-3912-00 | Sí | Agosto de 2009 |
| <i>Rutinas y vistas administrativas</i> | SC11-3909-00 | No | Agosto de 2009 |
| <i>Call Level Interface Guide and Reference, Volume 1</i> | SC27-2437-00 | Sí | Agosto de 2009 |
| <i>Call Level Interface Guide and Reference, Volume 2</i> | SC27-2438-00 | Sí | Agosto de 2009 |
| <i>Consulta de mandatos</i> | SC11-3914-00 | Sí | Agosto de 2009 |
| <i>Data Movement Utilities Guide and Reference</i> | SC27-2440-00 | Sí | Agosto de 2009 |
| <i>Data Recovery and High Availability Guide and Reference</i> | SC27-2441-00 | Sí | Agosto de 2009 |
| <i>Database Administration Concepts and Configuration Reference</i> | SC27-2442-00 | Sí | Agosto de 2009 |
| <i>Database Monitoring Guide and Reference</i> | SC27-2458-00 | Sí | Agosto de 2009 |
| <i>Database Security Guide</i> | SC27-2443-00 | Sí | Agosto de 2009 |
| <i>Guía de DB2 Text Search</i> | SC11-3927-00 | Sí | Agosto de 2009 |
| <i>Developing ADO.NET and OLE DB Applications</i> | SC27-2444-00 | Sí | Agosto de 2009 |
| <i>Developing Embedded SQL Applications</i> | SC27-2445-00 | Sí | Agosto de 2009 |
| <i>Desarrollo de aplicaciones Java</i> | SC11-3907-00 | Sí | Agosto de 2009 |
| <i>Desarrollo de aplicaciones Perl, PHP, Python y Ruby on Rails</i> | SC11-3908-00 | No | Agosto de 2009 |
| <i>Developing User-defined Routines (SQL and External)</i> | SC27-2448-00 | Sí | Agosto de 2009 |
| <i>Getting Started with Database Application Development</i> | GI11-9410-00 | Sí | Agosto de 2009 |

Tabla 295. Información técnica de DB2 (continuación)

| Nombre | Número de documento | Copia impresa disponible | Última actualización |
|--|---------------------|--------------------------|----------------------|
| <i>Iniciación a la instalación y administración de DB2 en Linux y Windows</i> | GI11-8640-00 | Sí | Agosto de 2009 |
| <i>Globalization Guide</i> | SC27-2449-00 | Sí | Agosto de 2009 |
| <i>Instalación de servidores DB2</i> | SC11-3916-00 | Sí | Agosto de 2009 |
| <i>Instalación de clientes de servidor de datos de IBM</i> | SC11-3917-00 | No | Agosto de 2009 |
| <i>Consulta de mensajes Volumen 1</i> | SC11-3922-00 | No | Agosto de 2009 |
| <i>Consulta de mensajes Volumen 2</i> | SC11-3923-00 | No | Agosto de 2009 |
| <i>Net Search Extender Guía de administración y del usuario</i> | SC11-3926-00 | No | Agosto de 2009 |
| <i>Partitioning and Clustering Guide</i> | SC27-2453-00 | Sí | Agosto de 2009 |
| <i>pureXML Guide</i> | SC27-2465-00 | Sí | Agosto de 2009 |
| <i>Query Patroller Administration and User's Guide</i> | SC27-2467-00 | No | Agosto de 2009 |
| <i>Spatial Extender and Geodetic Data Management Feature Guía del usuario y manual de consulta</i> | SC11-3925-00 | No | Agosto de 2009 |
| <i>SQL Procedural Languages: Application Enablement and Support</i> | SC27-2470-00 | Sí | Agosto de 2009 |
| <i>Consulta de SQL, Volumen 1</i> | SC11-3910-00 | Sí | Agosto de 2009 |
| <i>Consulta de SQL, Volumen 2</i> | SC11-3911-00 | Sí | Agosto de 2009 |
| <i>Troubleshooting and Tuning Database Performance</i> | SC27-2461-00 | Sí | Agosto de 2009 |
| <i>Actualización a DB2 Versión 9.7</i> | SC11-3915-00 | Sí | Agosto de 2009 |
| <i>Guía de aprendizaje de Visual Explain</i> | SC11-3924-00 | No | Agosto de 2009 |
| <i>Novedades en DB2 Versión 9.7</i> | SC11-3921-00 | Sí | Agosto de 2009 |
| <i>Workload Manager Guide and Reference</i> | SC27-2464-00 | Sí | Agosto de 2009 |
| <i>XQuery Reference</i> | SC27-2466-00 | No | Agosto de 2009 |

Tabla 296. Información técnica específica de DB2 Connect

| Nombre | Número de documento | Copia impresa disponible | Última actualización |
|--|---------------------|--------------------------|----------------------|
| <i>Instalación y configuración de DB2 Connect Personal Edition</i> | SC11-3919-00 | Sí | Agosto de 2009 |
| <i>Instalación y configuración de servidores DB2 Connect</i> | SC11-3920-00 | Sí | Agosto de 2009 |
| <i>Guía del usuario de DB2 Connect</i> | SC11-3918-00 | Sí | Agosto de 2009 |

Tabla 297. Información técnica de Information Integration

| Nombre | Número de documento | Copia impresa disponible | Última actualización |
|---|---------------------|--------------------------|----------------------|
| <i>Information Integration: Administration Guide for Federated Systems</i> | SC19-1020-02 | Sí | Agosto de 2009 |
| <i>Information Integration: ASNCLP Program Reference for Replication and Event Publishing</i> | SC11-3900-04 | Sí | Agosto de 2009 |
| <i>Information Integration: Configuration Guide for Federated Data Sources</i> | SC19-1034-02 | No | Agosto de 2009 |
| <i>Information Integration: SQL Replication Guide and Reference</i> | SC11-3899-02 | Sí | Agosto de 2009 |
| <i>Information Integration: Introduction to Replication and Event Publishing</i> | GC19-1028-02 | Sí | Agosto de 2009 |

Pedido de manuales de DB2 en copia impresa

Si necesita manuales de DB2 en copia impresa, puede comprarlos en línea en varios países o regiones, pero no en todos. Siempre puede hacer pedidos de manuales de DB2 en copia impresa a través del representante local de IBM. Recuerde que algunas publicaciones en copia software del DVD *Documentación en PDF de DB2* no están disponibles en copia impresa. Por ejemplo, no está disponible la publicación *Consulta de mensajes de DB2* en copia impresa.

Las versiones impresas de muchas de las publicaciones de DB2 disponibles en el DVD de Documentación en PDF de DB2 se pueden solicitar a IBM por una cantidad. Dependiendo desde dónde realice el pedido, podrá solicitar manuales en línea, desde el Centro de publicaciones de IBM. Si la realización de pedidos en línea no está disponible en su país o región, siempre puede hacer pedidos de manuales de DB2 en copia impresa al representante local de IBM. Tenga en cuenta que no todas las publicaciones del DVD de Documentación en PDF de DB2 están disponibles en copia impresa.

Nota: La documentación más actualizada y completa de DB2 se conserva en el Centro de información de DB2 en <http://publib.boulder.ibm.com/infocenter/db2luw/v9r7>.

Para hacer pedidos de manuales de DB2 en copia impresa:

- Para averiguar si puede hacer pedidos de manuales de DB2 en copia impresa en línea en su país o región, consulte el Centro de publicaciones de IBM en el sitio <http://www.ibm.com/shop/publications/order>. Debe seleccionar un país, región o idioma para poder acceder a la información sobre pedidos de publicaciones y, a continuación, seguir las instrucciones sobre pedidos para su localidad.
- Para hacer pedidos de manuales de DB2 en copia impresa a través del representante local de IBM:
 1. Localice la información de contacto de su representante local desde uno de los siguientes sitios Web:
 - El directorio de IBM de contactos en todo el mundo en el sitio www.ibm.com/planetwide
 - El sitio Web de publicaciones de IBM en el sitio <http://www.ibm.com/shop/publications/order>. Tendrá que seleccionar su país, región o idioma para acceder a la página de presentación de las publicaciones apropiadas para su localidad. Desde esta página, siga el enlace "Acerca de este sitio".
 2. Cuando llame, indique que desea hacer un pedido de una publicación de DB2.
 3. Proporcione al representante los títulos y números de documento de las publicaciones que desee solicitar. Si desea consultar los títulos y los números de documento, consulte el apartado "Biblioteca técnica de DB2 en copia impresa o en formato PDF" en la página 1189.

Visualización de la ayuda para estados de SQL desde el procesador de línea de mandatos

Los productos DB2 devuelven un valor de SQLSTATE para las condiciones que pueden ser el resultado de una sentencia de SQL. La ayuda de SQLSTATE explica los significados de los estados de SQL y los códigos de las clases de estados de SQL.

Para iniciar la ayuda para estados de SQL, abra el procesador de línea de mandatos y entre:

```
? sqlstate o ? código de clase
```

donde *sqlstate* representa un estado de SQL válido de cinco dígitos y *código de clase* representa los dos primeros dígitos del estado de SQL.

Por ejemplo, ? 08003 visualiza la ayuda para el estado de SQL 08003, y ? 08 visualiza la ayuda para el código de clase 08.

Acceso a diferentes versiones del Centro de información de DB2

Para los temas de la versión 9.7 de DB2, el URL del Centro de información de DB2 es <http://publib.boulder.ibm.com/infocenter/db2luw/v9r7>

Para los temas de DB2 Versión 9.5, el URL del Centro de información de DB2 es <http://publib.boulder.ibm.com/infocenter/db2luw/v9r5/>

Para los temas de DB2 Versión 9, el URL del Centro de información de DB2 es <http://publib.boulder.ibm.com/infocenter/db2luw/v9/>

Para los temas de DB2 Versión 8, vaya al URL del Centro de información de la Versión 8 en el sitio: <http://publib.boulder.ibm.com/infocenter/db2luw/v8/>

Visualización de temas en su idioma preferido en el Centro de información de DB2

El Centro de información de DB2 intenta visualizar los temas en el idioma especificado en las preferencias del navegador. Si un tema no se ha traducido al idioma preferido, el Centro de información de DB2 visualiza dicho tema en inglés.

- Para visualizar temas en su idioma preferido en el navegador Internet Explorer:
 1. En Internet Explorer, pulse en el botón **Herramientas** —> **Opciones de Internet** —> **Idiomas...** Se abrirá la ventana Preferencias de idioma.
 2. Asegúrese de que su idioma preferido esté especificado como la primera entrada de la lista de idiomas.
 - Para añadir un nuevo idioma a la lista, pulse el botón **Agregar...**

Nota: La adición de un idioma no garantiza que el sistema tenga los fonts necesarios para visualizar los temas en el idioma preferido.
 - Para mover un idioma hacia el principio de la lista, seleccione el idioma y pulse el botón **Subir** hasta que el idioma esté en primer lugar en la lista de idiomas.
 3. Borre la antememoria del navegador y, a continuación, renueve la página para visualizar el Centro de información de DB2 en su idioma preferido.
- Para visualizar temas en su idioma preferido en un navegador Firefox o Mozilla:
 1. Seleccione el botón en la sección **Idiomas** del diálogo **Herramientas** —> **Opciones** —> **Avanzado**. Se visualizará el panel Idiomas en la ventana Preferencias.
 2. Asegúrese de que su idioma preferido esté especificado como la primera entrada de la lista de idiomas.
 - Para añadir un nuevo idioma a la lista, pulse el botón **Añadir...** a fin de seleccionar un idioma en la ventana Añadir idiomas.
 - Para mover un idioma hacia el principio de la lista, seleccione el idioma y pulse el botón **Subir** hasta que el idioma esté en primer lugar en la lista de idiomas.
 3. Borre la antememoria del navegador y, a continuación, renueve la página para visualizar el Centro de información de DB2 en su idioma preferido.

En algunas combinaciones de navegador y sistema operativo, también debe cambiar los valores regionales del sistema operativo al entorno local y al idioma de su elección.

Actualización del Centro de información de DB2 instalado en el sistema o en el servidor de intranet

El Centro de información de DB2 instalado en local se debe actualizar periódicamente.

Antes de empezar

Ya debe haber un Centro de información de DB2 Versión 9.7 instalado. Para obtener información adicional, consulte el tema “Instalación del Centro de información de DB2 utilizando el Asistente de instalación de DB2” en la publicación *Instalación de servidores DB2*. Todos los requisitos previos y las restricciones aplicables a la instalación del Centro de información se aplican también a la actualización del Centro de información.

Acerca de esta tarea

Un Centro de información de DB2 existente se puede actualizar automática o manualmente:

- Actualizaciones automáticas: actualiza las funciones y los idiomas del Centro de información existentes. Una ventaja adicional de las actualizaciones automáticas es que el Centro de información deja de estar disponible durante un período de tiempo mínimo mientras se realiza la actualización. Además, la ejecución de las actualizaciones automáticas se puede configurar como parte de otros trabajos de proceso por lotes que se ejecutan periódicamente.
- Actualizaciones manuales: se deben utilizar si se quieren añadir funciones o idiomas durante el proceso de actualización. Por ejemplo, un Centro de información en local se instaló inicialmente tanto en inglés como en francés, y ahora se desea instalar el idioma alemán. Con la actualización manual, se instalará el alemán y se actualizarán además las funciones y los idiomas existentes del Centro de información. No obstante, la actualización manual requiere que el usuario detenga, actualice y reinicie manualmente el Centro de información. El Centro de información no está disponible durante todo el proceso de actualización.

Procedimiento

Este tema detalla el proceso de las actualizaciones automáticas. Para conocer las instrucciones para la actualización manual, consulte el tema “Actualización manual del Centro de información de DB2 instalado en el sistema o en el servidor de intranet”.

Para actualizar automáticamente el Centro de información de DB2 instalado en el sistema o en el servidor de Intranet:

1. En sistemas operativos Linux,
 - a. Navegue hasta la vía de acceso en la que está instalado el Centro de información. Por omisión, el Centro de información de DB2 se instala en el directorio `/opt/ibm/db2ic/V9.7`.
 - b. Navegue desde el directorio de instalación al directorio `doc/bin`.
 - c. Ejecute el script `ic-update`:
`ic-update`
2. En sistemas operativos Windows,
 - a. Abra una ventana de mandatos.
 - b. Navegue hasta la vía de acceso en la que está instalado el Centro de información. Por omisión, el Centro de información de DB2 se instala en el directorio `<Archivos de programa>\IBM\Centro de información de DB2\Versión 9.7`, siendo `<Archivos de programa>` la ubicación del directorio Archivos de programa.
 - c. Navegue desde el directorio de instalación al directorio `doc\bin`.
 - d. Ejecute el archivo `ic-update.bat`:
`ic-update.bat`

Resultados

El Centro de información de DB2 se reinicia automáticamente. Si hay actualizaciones disponibles, el Centro de información muestra los temas nuevos y actualizados. Si no había actualizaciones del Centro de información disponibles, se añade un mensaje al archivo de anotaciones cronológicas. El archivo de anotaciones cronológicas está ubicado en el directorio `doc\eclipse\configuration`. El nombre del archivo de anotaciones cronológicas es un número generado aleatoriamente. Por ejemplo, `1239053440785.log`.

Actualización manual del Centro de información de DB2 instalado en el sistema o en el servidor de intranet

Si ha instalado localmente el Centro de información de DB2, puede obtener las actualizaciones de la documentación de IBM e instalarlas.

Para actualizar manualmente el Centro de información de DB2 instalado localmente es preciso que:

1. Detenga el Centro de información de DB2 en el sistema, y reinicie el Centro de información en modalidad autónoma. La ejecución del Centro de información en modalidad autónoma impide que otros usuarios de la red accedan al Centro de información y permite al usuario aplicar las actualizaciones. La versión para estaciones de trabajo del Centro de información de DB2 siempre se ejecuta en modalidad autónoma.
2. Utilice la función Actualizar para ver qué actualizaciones están disponibles. Si hay actualizaciones que debe instalar, puede utilizar la función Actualizar para obtenerlas y actualizarlas.

Nota: Si su entorno requiere la instalación de actualizaciones del Centro de información de DB2 en una máquina no conectada a Internet, duplique el sitio de actualizaciones en un sistema de archivos local utilizando una máquina que esté conectada a Internet y tenga instalado el Centro de información de DB2. Si muchos usuarios en la red van a instalar las actualizaciones de la documentación, puede reducir el tiempo necesario para realizar las actualizaciones duplicando también el sitio de actualizaciones localmente y creando un proxy para el sitio de actualizaciones.

Si hay paquetes de actualización disponibles, utilice la característica Actualizar para obtener los paquetes. Sin embargo, la característica Actualizar sólo está disponible en modalidad autónoma.

3. Detenga el Centro de información autónomo y reinicie el Centro de información de DB2 en su equipo.

Nota: En Windows 2008 y Windows Vista (y posterior), los mandatos listados más abajo deben ejecutarse como administrador. Para abrir un indicador de mandatos o una herramienta gráfica con privilegios de administrador completos, pulse con el botón derecho del ratón el atajo y, a continuación, seleccione **Ejecutar como administrador**.

Para actualizar el Centro de información de DB2 instalado en el sistema o en el servidor de Intranet:

1. Detenga el Centro de información de DB2.
 - En Windows, pulse **Inicio** → **Panel de control** → **Herramientas administrativas** → **Servicios**. A continuación, pulse con el botón derecho del ratón en el servicio **Centro de información de DB2** y seleccione **Detener**.

- En Linux, especifique el mandato siguiente:
`/etc/init.d/db2icdv97 stop`
2. Inicie el Centro de información en modalidad autónoma.
 - En Windows:
 - a. Abra una ventana de mandatos.
 - b. Navegue hasta la vía de acceso en la que está instalado el Centro de información. Por omisión, el Centro de información de DB2 se instala en el directorio <Archivos de programa>\IBM\Centro de información de DB2\Versión 9.7, siendo <Archivos de programa> la ubicación del directorio Archivos de programa.
 - c. Navegue desde el directorio de instalación al directorio doc\bin.
 - d. Ejecute el archivo help_start.bat:
`help_start.bat`
 - En Linux:
 - a. Navegue hasta la vía de acceso en la que está instalado el Centro de información. Por omisión, el Centro de información de DB2 se instala en el directorio /opt/ibm/db2ic/V9.7.
 - b. Navegue desde el directorio de instalación al directorio doc/bin.
 - c. Ejecute el script help_start:
`help_start`

Se abre el navegador Web por omisión de los sistemas para visualizar el Centro de información autónomo.

3. Pulse en el botón **Actualizar** (🔄). (JavaScript™ debe estar habilitado en el navegador.) En la derecha del panel del Centro de información, pulse en **Buscar actualizaciones**. Se visualiza una lista de actualizaciones para la documentación existente.
4. Para iniciar el proceso de instalación, compruebe las selecciones que desee instalar y, a continuación, pulse **Instalar actualizaciones**.
5. Cuando finalice el proceso de instalación, pulse **Finalizar**.
6. Detenga el Centro de información autónomo:
 - En Windows, navegue hasta el directorio doc\bin del directorio de instalación y ejecute el archivo help_end.bat:
`help_end.bat`

Nota: El archivo help_end de proceso por lotes contiene los mandatos necesarios para detener sin peligro los procesos que se iniciaron mediante el archivo help_start de proceso por lotes. No utilice Control-C ni ningún otro método para detener help_start.bat.

 - En Linux, navegue hasta el directorio de instalación doc/bin y ejecute el script help_end:
`help_end`

Nota: El script help_end contiene los mandatos necesarios para detener sin peligro los procesos que se iniciaron mediante el script help_start. No utilice ningún otro método para detener el script help_start.
7. Reinicie el Centro de información de DB2:
 - En Windows, pulse **Inicio** → **Panel de control** → **Herramientas administrativas** → **Servicios**. A continuación, pulse con el botón derecho del ratón en el servicio **Centro de información de DB2** y seleccione **Iniciar**.
 - En Linux, especifique el mandato siguiente:

```
/etc/init.d/db2icdv97 start
```

El Centro de información de DB2 actualizado visualiza los temas nuevos y actualizados.

Guías de aprendizaje de DB2

Las guías de aprendizaje de DB2 le ayudan a conocer diversos aspectos de productos DB2. Se proporcionan instrucciones paso a paso a través de lecciones.

Antes de comenzar

Puede ver la versión XHTML de la guía de aprendizaje desde el Centro de información en el sitio <http://publib.boulder.ibm.com/infocenter/db2help/>.

Algunas lecciones utilizan datos o código de ejemplo. Consulte la guía de aprendizaje para obtener una descripción de los prerrequisitos para las tareas específicas.

Guías de aprendizaje de DB2

Para ver la guía de aprendizaje, pulse el título.

“pureXML” en *pureXML Guide*

Configure una base de datos DB2 para almacenar datos XML y realizar operaciones básicas con el almacén de datos XML nativos.

“Visual Explain” en la *Guía de aprendizaje de Visual Explain*

Analizar, optimizar y ajustar sentencias de SQL para obtener un mejor rendimiento al utilizar Visual Explain.

Información de resolución de problemas de DB2

Existe una gran variedad de información para la resolución y determinación de problemas para ayudarle en la utilización de productos de base de datos DB2.

Documentación de DB2

Puede encontrar información sobre la resolución de problemas en la publicación *DB2 Troubleshooting Guide* o en la sección Conceptos fundamentales sobre bases de datos del Centro de información de DB2. En ellas encontrará información sobre cómo aislar e identificar problemas utilizando herramientas y programas de utilidad de diagnóstico de DB2, soluciones a algunos de los problemas más habituales y otros consejos sobre cómo solucionar problemas que podría encontrar en los productos DB2.

Sitio web de soporte técnico de DB2

Consulte el sitio Web de soporte técnico de DB2 si tiene problemas y desea obtener ayuda para encontrar las causas y soluciones posibles. El sitio de soporte técnico tiene enlaces a las publicaciones más recientes de DB2, notas técnicas, Informes autorizados de análisis del programa (APAR o arreglos de defectos), fixpacks y otros recursos. Puede buscar en esta base de conocimiento para encontrar posibles soluciones a los problemas.

Acceda al sitio Web de soporte técnico de DB2 en la dirección http://www.ibm.com/software/data/db2/support/db2_9/

Términos y condiciones

Los permisos para utilizar estas publicaciones se otorgan sujetos a los siguientes términos y condiciones.

Uso personal: Puede reproducir estas publicaciones para su uso personal, no comercial, siempre y cuando se mantengan los avisos sobre la propiedad. No puede distribuir, visualizar o realizar trabajos derivados de estas publicaciones, o de partes de las mismas, sin el consentimiento expreso de IBM.

Uso comercial: Puede reproducir, distribuir y visualizar estas publicaciones únicamente dentro de su empresa, siempre y cuando se mantengan todos los avisos sobre la propiedad. No puede realizar trabajos derivados de estas publicaciones, ni reproducirlas, distribuirlas o visualizarlas, ni de partes de las mismas fuera de su empresa, sin el consentimiento expreso de IBM.

Excepto lo expresamente concedido en este permiso, no se conceden otros permisos, licencias ni derechos, explícitos o implícitos, sobre las publicaciones ni sobre ninguna información, datos, software u otra propiedad intelectual contenida en el mismo.

IBM se reserva el derecho de retirar los permisos aquí concedidos cuando, a su discreción, el uso de las publicaciones sea en detrimento de su interés o cuando, según determine IBM, las instrucciones anteriores no se cumplan correctamente.

No puede descargar, exportar ni volver a exportar esta información excepto en el caso de cumplimiento total con todas las leyes y regulaciones vigentes, incluyendo todas las leyes y regulaciones sobre exportación de los Estados Unidos.

IBM NO GARANTIZA EL CONTENIDO DE ESTAS PUBLICACIONES. LAS PUBLICACIONES SE PROPORCIONAN "TAL CUAL" Y SIN GARANTÍA DE NINGUNA CLASE, NI EXPLÍCITA NI IMPLÍCITA, INCLUYENDO PERO SIN LIMITARSE A LAS GARANTÍAS IMPLÍCITAS DE COMERCIALIZACIÓN, NO VULNERACIÓN E IDONEIDAD PARA UN FIN DETERMINADO.

Apéndice B. Avisos

Esta información ha sido desarrollada para productos y servicios que se ofrecen en Estados Unidos de América. La información acerca de productos que no son IBM se basa en la información disponible cuando se publicó este documento por primera vez y está sujeta a cambio.

Es posible que IBM no comercialice en otros países algunos productos, servicios o características descritos en este manual. Consulte al representante local de IBM para obtener información sobre los productos y servicios que actualmente pueden adquirirse en su zona. Cualquier referencia a un producto, programa o servicio de IBM no pretende afirmar ni implicar que sólo se pueda utilizar dicho producto, programa o servicio de IBM. En su lugar se puede utilizar cualquier producto, programa o servicio funcionalmente equivalente que no vulnere ninguno de los derechos de propiedad intelectual de IBM. Sin embargo, es responsabilidad del usuario evaluar y verificar el funcionamiento de cualquier producto, programa o servicio que no sea de IBM.

IBM puede tener patentes o solicitudes de patentes en tramitación que afecten al tema tratado en este documento. La posesión de este documento no confiere ninguna licencia sobre dichas patentes. Puede realizar consultas sobre licencias escribiendo a:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
EE.UU.

Para realizar consultas sobre licencias referentes a información de juegos de caracteres de doble byte (DBCS), puede ponerse en contacto con el Departamento de Propiedad Intelectual de IBM de su país o escribir a:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
3-2-12, Roppongi, Minato-ku, Tokyo 106-8711 Japón

El párrafo siguiente no es aplicable al Reino Unido ni a ningún país/región en donde tales disposiciones sean incompatibles con la legislación local:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROPORCIONA ESTA PUBLICACIÓN "TAL CUAL", SIN GARANTÍA DE NINGUNA CLASE, NI EXPLÍCITA NI IMPLÍCITA, INCLUIDAS, PERO SIN LIMITARSE A ELLAS, LAS GARANTÍAS IMPLÍCITAS DE NO VULNERACIÓN DE DERECHOS, COMERCIALIZACIÓN O IDONEIDAD PARA UN FIN DETERMINADO. Algunos estados no permiten la exclusión de garantías expresas o implícitas en determinadas transacciones, por lo que es posible que esta declaración no sea aplicable en su caso.

Esta publicación puede contener inexactitudes técnicas o errores tipográficos. Periódicamente se efectúan cambios en la información aquí contenida; dichos cambios se incorporarán a las nuevas ediciones de la publicación. IBM puede

efectuar, en cualquier momento y sin previo aviso, mejoras y cambios en los productos y programas descritos en esta publicación.

Las referencias hechas en esta publicación a sitios web que no son de IBM se proporcionan sólo para la comodidad del usuario y no constituyen un aval de esos sitios web. La información de esos sitios web no forma parte de la información del presente producto de IBM y la utilización de esos sitios web se realiza bajo la responsabilidad del usuario.

IBM puede utilizar o distribuir cualquier información que se le facilite de la manera que considere adecuada, sin contraer por ello ninguna obligación con el remitente.

Los licenciatarios de este programa que deseen obtener información sobre él con el fin de habilitar: (i) el intercambio de información entre programas creados de forma independiente y otros programas (incluido éste) y (ii) el uso mutuo de la información intercambiada, deben ponerse en contacto con:

IBM Canada Limited
Office of the Lab Director
8200 Warden Avenue
Markham, Ontario
L6G 1C7
CANADÁ

Dicha información puede estar disponible, sujeta a los términos y condiciones apropiados, incluido en algunos casos el pago de una tarifa.

El programa bajo licencia descrito en este documento y todo el material bajo licencia asociado a él, los proporciona IBM según los términos del Acuerdo de Cliente de IBM, el Acuerdo Internacional de Programas Bajo Licencia de IBM o cualquier acuerdo equivalente entre el usuario e IBM.

Los datos de rendimiento contenidos en este documento se obtuvieron en un entorno controlado. Por lo tanto, los resultados obtenidos en otros entornos operativos pueden variar significativamente. Algunas mediciones pueden haberse realizado en sistemas experimentales y no es seguro que estas mediciones sean las mismas en los sistemas disponibles comercialmente. Además, algunas mediciones pueden haberse calculado mediante extrapolación. Los resultados reales pueden variar. Los usuarios del presente manual deben verificar los datos aplicables para su entorno específico.

La información referente a productos que no son de IBM se ha obtenido de los proveedores de esos productos, de sus anuncios publicados o de otras fuentes disponibles públicamente. IBM no ha probado esos productos y no puede confirmar la exactitud del rendimiento, la compatibilidad ni ninguna otra afirmación referente a productos que no son de IBM. Las preguntas sobre las prestaciones de productos que no son de IBM deben dirigirse a los proveedores de esos productos.

Todas las declaraciones de intenciones de IBM están sujetas a cambio o cancelación sin previo aviso, y sólo representan objetivos.

Este manual puede contener ejemplos de datos e informes que se utilizan en operaciones comerciales diarias. Para ilustrarlos de la forma más completa posible, los ejemplos incluyen nombres de personas, empresas, marcas y productos. Todos

estos nombres son ficticios y cualquier similitud con nombres y direcciones utilizados por una empresa real es totalmente fortuita.

LICENCIA DE COPYRIGHT:

Este manual contiene programas de aplicaciones de ejemplo escritos en lenguaje fuente, que muestran técnicas de programación en diversas plataformas operativas. Puede copiar, modificar y distribuir estos programas de ejemplo como desee, sin pago alguno a IBM con la intención de desarrollar, utilizar, comercializar o distribuir programas de aplicaciones de acuerdo con la interfaz de programación de aplicaciones correspondiente a la plataforma operativa para la que están escritos los programas de ejemplo. Estos ejemplos no se han probado exhaustivamente bajo todas las condiciones. Por lo tanto, IBM no puede asegurar ni implicar la fiabilidad, utilidad o función de estos programas. Los programas de ejemplo se proporcionan "TAL CUAL", sin ningún tipo de garantía. IBM no se hará responsable de los daños derivados de la utilización que haga el usuario de los programas de ejemplo.

Cada copia o parte de estos programas de ejemplo o cualquier trabajo derivado debe incluir una nota de copyright como la siguiente:

© (*nombre de la empresa*) (*año*). Partes de este código proceden de programas de ejemplo de IBM Corp. © Copyright IBM Corp. *_entre el o los años_*. Reservados todos los derechos.

Marcas registradas

IBM, el logotipo de IBM e ibm.com son marcas registradas de International Business Machines Corp., que se han registrado en muchas otras jurisdicciones. Otros nombres de productos y servicios pueden ser marcas registradas de IBM o de otras empresas. Puede consultarse en línea una lista actualizada de las marcas registradas de IBM en la sección Copyright and trademark information de la web www.ibm.com/legal/copytrade.shtml.

Los siguientes términos son marcas registradas de otras empresas.

- Linux es una marca registrada de Linus Torvalds en los Estados Unidos y/o en otros países.
- Java y todas las marcas registradas basadas en Java son marcas registradas de Sun Microsystems, Inc. en los Estados Unidos y/o en otros países.
- UNIX es una marca registrada de The Open Group en los Estados Unidos y/o en otros países.
- Intel[®], el logotipo de Intel, Intel Inside[®], el logotipo de Intel Inside, Intel[®] Centrino[®], el logotipo de Intel Centrino, Celeron[®], Intel[®] Xeon[®], Intel SpeedStep[®], Itanium[®] y Pentium[®] son marcas registradas de Intel Corporation o de sus empresas subsidiarias en Estados Unidos y en otros países.
- Microsoft[®], Windows, Windows NT[®] y el logotipo de Windows son marcas registradas de Microsoft Corporation en los Estados Unidos y/o en otros países.

Otros nombres de empresas, productos o servicios, pueden ser marcas registradas o marcas de servicio de otras empresas.

Índice

A

- actualizaciones
 - Centro de información de DB2 1194, 1196
- ADD CONTACT, mandato
 - sintaxis utilizando ADMIN_CMD 39
- ADD CONTACTGROUP, mandato
 - sintaxis utilizando ADMIN_CMD 41
- ADMIN_CMD, procedimiento 37
 - eliminación de mensajes 242
 - mandatos soportados
 - ADD CONTACT 39
 - ADD CONTACTGROUP 41
 - AUTOCONFIGURE 42
 - BACKUP DATABASE 47
 - DESCRIBE 55
 - DROP CONTACT 69
 - DROP CONTACTGROUP 70
 - EXPORT 71
 - FORCE APPLICATION 83
 - GET STMM TUNING DBPARTITIONNUM 85
 - IMPORT 86
 - INITIALIZE TAPE 114
 - LOAD 115
 - PRUNE HISTORY/LOGFILE 158
 - QUIESCE DATABASE 160
 - QUIESCE TABLESPACES FOR TABLE 162
 - REDISTRIBUTE DATABASE PARTITION GROUP 165
 - REORG INDEXES/TABLE 176
 - RESET ALERT CONFIGURATION 190
 - RESET DATABASE CONFIGURATION 192
 - RESET DATABASE MANAGER CONFIGURATION 194
 - REWIND TAPE 195
 - RUNSTATS 196
 - SET TAPE POSITION 208
 - UNQUIESCE DATABASE 209
 - UPDATE ALERT CONFIGURATION 210
 - UPDATE CONTACT 216
 - UPDATE CONTACTGROUP 217
 - UPDATE DATABASE CONFIGURATION 218
 - UPDATE DATABASE MANAGER CONFIGURATION 222
 - UPDATE HEALTH NOTIFICATION CONTACT LIST 224
 - UPDATE HISTORY 225
 - UPDATE STMM TUNING DBPARTITIONNUM 227
 - recuperación de mensajes 239
- ADMIN_COPY_SCHEMA, procedimiento 921
- ADMIN_DROP_SCHEMA, procedimiento 925
- admin_get_dbp_mem_usage
 - función de tabla 230
- ADMIN_GET_INDEX_COMPRESS_INFO, función de tabla 232
- ADMIN_GET_INDEX_INFO, función de tabla 236
- ADMIN_GET_MSGS, función de tabla 239
- ADMIN_GET_TAB_COMPRESS_INFO_V97, función de tabla 246
- ADMIN_GET_TAB_INFO, función de tabla 993
- ADMIN_GET_TAB_INFO_V97, función de tabla 254
- ADMIN_GET_TEMP_COLUMNS, función de tabla 263
- ADMIN_GET_TEMP_TABLES, función de tabla 266
- ADMIN_IS_INLINED, función 228, 241
- ADMIN_MOVE_TABLE, procedimiento 928
- ADMIN_MOVE_TABLE_UTIL, procedimiento 939
- ADMIN_REMOVE_MSGS, procedimiento 242
- ADMIN_REVALIDATE_DB_OBJECTS, procedimiento 243
- ADMIN_TASK_ADD, procedimiento 271
- ADMIN_TASK_LIST, vista administrativa 276
- ADMIN_TASK_REMOVE, procedimiento 278
- ADMIN_TASK_STATUS, vista administrativa 279
- ADMIN_TASK_UPDATE, procedimiento 281
- ADMINTABCOMPRESSINFO 1001
- ADMINTABINFO, vista administrativa 254
- ADMINTEMPCOLUMNS, vista administrativa 263
- ADMINTEMPTABLES, vista administrativa 266
- alertas de salud
 - configuración de acciones de alertas 1033
 - configuración de alertas 1036
- AM_BASE_RPT_RECOMS, función de tabla 21
- AM_BASE_RPTS, función de tabla 22
- AM_DROP_TASK, procedimiento 23
- AM_GET_LOCK_CHN_TB, procedimiento 24
- AM_GET_LOCK_CHNS, procedimiento 25
- AM_GET_LOCK_RPT, procedimiento 25
- AM_GET_RPT, procedimiento 33
- AM_SAVE_TASK, procedimiento 34
- ANALYZE_LOG_SPACE, procedimiento 835
- API de SQL común 295
 - documentos de entrada XML 297
 - Documentos de mensaje XML 300
 - filtrado de salidas 299
 - modalidad completa 298
 - salida XML 298
 - signatura 295
- APPL_PERFORMANCE, vista administrativa 489
- APPLICATION_ID, función escalar 944
- APPLICATIONS, vista administrativa 490
- archivo histórico
 - recuperación de información 949
- AUDIT_ARCHIVE 285
- AUDIT_DELM_EXTRACT 286
- AUDIT_LIST_LOGS 287
- AUTH_LIST_AUTHORITIES_FOR_AUTHID, función de tabla 477
- AUTH_LIST_GROUPS_FOR_AUTHID, función de tabla 480
- AUTH_LIST_ROLES_FOR_AUTHID 482
- AUTHORIZATIONIDS, vista administrativa 484
- AUTOCONFIGURE, mandato
 - mediante ADMIN_CMD 42
- AUTOMAINT_GET_POLICY, procedimiento almacenado 289
- AUTOMAINT_GET_POLICYFILE, procedimiento almacenado 290
- AUTOMAINT_SET_POLICY, procedimiento almacenado 291
- AUTOMAINT_SET_POLICYFILE, procedimiento almacenado 292
- autorizaciones
 - para vistas administrativas 1
 - recuperación de ID de autorización 484
 - recuperación de pertenencia a grupo 480
- aviso 1201
- ayuda
 - idioma de configuración 1194

ayuda (*continuación*)
sentencias SQL 1193

B

BACKUP DATABASE, mandato
mediante ADMIN_CMD 47
BP_HITRATIO, vista administrativa 494
BP_READ_IO, vista administrativa 497
BP_WRITE_IO, vista administrativa 499

C

CANCEL_WORK, procedimiento almacenado 301
CAPTURE_STORAGEMGMT_INFO, procedimiento 845
Centro de información de DB2
actualización 1194, 1196
idiomas 1194
versiones 1193
visualización en distintos idiomas 1194
COMPILATION_ENV, función de tabla 945
CONTACTGROUPS, vista administrativa 947
contactos
recuperación de grupos de contactos 947
recuperación de listas de contactos 948
CONTACTS, vista administrativa 948
CONTAINER_UTILIZATION, vista administrativa 501
copia
esquemas y objetos 921
CREATE_STORAGEMGMT_TABLES, procedimiento 846

D

DB_HISTORY, vista administrativa
descripción 949
DB_PARTITIONS, función de tabla 341
DBCFG, vista administrativa 342
DBMCFG, vista administrativa 344
DBPATHS, vista administrativa 954
descartar
esquemas y sus objetos 925
DESCRIBE, mandato
descripción
con el procedimiento ADMIN_CMD 55
determinación de problemas
guías de aprendizaje 1198
información disponible 1198
mensajes de anotación cronológica de notificación 975
dividir duplicación
recuperación de vías de acceso de base de datos 954
documentación
copia impresa 1189
PDF 1189
términos y condiciones de uso 1199
visión general 1189
Documentos de salida XML
creación de versiones para la API de SQL común 296
DROP CONTACT, mandato
mediante ADMIN_CMD 69
DROP CONTACTGROUP, mandato
mediante ADMIN_CMD 70
DROP_STORAGEMGMT_TABLES, procedimiento 847

E

ENV_FEATURE_INFO, vista administrativa
descripción 349
ENV_INST_INFO, vista administrativa 350
ENV_PROD_INFO, vista administrativa 352
ENV_SYS_INFO, vista administrativa 353
ENV_SYS_RESOURCES, vista administrativa 354
esquemas
copia de esquemas y objetos 921
descartar objetos 925
EVMON_FORMAT_UE_TO_TABLES, procedimiento
descripción 360
EVMON_FORMAT_UE_TO_XML, función de tabla
descripción 367
EXPLAIN_FORMAT_STATS, función escalar 958
EXPLAIN_GET_MSGS, función de tabla 962
EXPORT, mandato
descripción
con el procedimiento ADMIN_CMD 71
extensiones
estado del traslado 399

F

FORCE APPLICATION, mandato
mediante ADMIN_CMD 83
función de tabla
ADMIN_GET_INDEX_COMPRESS_INFO, función de
tabla 232
ADMIN_GET_INDEX_INFO, función de tabla 236
Función de tabla ADMIN_GET_TAB_COMPRESS_INFO 1001
funciones
ADMIN_IS_INLINED 228, 241
AUDIT_LIST_LOGS 287
escalares
APPLICATION_ID 944
GET_ROUTINE_OPTS 828
MQPUBLISH 457
MQREAD 458
MQREADCLOB 463
MQRECEIVE 464
MQRECEIVECLOB 470
MQSEND 472
MQSUBSCRIBE 473
MQUNSUBSCRIBE 474
SQLERRM 986
funciones de tabla 1
ADMIN_GET_MSGS 239
ADMIN_GET_TAB_COMPRESS_INFO 1001
ADMIN_GET_TAB_COMPRESS_INFO_V97 246
ADMIN_GET_TAB_INFO 993
ADMIN_GET_TAB_INFO_V97 254
ADMIN_GET_TEMP_COLUMNS 263
ADMIN_GET_TEMP_TABLES 266
AM_BASE_RPT_RECOMS 21
AM_BASE_RPTS 22
AUDIT_ARCHIVE 285
AUTH_LIST_AUTHORITIES_FOR_AUTHID 477
AUTH_LIST_GROUPS_FOR_AUTHID 480
comparadas con vistas administrativas 1
COMPILATION_ENV 945
DB_PARTITIONS 341
EVMON_FORMAT_UE_TO_XML 367
EXPLAIN_GET_MSGS 962
GET_DB_CONFIG 1006
GET_DBM_CONFIG 1008

funciones (continuación)

funciones de tabla (continuación)

HEALTH_CONT_HI 1009
 HEALTH_CONT_HI_HIS 1010
 HEALTH_CONT_INFO 1012
 HEALTH_DB_HI 1014
 HEALTH_DB_HI_HIS 1017
 HEALTH_DB_HIC 1021
 HEALTH_DB_HIC_HIS 1023
 HEALTH_DB_INFO 1026
 HEALTH_DBM_HI 1027
 HEALTH_DBM_HI_HIS 1029
 HEALTH_DBM_INFO 1031
 HEALTH_GET_ALERT_ACTION_CFG 1033
 HEALTH_GET_ALERT_CFG 1036
 HEALTH_GET_IND_DEFINITION 1039
 HEALTH_TBS_HI 1043
 HEALTH_TBS_HI_HIS 1046
 HEALTH_TBS_INFO 1049
 MON_GET_ACTIVITY_DETAILS 370
 MON_GET_BUFFERPOOL 379
 MON_GET_CONNECTION 383
 MON_GET_CONNECTION_DETAILS 389
 MON_GET_CONTAINER 396
 MON_GET_EXTENT_MOVEMENT_STATUS 399
 MON_GET_INDEX 400
 MON_GET_PKG_CACHE_STMT 402
 MON_GET_SERVICE_SUBCLASS 408
 MON_GET_SERVICE_SUBCLASS_DETAILS 415
 MON_GET_TABLE 422
 MON_GET_TABLESPACE 425
 MON_GET_UNIT_OF_WORK 429
 MON_GET_UNIT_OF_WORK_DETAILS 435
 MON_GET_WORKLOAD 443
 MON_GET_WORKLOAD_DETAILS 449
 MQREADALL 459
 MQREADALLCLOB 461
 MQRECEIVEALL 466
 MQRECEIVEALLCLOB 468
 obsoletas 991
 PD_GET_DIAG_HIST 968
 PD_GET_LOG_MSGS 975
 SNAP_GET_AGENT 515, 667
 SNAP_GET_AGENT_MEMORY_POOL 518, 670
 SNAP_GET_APPL 1051
 SNAP_GET_APPL_INFO 1059
 SNAP_GET_APPL_INFO_V95 523, 674
 SNAP_GET_APPL_V95 530, 682
 SNAP_GET_BP 1065
 SNAP_GET_BP_PART 546, 698
 SNAP_GET_BP_V95 540, 692
 SNAP_GET_CONTAINER (obsoleta) 1070
 SNAP_GET_CONTAINER_V91 549, 701
 SNAP_GET_DB (obsoleta) 1071
 SNAP_GET_DB_MEMORY_POOL 567, 719
 SNAP_GET_DB_V91 1083
 SNAP_GET_DB_V95 554, 706
 SNAP_GET_DBM 1079
 SNAP_GET_DBM_MEMORY_POOL 576, 728
 SNAP_GET_DBM_V95 572, 724
 SNAP_GET_DETAIL_LOG_V91 580, 732
 SNAP_GET_DYN_SQL (obsoleta) 1099
 SNAP_GET_DYN_SQL_V91 1095
 SNAP_GET_DYN_SQL_V95 583, 735
 SNAP_GET_FCM 589, 741
 SNAP_GET_FCM_PART 591, 743
 SNAP_GET_HADR 594, 746

funciones (continuación)

funciones de tabla (continuación)

SNAP_GET_LOCK 599, 751
 SNAP_GET_LOCKWAIT 604, 756
 SNAP_GET_STMT 611, 763
 SNAP_GET_STO_PATHS (obsoleta) 1101
 SNAP_GET_STORAGE_PATHS 770
 SNAP_GET_STORAGE_PATHS_V97 618
 SNAP_GET_SUBSECTION 621, 773
 SNAP_GET_SWITCHES 625, 777
 SNAP_GET_TAB (obsoleta) 1102
 SNAP_GET_TAB_REORG 632, 784
 SNAP_GET_TAB_V91 628, 781
 SNAP_GET_TBSP (obsoleta) 1103
 SNAP_GET_TBSP_PART (obsoleta) 1107
 SNAP_GET_TBSP_PART_V91 797
 SNAP_GET_TBSP_PART_V97 645
 SNAP_GET_TBSP QUIESCER 650, 803
 SNAP_GET_TBSP_RANGE 654, 807
 SNAP_GET_TBSP_V91 639, 791
 SNAP_GET_UTIL 658, 811
 SNAP_GET_UTIL_PROGRESS 662, 815
 SNAPSHOT_AGENT (obsoleta) 1109
 SNAPSHOT_APPL (obsoleta) 1110
 SNAPSHOT_APPL_INFO (obsoleta) 1116
 SNAPSHOT_BP (obsoleta) 1118
 SNAPSHOT_CONTAINER (obsoleta) 1121
 SNAPSHOT_DATABASE (obsoleta) 1122
 SNAPSHOT_DBM (obsoleta) 1129
 SNAPSHOT_DYN_SQL (obsoleta) 1131
 SNAPSHOT_FCM (obsoleta) 1133
 SNAPSHOT_FCMNODE (obsoleta) 1134
 SNAPSHOT_LOCK (obsoleta) 1136
 SNAPSHOT_LOCKWAIT (obsoleta) 1137
 SNAPSHOT QUIESCERS (obsoleta) 1138
 SNAPSHOT_RANGES (obsoleta) 1140
 SNAPSHOT_STATEMENT (obsoleta) 1141
 SNAPSHOT_SUBSECT (obsoleta) 1143
 SNAPSHOT_SWITCHES (obsoleta) 1145
 SNAPSHOT_TABLE (obsoleta) 1147
 SNAPSHOT_TBREORG (obsoleta) 1148
 SNAPSHOT_TBS (obsoleta) 1149
 SNAPSHOT_TBS_CFG (obsoleta) 1152
 soportadas 3
 SQLCACHE_SNAPSHOT (obsoleta) 1154
 WLM_GET_ACTIVITY_DETAILS 1156
 WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES 1168
 funciones escalares
 EXPLAIN_FORMAT_STATS 958
 procedimientos almacenados
 SYSTS ALTER 850
 SYSTS_CLEAR_COMMANDLOCKS 854
 SYSTS_CLEAR_EVENTS 856
 SYSTS_DROP 868
 SYSTS_ENABLE 870
 SYSTS_UPDATE 871
 soportadas 3
 tabla
 WLM_GET_QUEUE_STATS 879
 WLM_GET_SERVICE_CLASS_AGENTS 1163
 WLM_GET_SERVICE_CLASS_AGENTS_V97 883
 WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES_V97 890
 WLM_GET_SERVICE_SUBCLASS_STATS 1173
 WLM_GET_SERVICE_SUBCLASS_STATS_V97 895
 WLM_GET_SERVICE_SUPERCLASS_STATS 904

funciones (continuación)

tabla (continuación)

WLM_GET_WORK_ACTION_SET_STATS 906
WLM_GET_WORKLOAD_OCCURRENCE
_ACTIVITIES 1180
WLM_GET_WORKLOAD_OCCURRENCE
_ACTIVITIES_V97 908
WLM_GET_WORKLOAD_STATS 1185
WLM_GET_WORKLOAD_STATS_V97 913

funciones de tabla

admin_get_dbp_mem_usage 230
ADMIN_GET_MSGS 239
ADMIN_GET_TAB_COMPRESS_INFO_V97 246
ADMIN_GET_TAB_INFO_V97 254
ADMIN_GET_TEMP_COLUMNS 263
ADMIN_GET_TEMP_TABLES 266
AUDIT_ARCHIVE 285
AUTH_LIST_GROUPS_FOR_AUTHID 480
comparadas con vistas administrativas 1
HEALTH_GET_ALERT_ACTION_CFG 1033
HEALTH_GET_ALERT_CFG 1036
HEALTH_GET_IND_DEFINITION 1039
obsoleta, funcionalidad 991
ADMIN_GET_TAB_INFO 993
SNAP_GET_APPL_INFO 1059
SNAP_GET_BP 1065
SNAP_GET_DB_V91 1083
SNAP_GET_DBM 1079
SNAP_GET_DYN_SQL_V91 1095
PD_GET_DIAG_HIST 968
PD_GET_LOG_MSGS 975
SNAP_GET_AGENT 515, 667
SNAP_GET_AGENT_MEMORY_POOL 518, 670
SNAP_GET_APPL_INFO_V95 523, 674
SNAP_GET_APPL_V95 530, 682
SNAP_GET_BP_PART 546, 698
SNAP_GET_BP_V95 540, 692
SNAP_GET_CONTAINER_V91 549, 701
SNAP_GET_DB_MEMORY_POOL 567, 719
SNAP_GET_DB_V95 554, 706
SNAP_GET_DBM_MEMORY_POOL 576, 728
SNAP_GET_DBM_V95 572, 724
SNAP_GET_DETAIL_LOG_V91 580, 732
SNAP_GET_DYN_SQL_V95 583, 735
SNAP_GET_FCM 589, 741
SNAP_GET_FCM_PART 591, 743
SNAP_GET_HADR 594, 746
SNAP_GET_LOCK 599, 751
SNAP_GET_LOCKWAIT 604, 756
SNAP_GET_STMT 611, 763
SNAP_GET_STORAGE_PATHS 770
SNAP_GET_STORAGE_PATHS_V97 618
SNAP_GET_SUBSECTION 621, 773
SNAP_GET_SWITCHES 625, 777
SNAP_GET_TAB_REORG 632, 784
SNAP_GET_TAB_V91 628, 781
SNAP_GET_TBSP_PART_V91 797
SNAP_GET_TBSP_PART_V97 645
SNAP_GET_TBSP QUIESCER 650, 803
SNAP_GET_TBSP_RANGE 654, 807
SNAP_GET_TBSP_V91 639, 791
SNAP_GET_UTIL 658, 811
SNAP_GET_UTIL_PROGRESS 662, 815
soportadas 3
funciones escalares
SQLERRM 986

G

GENERATE_DISTFILE, procedimiento 837
GET_CONFIG, procedimiento almacenado 308
GET_DB_CONFIG, función de tabla 1006
GET_DBM_CONFIG, función de tabla 1008
GET_DBSIZE_INFO, procedimiento 965
GET_MESSAGE, procedimiento almacenado 315
GET_ROUTINE_OPTS, función escalar 828
GET_ROUTINE_SAR, procedimiento 828
GET STMM TUNING DBPARTITIONNUM, mandato
mediante ADMIN_CMD 85
GET_SWRD_SETTINGS, procedimiento 838
GET_SYSTEM_INFO, procedimiento almacenado 323
grupos
recuperación de pertenencia a grupo 480
guías de aprendizaje
determinación de problemas 1198
resolución de problemas 1198
Visual Explain 1198

H

HEALTH_CONT_HI, función de tabla 1009
HEALTH_CONT_HI_HIS, función de tabla 1010
HEALTH_CONT_INFO, función de tabla 1012
HEALTH_DB_HI, función de tabla 1014
HEALTH_DB_HI_HIS, función de tabla 1017
HEALTH_DB_HIC, función de tabla 1021
HEALTH_DB_HIC_HIS, función de tabla 1023
HEALTH_DB_INFO, función de tabla 1026
HEALTH_DBM_HI, función de tabla 1027
HEALTH_DBM_HI_HIS, función de tabla 1029
HEALTH_DBM_INFO, función de tabla 1031
HEALTH_GET_ALERT_ACTION_CFG, función de
tabla 1033
HEALTH_GET_ALERT_CFG, función de tabla 1036
HEALTH_GET_IND_DEFINITION, función de tabla 1039
HEALTH_HI_REC, procedimiento 1041
HEALTH_TBS_HI, función de tabla 1043
HEALTH_TBS_HI_HIS, función de tabla 1046
HEALTH_TBS_INFO, función de tabla 1049
herramienta de gestión de almacenamiento
procedimientos almacenados 845, 846, 847

I

IMPORT, mandato
mediante ADMIN_CMD 86
indicadores de salud
recuperar definición de indicador de salud 1039
información del sistema
obtener 323
recuperación 353
INITIALIZE TAPE, mandato
mediante ADMIN_CMD 114
instalación
devolución de información de licencia del producto
DB2 349, 354
recuperación de información del producto DB2 352
instancias
recuperación de información de instancia actual 350

L

- listas de contactos
 - recuperación de contactos 948
 - recuperación de listas de grupos de contactos 947
- listas de notificaciones
 - recuperación de lista de contactos 967
- LOAD, mandato
 - mediante ADMIN_CMD 115
- LOCKS_HELD, vista administrativa 503
- LOCKWAIT, vista administrativa 506
- LOG_UTILIZATION, vista administrativa 510

M

- mandato QUIESCE DATABASE
 - mediante ADMIN_CMD 160
- mandato RESET DATABASE CONFIGURATION
 - mediante ADMIN_CMD 192
- mandato UPDATE STMM TUNING DBPARTITIONNUM
 - mediante ADMIN_CMD 227
- mandatos
 - ADD CONTACT 39
 - ADD CONTACTGROUP 41
 - AUTOCONFIGURE 42
 - BACKUP DATABASE 47
 - DESCRIBE 55
 - DROP CONTACT 69
 - DROP CONTACTGROUP 70
 - EXPORT 71
 - FORCE APPLICATION 83
 - GET STMM TUNING DBPARTITIONNUM 85
 - IMPORT 86
 - INITIALIZE TAPE 114
 - invocar desde procedimiento 37, 849
 - LOAD 115
 - PRUNE HISTORY/LOGFILE 158
 - QUIESCE DATABASE 160
 - QUIESCE TABLESPACES FOR TABLE 162
 - REDISTRIBUTE DATABASE PARTITION GROUP 165
 - REORG INDEXES/TABLE 176
 - RESET ALERT CONFIGURATION 190
 - RESET DATABASE CONFIGURATION 192
 - RESET DATABASE MANAGER CONFIGURATION 194
 - REWIND TAPE 195
 - RUNSTATS 196
 - SET TAPE POSITION 208
 - UNQUIESCE DATABASE 209
 - UPDATE ALERT CONFIGURATION 210
 - UPDATE CONTACT 216
 - UPDATE CONTACTGROUP 217
 - UPDATE DATABASE CONFIGURATION 218
 - UPDATE DATABASE MANAGER CONFIGURATION 222
 - UPDATE HEALTH NOTIFICATION CONTACT LIST 224
 - UPDATE HISTORY 225
 - UPDATE STMM TUNING DBPARTITIONNUM 227
- manuales
 - copia impresa
 - pedido 1192
- mensajes de anotación cronológica de notificación
 - recuperación 975
- mensajes de error
 - recuperación
 - SQLERRM, funciones escalares 986
- modalidad completa
 - API de SQL común 298

- MON_GET_ACTIVITY_DETAILS, función de tabla
 - descripción 370
- MON_GET_BUFFERPOOL, función de tabla
 - descripción 379
- MON_GET_CONNECTION, función de tabla
 - descripción 383
- MON_GET_CONNECTION_DETAILS, función de tabla
 - descripción 389
- MON_GET_CONTAINER, función de tabla
 - descripción 396
- MON_GET_EXTENT_MOVEMENT_STATUS
 - descripción 399
- MON_GET_INDEX, función de tabla
 - descripción 400
- MON_GET_PKG_CACHE_STMT, función de tabla
 - descripción 402
- MON_GET_SERVICE_SUBCLASS, función de tabla
 - descripción 408
- MON_GET_SERVICE_SUBCLASS_DETAILS, función de tabla
 - descripción 415
- MON_GET_TABLE, función de tabla
 - descripción 422
- MON_GET_TABLESPACE, función de tabla
 - descripción 425
- MON_GET_UNIT_OF_WORK, función de tabla
 - descripción 429
- MON_GET_UNIT_OF_WORK_DETAILS, función de tabla
 - descripción 435
- MON_GET_WORKLOAD, función de tabla
 - descripción 443
- MON_GET_WORKLOAD_DETAILS, función de tabla
 - descripción 449
- MQPUBLISH, función escalar 457
- MQREAD, función escalar 458
- MQREADALL, función de tabla 459
- MQREADALLCLOB, función de tabla 461
- MQREADCLOB, función escalar 463
- MQRECEIVE, función escalar 464
- MQRECEIVEALL, función de tabla 466
- MQRECEIVEALLCLOB, función de tabla 468
- MQRECEIVECLOB, función escalar 470
- MQSEND, función escalar 472
- MQSUBSCRIBE, función escalar 473
- MQUNSUBSCRIBE, función escalar 474

N

- NOTIFICATIONLIST, vista administrativa 967

O

- OBJECTOWNERS, vista administrativa 485
- objetos
 - recuperación de propiedad de objeto 485
- obsoleta, funcionalidad
 - funciones de tabla
 - GET_DBM_CONFIG 1008
 - SNAP_GET_APP 1051
 - SNAP_GET_BP 1065
 - SNAP_GET_CONTAINER 1070
 - SNAP_GET_DB 1071
 - SNAP_GET_DB_V91 1083
 - SNAP_GET_DBM 1079
 - SNAP_GET_DYN_SQL 1099
 - SNAP_GET_STO_PATHS 1101
 - SNAP_GET_TAB 1102

obsoleta, funcionalidad (continuación)

funciones de tabla (continuación)

SNAP_GET_TBSP 1103
SNAP_GET_TBSP_PART 1107
SNAP_GET_TBSP_PART_V91 797
SNAPSHOT_AGENT 1109
SNAPSHOT_APPL 1110
SNAPSHOT_APPL_INFO 1116
SNAPSHOT_BP 1118
SNAPSHOT_CONTAINER 1121
SNAPSHOT_DATABASE 1122
SNAPSHOT_DBM 1129
SNAPSHOT_DYN_SQL 1131
SNAPSHOT_FCM 1133
SNAPSHOT_FCMNODE 1134
SNAPSHOT_LOCK 1136
SNAPSHOT_LOCKWAIT 1137
SNAPSHOT QUIESCERS 1138
SNAPSHOT_RANGES 1140
SNAPSHOT_STATEMENT 1141
SNAPSHOT_SUBSECT 1143
SNAPSHOT_SWITCHES 1145
SNAPSHOT_TABLE 1147
SNAPSHOT_TBREORG 1148
SNAPSHOT_TBS 1149
SNAPSHOT_TBS_CFG 1152
SNAPSTORAGE_PATHS 770
SQLCACHE_SNAPSHOT 1154

procedimientos

GET_DB_CONFIG 1006
HEALTH_CONT_HI 1009
HEALTH_CONT_HI_HIS 1010
HEALTH_CONT_INFO 1012
HEALTH_DB_HI 1014
HEALTH_DB_HI_HIS 1017
HEALTH_DB_HIC 1021
HEALTH_DB_HIC_HIS 1023
HEALTH_DB_INFO 1026
HEALTH_DBM_HI 1027
HEALTH_DBM_HI_HIS 1029
HEALTH_DBM_INFO 1031
HEALTH_GET_ALERT_ACTION_CFG 1033
HEALTH_GET_ALERT_CFG 1036
HEALTH_GET_IND_DEFINITION 1039
HEALTH_HI_REC 1041
HEALTH_TBS_HI 1043
HEALTH_TBS_HI_HIS 1046
HEALTH_TBS_INFO 1049
SNAPSHOT_FILEW 1135
SYSINSTALLROUTINES 1156
WLM_GET_SERVICE_SUBCLASS_STATS 1173
WLM_GET_WORKLOAD_STATS 1185

rutinas administrativas de SQL 991

P

paquetes

volver a vincular 831

parámetros de configuración

determinación mediante SET_CONFIG 329

obtener 308

parámetros de configuración de base de datos

recuperación 342

parámetros de configuración de gestor de bases de datos

recuperación de valores 344

PD_GET_DIAG_HIST, función de tabla

descripción 968

PD_GET_LOG_MSGS, función de tabla

descripción 975

PDLOGMSG_LAST24HOURS, vista administrativa

descripción 975

pedido de manuales de DB2 1192

planificador de tareas administrativas

definición de planificaciones de tareas 275

PRIVILEGES, vista administrativa 486

privilegios

información sobre otorgados

PRIVILEGES, vista administrativa 486

Procedimiento ALTER_ROUTINE_PACKAGE 827

Procedimiento ALTOBJ 941

Procedimiento SNAP_WRITE_FILE 666, 819

procedimientos

ADMIN_CMD

descripción 37

ADMIN_COPY_SCHEMA 921

ADMIN_DROP_SCHEMA 925

ADMIN_MOVE_TABLE 928

ADMIN_MOVE_TABLE_UTIL 939

ADMIN_REMOVE_MSGS 242

ADMIN_REVALIDATE_DB_OBJECTS 243

ADMIN_TASK_ADD 271

ADMIN_TASK_REMOVE 278

ADMIN_TASK_UPDATE 281

ALTER_ROUTINE_PACKAGE 827

ALTOBJ 941

AM_DROP_TASK 23

AM_GET_LOCK_CHN_TB 24

AM_GET_LOCK_CHNS 25

AM_GET_LOCK_RPT 25

AM_GET_RPT 33

AM_SAVE_TASK 34

ANALYZE_LOG_SPACE 835

API de SQL común 295, 299

AUDIT_ARCHIVE 285

AUDIT_DELIM_EXTRACT 286

AUTOMAINT_GET_POLICY 289

AUTOMAINT_GET_POLICYFILE 290

AUTOMAINT_SET_POLICYFILE 292

CANCEL_WORK 301

CAPTURE_STORAGE_MGMT_INFO 845

CREATE_STORAGE_MGMT_TABLES 846

DROP_STORAGE_MGMT_TABLES 847

EVMON_FORMAT_UE_TO_TABLES 360

GENERATE_DISTFILE 837

GET_CONFIG 308

GET_DBSIZE_INFO 965

GET_MESSAGE 315

GET_ROUTINE_SAR 828

GET_SWRD_SETTINGS 838

GET_SYSTEM_INFO 323

HEALTH_HI_REC 1041

obsoleta, funcionalidad 991

PUT_ROUTINE_SAR 829

REBIND_ROUTINE_PACKAGE 831

REORGCHK_IX_STATS 982

REORGCHK_TB_STATS 984

resumen 3

SET_CONFIG 329

SET_ROUTINE_OPTS 833

SET_SWRD_SETTINGS 841

SNAP_WRITE_FILE 666, 819

SNAPSHOT_FILEW 1135

STEPWISE_REDISTRIBUTE_DBPG 843

SYSINSTALLOBJECTS 989

procedimientos (*continuación*)
 SYSINSTALLROUTINES 1156
 SYSTS_ADMIN_CMD 849
 WLM_CANCEL_ACTIVITY 875
 WLM_CAPTURE_ACTIVITY_IN_PROGRESS 876
 WLM_COLLECT_STATS 878
 WLM_SET_CLIENT_INFO 918
 procedimientos almacenados
 AUDIT_ARCHIVE 285
 AUDIT_DELIM_EXTRACT 286
 procedimientos almacenados de la API de SQL común
 creación de versiones 296
 PRUNE HISTORY/LOGFILE, mandato
 con el procedimiento ADMIN_CMD 158
 PUT_ROUTINE_SAR, procedimiento 829

Q

QUERY_PREP_COST, vista administrativa 514
 QUIESCE TABLESPACES FOR TABLE, mandato
 mediante ADMIN_CMD 162

R

REBIND_ROUTINE_PACKAGE, procedimiento 831
 redistribución de datos
 procedimientos 835, 837, 838, 841, 843
 REDISTRIBUTE DATABASE PARTITION GROUP, mandato
 mediante ADMIN_CMD 165
 REG_VARIABLES, vista administrativa 347
 REORG INDEXES/TABLE, mandato
 mediante ADMIN_CMD 176
 REORGCHK_IX_STATS, procedimiento 982
 REORGCHK_TB_STATS, procedimiento 984
 RESET ALERT CONFIGURATION, mandato
 mediante ADMIN_CMD 190
 RESET DATABASE MANAGER CONFIGURATION, mandato
 mediante ADMIN_CMD 194
 resolución de problemas
 guías de aprendizaje 1198
 información en línea 1198
 revalidación de datos
 procedimientos 243
 REWIND TAPE, mandato
 mediante ADMIN_CMD 195
 RUNSTATS, mandato
 mediante ADMIN_CMD 196
 rutina administrativa
 Procedimiento AUTOMAINT_SET_POLICY 291
 rutinas
 administrativas de SQL 991
 soportadas 3
 supervisor 359
 rutinas administrativas
 AUTH_LIST_ROLES_FOR_AUTHID 482
 rutinas administrativas de SQL 858
 rutinas obsoletas 991

S

sentencias SQL
 visualización de la ayuda 1193
 SET_CONFIG, procedimiento almacenado 329
 SET_ROUTINE_OPTS, procedimiento 833
 SET_SWRD_SETTINGS, procedimiento 841

SET TAPE POSITION, mandato
 mediante ADMIN_CMD 208
 SNAP_GET_AGENT, función de tabla 515, 667
 SNAP_GET_AGENT_MEMORY_POOL, función de
 tabla 518, 670
 SNAP_GET_APPL_INFO, función de tabla 1059
 SNAP_GET_APPL_INFO_V95, función de tabla 523, 674
 SNAP_GET_APPL_V95, función de tabla 530, 682
 SNAP_GET_BP_PART, función de tabla 546, 698
 SNAP_GET_BP_V95, función de tabla 540, 692
 SNAP_GET_CONTAINER, función de tabla obsoleta 1070
 SNAP_GET_CONTAINER_V91, función de tabla 549, 701
 SNAP_GET_DB, función de tabla obsoleta 1071
 SNAP_GET_DB_MEMORY_POOL, función de tabla 567, 719
 SNAP_GET_DB_V95, función de tabla 554, 706
 SNAP_GET_DBM_MEMORY_POOL, función de tabla 576,
 728
 SNAP_GET_DBM_V95, función de tabla 572, 724
 SNAP_GET_DETAIL_LOG_V91, función de tabla 580, 732
 SNAP_GET_DYN_SQL, función de tabla obsoleta 1099
 SNAP_GET_DYN_SQL_V91, función de tabla 1095
 SNAP_GET_DYN_SQL_V95, función de tabla 583, 735
 SNAP_GET_FCM, función de tabla 589, 741
 SNAP_GET_FCM_PART, función de tabla 591, 743
 SNAP_GET_HADR, función de tabla 594, 746
 SNAP_GET_LOCK, función de tabla 599, 751
 SNAP_GET_LOCKWAIT, función de tabla 604, 756
 SNAP_GET_STMT, función de tabla 611, 763
 SNAP_GET_STO_PATHS, función de tabla obsoleta 1101
 SNAP_GET_STORAGE_PATHS, función de tabla 770
 SNAP_GET_STORAGE_PATHS_V97, función de tabla 618
 SNAP_GET_SUBSECTION, función de tabla 621, 773
 SNAP_GET_SWITCHES, función de tabla 625, 777
 SNAP_GET_TAB, función de tabla obsoleta 1102
 SNAP_GET_TAB_REORG, función de tabla 632, 784
 SNAP_GET_TAB_V91, función de tabla 628, 781
 SNAP_GET_TBSP, función de tabla obsoleta 1103
 SNAP_GET_TBSP_PART, función de tabla obsoleta 1107
 SNAP_GET_TBSP_PART_V91, función de tabla 797
 SNAP_GET_TBSP_PART_V97, función de tabla 645
 SNAP_GET_TBSP QUIESCER, función de tabla 650, 803
 SNAP_GET_TBSP_RANGE, función de tabla 654, 807
 SNAP_GET_TBSP_V91, función de tabla 639, 791
 SNAP_GET_UTIL, función de tabla 658, 811
 SNAP_GET_UTIL_PROGRESS, función de tabla 662, 815
 SNAPAGENT, vista administrativa 515, 667
 SNAPAGENT_MEMORY_POOL, vista administrativa 518,
 670
 SNAPAPPL, vista administrativa 530, 682
 SNAPAPPL_INFO, vista administrativa 523, 674
 SNAPBP, vista administrativa 540, 692
 SNAPBP_PART, vista administrativa 546, 698
 SNAPCONTAINER, vista administrativa 549, 701
 SNAPDB, vista administrativa 554, 706
 SNAPDB_MEMORY_POOL, vista administrativa 567, 719
 SNAPDBM, vista administrativa 572, 724
 SNAPDBM_MEMORY_POOL, vista administrativa 576, 728
 SNAPDETAILLOG, vista administrativa 580, 732
 SNAPDYN_SQL, vista administrativa 583, 735
 SNAPFCM, vista administrativa 589, 741
 SNAPFCM_PART, vista administrativa 591, 743
 SNAPHADR, vista administrativa 594, 746
 SNAPLOCK, vista administrativa 599, 751
 SNAPLOCKWAIT, vista administrativa 604, 756
 SNAPSHOT_AGENT, función de tabla obsoleta 1109
 SNAPSHOT_APPL, función de tabla obsoleta 1110
 SNAPSHOT_APPL_INFO, función de tabla obsoleta 1116

SNAPSHOT_BP, función de tabla obsoleta 1118
 SNAPSHOT_CONTAINER, función de tabla obsoleta 1121
 SNAPSHOT_DATABASE, función de tabla obsoleta 1122
 SNAPSHOT_DBM, función de tabla obsoleta 1129
 SNAPSHOT_DYN_SQL, función de tabla obsoleta 1131
 SNAPSHOT_FCM, función de tabla obsoleta 1133
 SNAPSHOT_FCMNODE, función de tabla obsoleta 1134
 SNAPSHOT_FILEW, procedimiento obsoleto 1135
 SNAPSHOT_LOCK, función de tabla obsoleta 1136
 SNAPSHOT_LOCKWAIT, función de tabla obsoleta 1137
 SNAPSHOT QUIESCERS, función de tabla obsoleta 1138
 SNAPSHOT_RANGES, función de tabla obsoleta 1140
 SNAPSHOT_STATEMENT, función de tabla obsoleta 1141
 SNAPSHOT_SUBSECT, función de tabla obsoleta 1143
 SNAPSHOT_SWITCHES, función de tabla obsoleta 1145
 SNAPSHOT_TABLE, función de tabla obsoleta 1147
 SNAPSHOT_TBREORG, función de tabla obsoleta 1148
 SNAPSHOT_TBS, función de tabla obsoleta 1149
 SNAPSHOT_TBS_CFG, función de tabla obsoleta 1152
 SNAPSTMT, vista administrativa 611, 763
 SNAPSTORAGE_PATHS, vista administrativa 618, 770
 SNAPSUBSECTION, vista administrativa 621, 773
 SNAPSWITCHES, vista administrativa 625, 777
 SNAPTAB, vista administrativa 628, 781
 SNAPTAB_REORG, vista administrativa 632, 784
 SNAPTbsp, vista administrativa 639, 791
 SNAPTbsp_QUIESCER, vista administrativa 650, 803
 SNAPTbsp_RANGE, vista administrativa 654, 807
 SNAPTbspPART, vista administrativa 645, 797
 SNAPUTIL, vista administrativa 658, 811
 SNAPUTIL_PROGRESS, vista administrativa 662, 815
 soportadas, funciones 3
 SQL, rutinas administrativas
 soportadas 3
 SQLCACHE_SNAPSHOT, función de tabla obsoleta 1154
 SQLCODE
 devolución de información de mensaje 315
 SQLERRM, función escalar 986
 STEPWISE_REDISTRIBUTE_DBPG, procedimiento 843
 supervisor
 rutinas 359
 SYSINSTALLOBJECTS, procedimiento 989
 SYSINSTALLROUTINES, procedimiento obsoleto 1156
 SYSTS_ADMIN_CMD, procedimiento 849
 SYSTS ALTER, procedimiento almacenado 850
 SYSTS_CLEAR_COMMANDLOCKS, procedimiento
 almacenado 854
 SYSTS_CLEAR_EVENTS, procedimiento almacenado 856
 SYSTS_CREATE 858
 SYSTS_CREATE, procedimiento 858
 SYSTS_DISABLE 865
 SYSTS_DISABLE, procedimiento 865
 SYSTS_DROP, procedimiento almacenado 868
 SYSTS_ENABLE, procedimiento almacenado 870
 SYSTS_UPDATE, procedimiento almacenado 871

T

tablas

 información de compresión 246
 recuperación de tamaño y estado 254, 993
 recuperar información de columna para tablas
 temporales 263
 recuperar información de tablas temporales 266
 traslado en línea 928, 939
 Tbsp_UTILIZATION, vista administrativa 820

términos y condiciones

 uso de publicaciones 1199
 TOP_DYNAMIC_SQL, vista administrativa 824
 traslado de tablas en línea 928, 939

U

UNQUIESCE DATABASE, mandato
 mediante ADMIN_CMD 209
 UPDATE ALERT CONFIGURATION, mandato
 mediante ADMIN_CMD 210
 UPDATE CONTACT, mandato
 mediante ADMIN_CMD 216
 UPDATE CONTACTGROUP, mandato
 mediante ADMIN_CMD 217
 UPDATE DATABASE CONFIGURATION, mandato
 mediante ADMIN_CMD 218
 UPDATE DATABASE MANAGER CONFIGURATION,
 mandato
 mediante ADMIN_CMD 222
 UPDATE HEALTH NOTIFICATION CONTACT LIST,
 mandato
 mediante ADMIN_CMD 224
 UPDATE HISTORY, mandato
 mediante ADMIN_CMD 225

V

variables de registro
 recuperación de valores en uso 347
 vías de acceso de base de datos
 recuperación 954
 Vista administrativa ADMINTABCOMPRESSINFO 246
 Vista administrativa LONG_RUNNING_SQL 511
 vistas
 vistas administrativas
 ADMIN_TASK_LIST 276
 ADMIN_TASK_STATUS 279
 ADMINTABCOMPRESSINFO 246, 1001
 ADMINTABINFO 254
 ADMINTEMPCOLUMNS 263
 ADMINTEMPTABLES 266
 APPL_PERFORMANCE 489
 APPLICATIONS 490
 AUTHORIZATIONIDS 484
 BP_HITRATIO 494
 BP_READ_IO 497
 BP_WRITE_IO 499
 CONTACTGROUPS 947
 CONTACTS 948
 CONTAINER_UTILIZATION 501
 DB_HISTORY 949
 DBCFCG 342
 DBMCFG 344
 DBPATHS 954
 ENV_FEATURE_INFO 349
 ENV_INST_INFO 350
 ENV_PROD_INFO 352
 ENV_SYS_INFO 353
 ENV_SYS_RESOURCES 354
 LOCKS_HELD 503
 LOCKWAIT 506
 LOG_UTILIZATION 510
 LONG_RUNNING_SQL 511
 NOTIFICATIONLIST 967
 OBJECTOWNERS 485

vistas (continuación)

vistas administrativas (continuación)

PDLOGMSG_LAST24HOURS 975
PRIVILEGES 486
QUERY_PREP_COST 514
REG_VARIABLES 347
SNAPAGENT 515, 667
SNAPAGENT_MEMORY_POOL 518, 670
SNAPAPPL 530, 682
SNAPAPPL_INFO 523, 674
SNAPBP 540, 692
SNAPBP_PART 546, 698
SNAPCONTAINER 549, 701
SNAPDB 554, 706
SNAPDB_MEMORY_POOL 567, 719
SNAPDBM 572, 724
SNAPDBM_MEMORY_POOL 576, 728
SNAPDETAILLOG 580, 732
SNAPDYN_SQL 583, 735
SNAPFCM 589, 741
SNAPFCM_PART 591, 743
SNAPHADR 594, 746
SNAPLOCK 599, 751
SNAPLOCKWAIT 604, 756
SNAPSTMT 611, 763
SNAPSTORAGE_PATHS 618, 770
SNAPSUBSECTION 621, 773
SNAPSWITCHES 625, 777
SNAPTAB 628, 781
SNAPTAB_REORG 632, 784
SNAPTbsp 639, 791
SNAPTbsp_QUIESCER 650, 803
SNAPTbsp_RANGE 654, 807
SNAPTbspPART 645, 797
SNAPUTIL 658, 811
SNAPUTIL_PROGRESS 662, 815
TBSP_UTILIZATION 820
TOP_DYNAMIC_SQL 824

vistas administrativas

ADMIN_TASK_LIST 276
ADMIN_TASK_STATUS 279
ADMINTABCOMPRESSINFO 246
ADMINTABINFO 254
ADMINTEMPCOLUMNS 263
ADMINTEMPTABLES 266
APPL_PERFORMANCE 489
APPLICATIONS 490
AUTHORIZATIONIDS 484
autorización 1
BP_HITRATIO 494
BP_READ_IO 497
BP_WRITE_IO 499
comparadas con funciones de tablas 1
CONTACTGROUPS 947
CONTACTS 948
CONTAINER_UTILIZATION 501
DB_HISTORY 949
DBCFCG 342
DBMCFG 344
DBPATHS 954
ENV_FEATURE_INFO 349
ENV_INST_INFO 350
ENV_PROD_INFO 352
ENV_SYS_INFO 353
ENV_SYS_RESOURCES 354
LOCKS_HELD 503
LOCKWAIT 506

vistas administrativas (continuación)

LOG_UTILIZATION 510
LONG_RUNNING_SQL 511
NOTIFICATIONLIST 967
OBJECTOWNERS 485
PDLOGMSG_LAST24HOURS 975
PRIVILEGES 486
QUERY_PREP_COST 514
REG_VARIABLES 347
SNAPAGENT 515, 667
SNAPAGENT_MEMORY_POOL 518, 670
SNAPAPPL 530, 682
SNAPAPPL_INFO 523, 674
SNAPBP 540, 692
SNAPBP_PART 546, 698
SNAPCONTAINER 549, 701
SNAPDB 554, 706
SNAPDB_MEMORY_POOL 567, 719
SNAPDBM 572, 724
SNAPDBM_MEMORY_POOL 576, 728
SNAPDETAILLOG 580, 732
SNAPDYN_SQL 583, 735
SNAPFCM 589, 741
SNAPFCM_PART 591, 743
SNAPHADR 594, 746
SNAPLOCK 599, 751
SNAPLOCKWAIT 604, 756
SNAPSTMT 611, 763
SNAPSTORAGE_PATHS 618, 770
SNAPSUBSECTION 621, 773
SNAPSWITCHES 625, 777
SNAPTAB 628, 781
SNAPTAB_REORG 632, 784
SNAPTbsp 639, 791
SNAPTbsp_QUIESCER 650, 803
SNAPTbsp_RANGE 654, 807
SNAPTbspPART 645, 797
SNAPUTIL 658, 811
SNAPUTIL_PROGRESS 662, 815
soportadas 3
TBSP_UTILIZATION 820
TOP_DYNAMIC_SQL 824
visión general 1
Visual Explain
guía de aprendizaje 1198

W

WLM_CANCEL_ACTIVITY, procedimiento 875
WLM_CAPTURE_ACTIVITY_IN_PROGRESS,
procedimiento 876
WLM_COLLECT_STATS, procedimiento
descripción 878
WLM_GET_ACTIVITY_DETAILS, función de tabla
descripción 1156
WLM_GET_QUEUE_STATS, función de tabla 879
WLM_GET_SERVICE_CLASS_AGENTS, función de
tabla 1163
WLM_GET_SERVICE_CLASS_AGENTS_V97, función de tabla
descripción 883
WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES,
función de tabla
descripción 1168
WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES_V97,
función de tabla
descripción 890

WLM_GET_SERVICE_SUBCLASS_STATS, función de tabla
descripción 1173
WLM_GET_SERVICE_SUBCLASS_STATS_V97, función de
tabla
descripción 895
WLM_GET_SERVICE_SUPERCLASS_STATS, función de
tabla 904
WLM_GET_WORK_ACTION_SET_STATS, función de tabla
descripción 906
WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES,
función de tabla 1180
WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES_V97,
función de tabla
descripción 908
WLM_GET_WORKLOAD_STATS, función de tabla
descripción 1185
WLM_GET_WORKLOAD_STATS_V97, función de tabla
descripción 913
WLM_SET_CLIENT_INFO, procedimiento 918

X

XML
entrada válida para la API de SQL común 298



SC11-3909-00



Spine information:

IBM DB2 9.7 para Linux, UNIX y Windows

Rutinas y vistas administrativas

