



**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 apartado Apéndice B, "Avisos", en la página 873.

**Nota de edición**

Esta publicación es la traducción del original inglés: DB2 Version 9.5 for Linux, UNIX, and Windows - Administrative Routines and Views, (SC23-5843-00).

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](http://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](http://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 1993, 2007. Reservados todos los derechos.

# Contenido

## Parte 1. Vistas y rutinas administrativas SQL . . . . . 1

### Capítulo 1. Autorización para vistas administrativas. . . . . 3

### Capítulo 2. Vistas administrativas comparadas con funciones de tablas . . 5

### Capítulo 3. Vistas y rutinas administrativas SQL soportadas . . . . 7

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
Procedimiento ADMIN_CMD y rutinas asociadas . . . . .	36
ADMIN_CMD – Ejecutar mandatos administrativos . . . . .	36
Función de tabla ADMIN_GET_DBP_MEM_USAGE - Obtener el consumo de memoria total por instancia . . . . .	211
Función de tabla ADMIN_GET_MSGS - Recuperar mensajes generados por un programa de utilidad de movimiento de datos que utiliza el procedimiento ADMIN_CMD . . . . .	213
Vista ADMIN_TABCOMPRESSINFO y ADMIN_GET_TAB_COMPRESS_INFO . . . . .	215
Procedimiento ADMIN_REMOVE_MSGS - Limpiar los mensajes que genera un programa de utilidad de movimiento de datos que utiliza el procedimiento ADMIN_CMD . . . . .	219
Vista administrativa ADMIN_TABINFO y función de tabla ADMIN_GET_TAB_INFO_V95 – Recuperar información de tamaño y estado para las tablas . . . . .	220
Rutinas y procedimientos de auditoría . . . . .	228
Procedimiento y función de tabla AUDIT_ARCHIVE - Archivar archivo de registro de comprobación . . . . .	228

AUDIT_DELIM_EXTRACT - realiza una extracción al archivo delimitado . . . . .	229
Función de tabla AUDIT_LIST_LOGS - Enumera los archivos de registro de comprobación archivados . . . . .	230
Rutinas de mantenimiento automático . . . . .	231
Procedimiento AUTOMAINT_GET_POLICY - recuperar la política de mantenimiento automático . . . . .	231
Procedimiento AUTOMAINT_GET_POLICYFILE - recuperar el mantenimiento automático . . . . .	232
Procedimiento AUTOMAINT_SET_POLICY - configurar la política de mantenimiento automático . . . . .	234
Procedimiento AUTOMAINT_SET_POLICYFILE - configurar la política de mantenimiento automático . . . . .	235
Rutinas y vistas de configuración. . . . .	236
DB_PARTITIONS . . . . .	236
Vista administrativa DBCFG - Recuperar información sobre los parámetros de configuración de la base de datos . . . . .	237
Vista administrativa DBMCFG - Recuperar información sobre los parámetros de configuración del gestor de bases de datos . . . . .	239
Vista administrativa REG_VARIABLES - Recuperar valores de registro de DB2 en uso . . . . .	241
Vistas del entorno . . . . .	243
Vista administrativa ENV_INST_INFO – Recuperar información sobre la instancia actual . . . . .	243
Vista administrativa ENV_PROD_INFO - Recuperar información sobre productos DB2 instalados . . . . .	245
Vista administrativa ENV_FEATURE_INFO - Devolver información de licencia para las características de DB2 . . . . .	246
Vista administrativa ENV_SYS_INFO – Recuperar información sobre el sistema . . . . .	247
Vista administrativa ENV_SYS_RESOURCES - Devolver información de sistema . . . . .	248
Rutinas de instantáneas de salud . . . . .	252
EALTH_CONT_HI . . . . .	252
HEALTH_CONT_HI_HIS . . . . .	254
HEALTH_CONT_INFO . . . . .	256
HEALTH_DB_HI . . . . .	257
HEALTH_DB_HI_HIS . . . . .	261
HEALTH_DB_HIC . . . . .	264
HEALTH_DB_HIC_HIS . . . . .	266
HEALTH_DB_INFO . . . . .	269
HEALTH_DBM_HI . . . . .	270
HEALTH_DBM_HI_HIS . . . . .	271
HEALTH_DBM_INFO . . . . .	274

Función de tabla	Vista administrativa
HEALTH_GET_ALERT_ACTION_CFG -	CONTAINER_UTILIZATION – Recuperar
Recuperar valores de configuración de acciones	información de utilización y contenedores de
de alerta de salud . . . . . 275	espacio de tablas . . . . . 333
Función de tabla HEALTH_GET_ALERT_CFG –	Vista administrativa LOCKS_HELD – Recuperar
Recuperar valores de configuración de alertas	información acerca de los bloqueos retenidos. . . 335
de salud . . . . . 278	Vista administrativa LOCKWAITS – Recuperar
Función de tabla	información de bloqueos actuales en espera de
HEALTH_GET_IND_DEFINITION – Recuperar	ser activados . . . . . 337
definiciones de indicador de salud . . . . . 281	Vista administrativa LOG_UTILIZATION –
RHEALTH_HI_REC . . . . . 283	Recuperar información de utilización de las
HEALTH_TBS_HI . . . . . 285	anotaciones cronológicas . . . . . 341
HEALTH_TBS_HI_HIS . . . . . 287	LONG_RUNNING_SQL, vista administrativa 342
HEALTH_TBS_INFO . . . . . 291	Vista administrativa QUERY_PREP_COST –
Rutinas de MQSeries . . . . . 293	Recuperar información de tiempo de
MQPUBLISH . . . . . 293	preparación de sentencias . . . . . 345
MQREAD . . . . . 294	Vista administrativa SNAPAGENT y función de
MQREADALL . . . . . 295	tabla SNAP_GET_AGENT – Recuperar
MQREADALLCLOB . . . . . 297	información de instantáneas de la aplicación del
MQREADCLOB . . . . . 299	grupo de datos lógicos agent . . . . . 346
MQRECEIVE . . . . . 300	Vista administrativa
MQRECEIVEALL . . . . . 302	SNAPAGENT_MEMORY_POOL y función de
MQRECEIVEALLCLOB . . . . . 304	tabla SNAP_GET_AGENT_MEMORY_POOL –
MQRECEIVECLOB . . . . . 306	Recuperar información de instantáneas del
MQSEND . . . . . 307	grupo de datos lógicos memory_pool . . . . . 349
MQSUBSCRIBE . . . . . 309	Vista administrativa SNAPAPPL_INFO y
MQUNSUBSCRIBE . . . . . 310	función de tabla SNAP_GET_APPL_INFO_V95
Rutinas y vistas de seguridad . . . . . 311	– Recuperar información de instantáneas del
AUTH_LIST_AUTHORITIES_FOR_AUTHID . . 311	grupo de datos lógicos appl_info . . . . . 352
Función de tabla	Vista administrativa SNAPAPPL y función de
AUTH_LIST_GROUPS_FOR_AUTHID –	tabla SNAP_GET_APPL_V95 – Recuperar
Recuperación lista de pertenencia al grupo para	información de instantáneas del grupo de datos
un ID de autorización determinado . . . . . 315	lógicos appl . . . . . 360
Función AUTH_LIST_ROLES_FOR_AUTHID -	Vista administrativa SNAPBP y función de tabla
Devuelve la lista de funciones. . . . . 316	SNAP_GET_BP_V95 – Recuperar información
Vista administrativa AUTHORIZATIONIDS –	de instantáneas del grupo lógico bufferpool . . . 369
Recuperar ID y tipos de autorización . . . . . 318	Vista administrativa SNAPBP_PART y función
Vista administrativa OBJECTOWNERS –	de tabla SNAP_GET_BP_PART – Recuperar
Recuperar información sobre propiedad de	información de instantáneas del grupo de datos
objetos . . . . . 319	lógicos bufferpool_nodeinfo . . . . . 374
Vista administrativa PRIVILEGES – Recuperar	Vista administrativa SNAPCONTAINER y
información sobre privilegios . . . . . 320	función de tabla SNAP_GET_CONTAINER_V91
Rutinas y vistas de instantáneas . . . . . 322	– Recuperar información acerca de la
Vista administrativa APPL_PERFORMANCE –	instantánea de grupo de datos lógicos
Recuperar porcentaje de filas seleccionadas para	tablespace_container . . . . . 378
una aplicación . . . . . 322	Vista administrativa SNAPDB y función de
Vista administrativa APPLICATIONS –	tabla SNAP_GET_DB_V95 – Recuperar
Recuperar información de aplicación de base de	información de instantánea desde el grupo
datos conectada . . . . . 323	lógico dbase. . . . . 382
Vista administrativa BP_HITRATIO – Recuperar	Vista administrativa SNAPDB_MEMORY_POOL
información sobre la tasa de aciertos de	y función de tabla
agrupación de almacenamientos intermedios . . 327	SNAP_GET_DB_MEMORY_POOL – Recuperar
Vista administrativa BP_READ_IO – Recuperar	información sobre el uso de la memoria en el
información de rendimiento de lectura de las	nivel de la base de datos . . . . . 395
agrupaciones de almacenamientos intermedios . 329	Vista administrativa SNAPDBM y función de
Vista administrativa BP_WRITE_IO – Recuperar	tabla SNAP_GET_DBM_V95 – Recuperar la
información de rendimiento de grabación de las	información de instantáneas del grupo de datos
agrupaciones de almacenamientos intermedios . 331	lógicos dbm . . . . . 399

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 . . . . .	403	Vista administrativa SNAPTBSP_PART y función de tabla SNAP_GET_TBSP_PART_V91 – Recuperar información de instantáneas del grupo de datos lógicos tablespace_nodeinfo . . . . .	466
Vista administrativa SNAPDETAILLOG y función de tabla SNAP_GET_DETAILLOG_V91 – Recuperar información de instantánea desde el grupo de datos lógicos detail_log . . . . .	406	Vista administrativa SNAPTBSP_QUIESCER y función de tabla SNAP_GET_TBSP_QUIESCER – Recuperar información acerca de la instantánea de espacio de tablas de inmovilizador . . . . .	471
Vista administrativa SNAPDYN_SQL y función de tabla SNAP_GET_DYN_SQL_V95 – Recuperar información de instantáneas del grupo de datos lógicos dynsql. . . . .	409	Vista administrativa SNAPTBSP_RANGE y función de tabla SNAP_GET_TBSP_RANGE – Recuperar información acerca de la instantánea de rango . . . . .	474
Vista administrativa SNAPFCM y función de tabla SNAP_GET_FCM – Recuperar la información de instantáneas del grupo de datos lógicos fcm . . . . .	415	Vista administrativa SNAPUTIL y función de tabla SNAP_GET_UTIL – Recuperar información acerca de la instantánea de grupo de datos lógicos utility_info. . . . .	478
Vista administrativa SNAPFCM_PART y función de tabla SNAP_GET_FCM_PART – Recuperar la información de instantáneas del grupo de datos lógicos fcm_node . . . . .	417	Vista administrativa SNAPUTIL_PROGRESS y función de tabla SNAP_GET_UTIL_PROGRESS – Recuperar información acerca de la instantánea de grupo de datos lógicos de progreso . . . . .	481
Vista administrativa SNAPHADR y función de tabla SNAP_GET_HADR – Recuperar información de instantáneas del grupo de datos lógicos hadr . . . . .	419	Procedimiento SNAP_WRITE_FILE . . . . .	484
Vista administrativa SNAPLOCK y función de tabla SNAP_GET_LOCK – Recuperar información de instantáneas del grupo de datos lógicos lock . . . . .	423	Vista administrativa SNAPAGENT y función de tabla SNAP_GET_AGENT – Recuperar información de instantáneas de la aplicación del grupo de datos lógicos agent . . . . .	485
Vista administrativa SNAPLOCKWAIT y función de tabla SNAP_GET_LOCKWAIT – Recuperar información de instantáneas del grupo de datos lógicos lockwait . . . . .	429	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 . . . . .	488
Vista administrativa SNAPSTMT y función de tabla SNAP_GET_STMT – Recuperar información de instantáneas de sentencias. . . . .	435	Vista administrativa SNAPAPPL_INFO y función de tabla SNAP_GET_APPL_INFO_V95 – Recuperar información de instantáneas del grupo de datos lógicos appl_info. . . . .	492
Vista administrativa SNAPSTORAGE_PATHS y función de tabla SNAP_GET_STORAGE_PATHS – Recuperar información de vía de acceso del almacenamiento automático . . . . .	442	Vista administrativa SNAPAPPL y función de tabla SNAP_GET_APPL_V95 – Recuperar información de instantáneas del grupo de datos lógicos appl . . . . .	499
Vista administrativa SNAPSUBSECTION y función de tabla SNAP_GET_SUBSECTION – Recuperar información de instantáneas del grupo de supervisor lógico de la subsección . . . . .	444	Vista administrativa SNAPBP y función de tabla SNAP_GET_BP_V95 – Recuperar información de instantáneas del grupo lógico bufferpool . . . . .	508
Vista administrativa SNAPSWITCHES y función de tabla SNAP_GET_SWITCHES – Recuperar información acerca del estado del conmutador de instantánea de base de datos . . . . .	448	Vista administrativa SNAPBP_PART y función de tabla SNAP_GET_BP_PART – Recuperar información de instantáneas del grupo de datos lógicos bufferpool_nodeinfo . . . . .	513
Vista administrativa SNAPTAB y función de tabla SNAP_GET_TAB_V91 – Recuperar información de instantáneas del grupo de datos lógicos table. . . . .	451	Vista administrativa SNAPCONTAINER y función de tabla SNAP_GET_CONTAINER_V91 – Recuperar información acerca de la instantánea de grupo de datos lógicos tablespace_container . . . . .	517
Vista administrativa SNAPTAB_REORG y función de tabla SNAP_GET_TAB_REORG - Recuperar información sobre instantáneas de reorganización de tabla . . . . .	454	Vista administrativa SNAPDB y función de tabla SNAP_GET_DB_V95 – Recuperar información de instantánea desde el grupo lógico dbase. . . . .	521
Vista administrativa SNAPTBSP y función de tabla SNAP_GET_TBSP_V91 – Recuperar información de instantáneas del grupo de datos lógicos de espacio de tablas . . . . .	459	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 . . . . .	534

Vista administrativa SNAPDBM y función de tabla SNAP_GET_DBM_V95 – Recuperar la información de instantáneas del grupo de datos lógicos dbm . . . . .	538
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 . . . . .	542
Vista administrativa SNAPDETAILLOG y función de tabla SNAP_GET_DETAILLOG_V91 – Recuperar información de instantánea desde el grupo de datos lógicos detail_log . . . . .	545
Vista administrativa SNAPDYN_SQL y función de tabla SNAP_GET_DYN_SQL_V95 – Recuperar información de instantáneas del grupo de datos lógicos dynsql. . . . .	548
Vista administrativa SNAPFCM y función de tabla SNAP_GET_FCM – Recuperar la información de instantáneas del grupo de datos lógicos fcm . . . . .	554
Vista administrativa SNAPFCM_PART y función de tabla SNAP_GET_FCM_PART – Recuperar la información de instantáneas del grupo de datos lógicos fcm_node . . . . .	556
Vista administrativa SNAPHADR y función de tabla SNAP_GET_HADR – Recuperar información de instantáneas del grupo de datos lógicos hadr . . . . .	558
Vista administrativa SNAPLOCK y función de tabla SNAP_GET_LOCK – Recuperar información de instantáneas del grupo de datos lógicos lock . . . . .	562
Vista administrativa SNAPLOCKWAIT y función de tabla SNAP_GET_LOCKWAIT – Recuperar información de instantáneas del grupo de datos lógicos lockwait . . . . .	568
Vista administrativa SNAPSTMT y función de tabla SNAP_GET_STMT – Recuperar información de instantáneas de sentencias. . . . .	574
Vista administrativa SNAPSTORAGE_PATHS y función de tabla SNAP_GET_STORAGE_PATHS – Recuperar información de vía de acceso del almacenamiento automático . . . . .	581
Vista administrativa SNAPSUBSECTION y función de tabla SNAP_GET_SUBSECTION – Recuperar información de instantáneas del grupo de supervisor lógico de la subsección . . . . .	583
Vista administrativa SNAPSWITCHES y función de tabla SNAP_GET_SWITCHES – Recuperar información acerca del estado del conmutador de instantánea de base de datos . . . . .	587
Vista administrativa SNAPTAB y función de tabla SNAP_GET_TAB_V91 – Recuperar información de instantáneas del grupo de datos lógicos table. . . . .	590
Vista administrativa SNAPTAB_REORG y función de tabla SNAP_GET_TAB_REORG - Recuperar información sobre instantáneas de reorganización de tabla . . . . .	593

Vista administrativa SNAPTbsp y función de tabla SNAP_GET_TBSP_V91 – Recuperar información de instantáneas del grupo de datos lógicos de espacio de tablas . . . . .	598
Vista administrativa SNAPTbsp_PART y función de tabla SNAP_GET_TBSP_PART_V91 – Recuperar información de instantáneas del grupo de datos lógicos tablespace_nodeinfo . . . . .	605
Vista administrativa SNAPTbsp QUIESCER y función de tabla SNAP_GET_TBSP QUIESCER – Recuperar información acerca de la instantánea de espacio de tablas de inmovilizador . . . . .	610
Vista administrativa SNAPTbsp_RANGE y función de tabla SNAP_GET_TBSP_RANGE – Recuperar información acerca de la instantánea de rango . . . . .	613
Vista administrativa SNAPUTIL y función de tabla SNAP_GET_UTIL – Recuperar información acerca de la instantánea de grupo de datos lógicos utility_info. . . . .	617
Vista administrativa SNAPUTIL_PROGRESS y función de tabla SNAP_GET_UTIL_PROGRESS – Recuperar información acerca de la instantánea de grupo de datos lógicos de progreso . . . . .	620
Procedimiento SNAP_WRITE_FILE . . . . .	623
Vista administrativa TBSP_UTILIZATION – Recuperar información de utilización y configuración de espacio de tablas . . . . .	624
Vista administrativa TOP_DYNAMIC_SQL – Recuperar información acerca de las sentencias de SQL dinámico más frecuentes . . . . .	628
Rutinas de procedimientos de SQL . . . . .	629
RGET_ROUTINE_OPTS . . . . .	629
GET_ROUTINE_SAR . . . . .	630
PUT_ROUTINE_SAR . . . . .	631
REBIND_ROUTINE_PACKAGE . . . . .	632
RSET_ROUTINE_OPTS . . . . .	633
Rutinas de redistribución paso a paso . . . . .	633
Procedimiento ANALYZE_LOG_SPACE - Recuperar información de análisis de espacio de anotaciones . . . . .	633
Procedimiento GENERATE_DISTFILE - Generar archivo de distribución de datos . . . . .	636
Procedimiento GET_SWRD_SETTINGS - Recuperar información de redistribución . . . . .	637
Procedimiento SET_SWRD_SETTINGS - Crear o cambiar registro de redistribución . . . . .	639
Procedimiento STEPWISE_REDISTRIBUTE_DBPG - Redistribuir parte del grupo de particiones de base de datos . . . . .	642
Rutinas de la herramienta de gestión de almacenamiento . . . . .	643
Procedimiento CAPTURE_STORAGEMGMT_INFO – Recuperar información relacionada con el almacenamiento para un objeto raíz concreto . . . . .	643



Procedimiento	PD_GET_DIAG_HIST - Devolver registros de un determinado recurso . . . . .	720
CREATE_STORAGEMGMT_TABLES – Creación de tablas de gestión de almacenamiento . . . . .	Vista administrativa	
Procedimiento	PDLOGMSG_LAST24HOURS y función de tabla PD_GET_LOG_MSGS – Recuperar mensajes de determinación de problemas . . . . .	727
DROP_STORAGEMGMT_TABLES – Eliminar todas las tablas de gestión de almacenamiento . . . . .	Procedimiento REORGCHK_IX_STATS - Recuperar estadísticas de índice para evaluación de la reorganización . . . . .	734
Rutinas de gestión de cargas de trabajo. . . . .	Procedimiento REORGCHK_TB_STATS - Recuperar estadísticas de tabla para evaluación de la reorganización . . . . .	736
WLM_CANCEL_ACTIVITY - Cancelar una actividad . . . . .	Funciones escalares SQLERRM – Recuperar información de mensajes de error . . . . .	738
WLM_CAPTURE_ACTIVITY_IN_PROGRESS - Recopilar información para supervisor de sucesos de actividades . . . . .	SYSINSTALOBJECTS . . . . .	740
WLM_COLLECT_STATS - Coleccionar y restaurar estadísticas de gestión de carga de trabajo. . . . .		
WLM_GET_ACTIVITY_DETAILS - Devolver información detallada sobre una actividad específica. . . . .		
Función de tabla WLM_GET_QUEUE_STATS - Devolver estadísticas de cola de umbral . . . . .		
WLM_GET_SERVICE_CLASS_AGENTS - Listar agentes en ejecución en una clase de servicio. . . . .		
WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES - Lista de ocurrencias de carga de trabajo . . . . .		
WLM_GET_SERVICE_SUBCLASS_STATS - Devolver estadísticas de subclases de servicio . . . . .		
WLM_GET_SERVICE_SUPERCLASS_STATS - Devolver estadísticas de superclases de servicio. . . . .		
WLM_GET_WORK_ACTION_SET_STATS - Devolver estadísticas de conjunto de acciones de trabajo. . . . .		
WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES - Devolver una lista de actividades . . . . .		
WLM_GET_WORKLOAD_STATS - Devolver estadísticas de carga de trabajo . . . . .		
Rutinas y vistas varias . . . . .		
Procedimiento ADMIN_COPY_SCHEMA – Copiar un esquema específico y sus objetos . . . . .		
Procedimiento ADMIN_DROP_SCHEMA - Eliminar un esquema específico y sus objetos. . . . .		
ALTOBJ . . . . .		
RAPPLICATION_ID . . . . .		
Función de tabla COMPILATION_ENV – Recuperación de elementos del entorno de compilación . . . . .		
Vista administrativa CONTACTGROUPS – Recuperar la lista de grupos de contactos . . . . .		
Vista administrativa CONTACTS - Recuperar lista de contactos . . . . .		
Vista administrativa DB_HISTORY – Recuperar información de archivos históricos . . . . .		
Vista administrativa DBPATHS – Recuperar vías de acceso de bases de datos . . . . .		
EXPLAIN_FORMAT_STATS . . . . .		
EXPLAIN_GET_MSGS . . . . .		
RGET_DBSIZE_INFO. . . . .		
Vista administrativa NOTIFICATIONLIST – Recuperar lista de contactos para notificación de salud . . . . .		
	<b>Capítulo 4. Rutinas administrativas de SQL obsoletas y sus vistas o rutinas de sustitución . . . . .</b>	<b>743</b>
	Tabla de función ADMIN_GET_TAB_INFO - Recuperar información de tamaño y estado para las tablas . . . . .	747
	RGET_DB_CONFIG . . . . .	757
	GET_DBM_CONFIG . . . . .	758
	Función de tabla SNAP_GET_APPL – Recuperar información de instantáneas del grupo de datos lógico appl . . . . .	759
	Función de tabla SNAP_GET_APPL_INFO – Recuperación de información de instantáneas del grupo de datos lógico appl_info . . . . .	767
	Función de tabla SNAP_GET_BP – Recuperación de la información de instantáneas del grupo de datos lógico bufferpool . . . . .	773
	SNAP_GET_CONTAINER . . . . .	778
	SNAP_GET_DB . . . . .	779
	Función de tabla SNAP_GET_DBM – Recuperación de la información de instantáneas del grupo de datos lógicos dbm . . . . .	787
	Función de tabla SNAP_GET_DB_V91 - Recuperar información de instantánea desde el grupo lógico dbase . . . . .	790
	Función de tabla SNAP_GET_DYN_SQL_V91 - Recuperar información de la instantánea del grupo de datos lógicos dynsql . . . . .	801
	SNAP_GET_DYN_SQL . . . . .	805
	SNAP_GET_STO_PATHS . . . . .	807
	SNAP_GET_TAB . . . . .	808
	SNAP_GET_TBSP . . . . .	809
	SNAP_GET_TBSP_PART . . . . .	813
	SNAPSHOT_AGENT. . . . .	815
	SNAPSHOT_APPL . . . . .	816
	SNAPSHOT_APPL_INFO . . . . .	822
	SNAPSHOT_BP . . . . .	824
	SNAPSHOT_CONTAINER . . . . .	827
	SNAPSHOT_DATABASE . . . . .	828
	SNAPSHOT_DBM. . . . .	835
	SNAPSHOT_DYN_SQL . . . . .	837
	SNAPSHOT_FCM. . . . .	838
	SNAPSHOT_FCMNODE . . . . .	840
	SNAPSHOT_FILEW . . . . .	840

SNAPSHOT_LOCK . . . . .	841
SNAPSHOT_LOCKWAIT . . . . .	842
SNAPSHOT_QUIESCERS . . . . .	844
SNAPSHOT_RANGES . . . . .	845
SNAPSHOT_STATEMENT . . . . .	846
SNAPSHOT_SUBSECT . . . . .	848
SNAPSHOT_SWITCHES . . . . .	850
SNAPSHOT_TABLE . . . . .	852
SNAPSHOT_TBREORG . . . . .	853
SNAPSHOT_TBS . . . . .	854
SNAPSHOT_TBS_CFG . . . . .	857
SQLCACHE_SNAPSHOT . . . . .	859
RSYSINSTALLROUTINES . . . . .	860

---

**Parte 2. Apéndices . . . . . 861**

<b>Apéndice A. Visión general de la información técnica de DB2 . . . . .</b>	<b>863</b>
--	------------

Biblioteca técnica de DB2 en copia impresa o en formato PDF . . . . .	864
Pedido de manuales de DB2 en copia impresa . . . . .	866
Visualización de la ayuda para estados de SQL desde el procesador de línea de mandatos. . . . .	867
Acceso a diferentes versiones del Centro de información de DB2 . . . . .	867
Visualización de temas en su idioma preferido en el Centro de información de DB2. . . . .	867
Actualización del Centro de información de DB2 instalado en el sistema o en el servidor de Intranet. . . . .	868
Guías de aprendizaje de DB2 . . . . .	870
Información de resolución de problemas de DB2 . . . . .	870
Términos y condiciones . . . . .	871

**Apéndice B. Avisos . . . . . 873**

<b>Índice. . . . .</b>	<b>877</b>
------------------------	------------

---

## Parte 1. Vistas y rutinas administrativas SQL

Las vistas y rutinas administrativas proporcionan una interfaz de programa principal fácil de utilizar para administrar DB2 mediante SQL. Incluyen una colección de vistas incorporadas, funciones de tabla, procedimientos y funciones escalares para realizar diversas tareas administrativas de DB2. Por ejemplo: 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.



---

## Capítulo 1. 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 una función 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.



---

## Capítulo 2. Vistas administrativas comparadas con funciones de tablas

DB2 Versión 9.5 presenta 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, \_V95). 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 3. Vistas y rutinas administrativas SQL 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 8
- Comprobar rutinas y procedimientos Tabla 3 en la página 8
- Vistas y rutinas administrativas SQL del mantenimiento automático: Tabla 4 en la página 8
- Vistas y rutinas administrativas SQL de la configuración: Tabla 5 en la página 9
- Entorno de vistas administrativas: Tabla 6 en la página 9
- Rutinas administrativas SQL de instantáneas de salud: Tabla 7 en la página 10
- Rutinas administrativas SQL de MQSeries: Tabla 8 en la página 11
- Vistas y rutinas administrativas SQL de la seguridad: Tabla 9 en la página 12
- Vistas y rutinas administrativas SQL de instantáneas: Tabla 10 en la página 13
- Rutinas administrativas SQL de procedimientos SQL: Tabla 11 en la página 17
- Rutinas administrativas SQL de redistribución paso a paso: Tabla 12 en la página 17
- Rutinas administrativas SQL de herramientas de almacenamiento: Tabla 13 en la página 18
- Rutinas de gestión de carga de trabajo: Tabla 14 en la página 18
- Vistas y rutinas administrativas SQL diversas: Tabla 15 en la página 19

Tabla 1. Rutinas administrativas SQL de supervisión de actividad

Nombre de la rutina	Esquema	Descripción
"AM_BASE_RPT_RECOMS – Recomendaciones para informes de actividad" en la página 21	SYSPROC	Esta función de tabla devuelve las recomendaciones para los informes de actividad que utiliza el supervisor de actividad.
"AM_BASE_RPTS – Informes del supervisor de actividad" en la página 22	SYSPROC	Esta función de tabla devuelve los informes de actividad que utiliza el supervisor de actividad.
"RAM_DROP_TASK – Supresión de una tarea de supervisión" en la página 23	SYSPROC	Este procedimiento suprime una tarea de supervisión.
"RAM_GET_LOCK_CHN_TB – Recuperación de los datos de la cadena de bloqueo de la aplicación en formato tabular" en la página 24	SYSPROC	Este procedimiento devuelve datos de la cadena de bloqueo de aplicación en formato tabular.
"RAM_GET_LOCK_CHNS – Recuperación de información de bloqueo de cadenas para una aplicación específica" en la página 25	SYSPROC	Este procedimiento visualiza las series de bloqueo de una aplicación especificada utilizando una serie con formato.
"RAM_GET_LOCK_RPT – Recuperación de los detalles del bloqueo de la aplicación" en la página 25	SYSPROC	Este procedimiento visualiza los detalles del bloque de una aplicación.
"RAM_GET_RPT – Recuperación de datos del supervisor de actividad" en la página 33	SYSPROC	Este procedimiento visualiza los datos sobre el supervisor de actividad para un informe.

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

Nombre de la rutina	Esquema	Descripción
“RAM_SAVE_TASK – Creación o modificación de una tarea de supervisión” en la página 34	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 – Ejecutar mandatos administrativos” en la página 36	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 de tabla ADMIN_GET_DBP_MEM_USAGE - Obtener el consumo de memoria total por instancia” en la página 211	SYSPROC	Esta función obtiene el consumo total de memoria de una instancia determinada.
“Función de tabla ADMIN_GET_MSGS - Recuperar mensajes generados por un programa de utilidad de movimiento de datos que utiliza el procedimiento ADMIN_CMD” en la página 213	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.
“Vista ADMINTABCOMPRESSINFO y ADMIN_GET_TAB_COMPRESS_INFO” en la página 215	SYSPROC	Esta función devuelve información de compresión para tablas, tablas de consultas materializadas (MQT) y tablas de jerarquía.
“Procedimiento ADMIN_REMOVE_MSGS - Limpiar los mensaje que genera un programa de utilidad de movimiento de datos que utiliza el procedimiento ADMIN_CMD” en la página 219	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.

Tabla 3. Comprobar rutinas y procedimientos

Nombre de la vista o rutina	Esquema	Descripción
“Procedimiento y función de tabla AUDIT_ARCHIVE - Archivar archivo de registro de comprobación” en la página 228	SYSPROC	Este procedimiento y la función de tabla archivan el registro de comprobación actual.
“AUDIT_DELIM_EXTRACT - realiza una extracción al archivo delimitado” en la página 229	SYSPROC	Este procedimiento extrae datos de los registros archivados de forma binaria y los carga en archivos delimitados.
“Función de tabla AUDIT_LIST_LOGS - Enumera los archivos de registro de comprobación archivados” en la página 230	SYSPROC	Este procedimiento devuelve una lista de los registros de comprobación archivados en la vía de acceso especificada para la base de datos actual.

Tabla 4. Vistas y rutinas administrativas SQL del mantenimiento automático

Nombre de la vista o rutina	Esquema	Descripción
“Procedimiento AUTOMAINT_GET_POLICY - recuperar la política de mantenimiento automático” en la página 231	SYSPROC	Este procedimiento obtiene los valores actuales del mantenimiento automático para la base de datos.

Tabla 4. Vistas y rutinas administrativas SQL del mantenimiento automático (continuación)

Nombre de la vista o rutina	Esquema	Descripción
“Procedimiento AUTOMAINT_GET_POLICYFILE - recuperar el mantenimiento automático” en la página 232	SYSPROC	Este procedimiento obtiene los valores actuales del mantenimiento automático para la base de datos.
“Procedimiento AUTOMAINT_SET_POLICY - configurar la política de mantenimiento automático” en la página 234	SYSPROC	Este procedimiento establece los valores de la política de mantenimiento automático para la base de datos conectada actualmente.
“Procedimiento AUTOMAINT_SET_POLICYFILE - configurar la política de mantenimiento automático” en la página 235	SYSPROC	Este procedimiento establece los valores del mantenimiento automático para la base de datos conectada actualmente.

Tabla 5. Vistas y rutinas administrativas SQL de la configuración

Nombre de la vista o rutina	Esquema	Descripción
“DB_PARTITIONS” en la página 236	SYSPROC	Esta función de tabla devuelve el contenido del archivo db2nodes.cfg en formato de tabla.
“Vista administrativa DBCFG - Recuperar información sobre los parámetros de configuración de la base de datos” en la página 237	SYSIBMADM	Esta vista administrativa devuelve información sobre la configuración de la base de datos.
“Vista administrativa DBMCFG - Recuperar información sobre los parámetros de configuración del gestor de bases de datos” en la página 239	SYSIBMADM	Esta vista administrativa devuelve información sobre la configuración del gestor de bases de datos.
“Vista administrativa REG_VARIABLES - Recuperar valores de registro de DB2 en uso” en la página 241	SYSIBMADM	Esta vista administrativa devuelve los valores de registro de DB2 de todas las particiones de bases de datos.

Tabla 6. Vistas administrativas del entorno

Nombre de vista	Esquema	Descripción
“Vista administrativa ENV_FEATURE_INFO - Devolver información de licencia para las características de DB2” en la página 246	SYSPROC	Esta función de tabla devuelve información sobre todas las características disponibles para las que se necesita una licencia.
“Vista administrativa ENV_INST_INFO – Recuperar información sobre la instancia actual” en la página 243	SYSIBMADM	Esta vista administrativa devuelve información acerca de la instancia actual.
“Vista administrativa ENV_PROD_INFO - Recuperar información sobre productos DB2 instalados” en la página 245	SYSIBMADM	Esta vista administrativa devuelve información acerca de los productos DB2 instalados.
“Vista administrativa ENV_SYS_INFO – Recuperar información sobre el sistema” en la página 247	SYSIBMADM	Esta vista administrativa devuelve información acerca del sistema.
“Vista administrativa ENV_SYS_RESOURCES - Devolver información de sistema” en la página 248	SYSIBMADM	Esta vista administrativa indica el sistema operativo, la CPU, la memoria y demás información relacionada con el sistema

Tabla 7. Rutinas administrativas SQL de instantáneas de salud

Nombre de la rutina	Esquema	Descripción
"EALTH_CONT_HI" en la página 252	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.
"HEALTH_CONT_HI_HIS" en la página 254	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" en la página 256	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.
"HEALTH_DB_HI" en la página 257	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.
"HEALTH_DB_HI_HIS" en la página 261	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" en la página 264	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" en la página 266	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" en la página 269	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.
"HEALTH_DBM_HI" en la página 270	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.
"HEALTH_DBM_HI_HIS" en la página 271	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" en la página 274	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.

Tabla 7. Rutinas administrativas SQL de instantáneas de salud (continuación)

Nombre de la rutina	Esquema	Descripción
“Función de tabla HEALTH_GET_ALERT_ACTION_CFG - Recuperar valores de configuración de acciones de alerta de salud” en la página 275	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 tabla y contenedores de espacios de tabla) y a diversos niveles de configuración (valor por omisión de instalación, instancia, global y objeto).
“Función de tabla HEALTH_GET_ALERT_CFG – Recuperar valores de configuración de alertas de salud” en la página 278	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 tabla y contenedores de espacios de tabla) y a diversos niveles de configuración (valor por omisión de instalación, global y objeto).
“Función de tabla HEALTH_GET_IND_DEFINITION – Recuperar definiciones de indicador de salud” en la página 281	SYSPROC	Esta función de tabla devuelve las definiciones del indicador de salud.
“RHEALTH_HI_REC” en la página 283	SYSPROC	Este procedimiento recupera un conjunto de recomendaciones que direccionan un indicador de salud en estado de alerta en un objeto DB2 determinado.
“HEALTH_TBS_HI” en la página 285	SYSPROC	Esta función de tabla devuelve una tabla con la información sobre el indicador de salud de los espacios de tabla a partir de una instantánea de salud de una base de datos.
“HEALTH_TBS_HI_HIS” en la página 287	SYSPROC	Esta función de tabla devuelve una tabla con la información sobre el indicador histórico de salud de los espacios de tabla a partir de una instantánea de salud de una base de datos.
“HEALTH_TBS_INFO” en la página 291	SYSPROC	Esta función de tabla devuelve una tabla con la información sobre el estado de la alerta retrotraída para espacios de tabla a partir de una instantánea de salud de una base de datos.

Tabla 8. Rutinas administrativas SQL de MQSeries

Nombre de la rutina	Esquema	Descripción
“MQPUBLISH” en la página 293	DB2MQ, DB2MQ1C	Esta función escalar publica datos en una ubicación MQSeries.
“MQREAD” en la página 294	DB2MQ, DB2MQ1C	Esta función escalar devuelve un mensaje de una ubicación MQSeries.
“MQREADALL” en la página 295	DB2MQ, DB2MQ1C	Esta función de tabla devuelve una tabla con mensajes y metadatos de mensajes de una ubicación MQSeries.

Tabla 8. Rutinas administrativas SQL de MQSeries (continuación)

Nombre de la rutina	Esquema	Descripción
"MQREADALLCLOB" en la página 297	DB2MQ	Esta función de tabla devuelve una tabla que contiene mensajes y metadatos de mensajes desde una ubicación MQSeries especificada.
"MQREADCLOB" en la página 299	DB2MQ	Esta función escalar devuelve un mensaje de una ubicación MQSeries especificada.
"MQRECEIVE" en la página 300	DB2MQ, DB2MQ1C	Esta función escalar devuelve un mensaje de una ubicación MQSeries y suprime el mensaje de la cola asociada.
"MQRECEIVEALL" en la página 302	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.
"MQRECEIVEALLCLOB" en la página 304	DB2MQ	Esta función de tabla devuelve una tabla que contiene mensajes y metadatos de mensajes desde una ubicación MQSeries especificada.
"MQRECEIVECLOB" en la página 306	DB2MQ	Esta función escalar devuelve un mensaje de una ubicación MQSeries especificada.
"MQSEND" en la página 307	DB2MQ, DB2MQ1C	Esta función escalar envía datos a una ubicación MQSeries.
"MQSUBSCRIBE" en la página 309	DB2MQ, DB2MQ1C	Esta función escalar suscribe a los mensajes MQSeries publicados sobre un tema específico.
"MQUNSUBSCRIBE" en la página 310	DB2MQ, DB2MQ1C	Esta función escalar cancela la suscripción a los mensajes MQSeries publicados sobre un tema específico.

Tabla 9. Vistas y rutinas administrativas SQL de la seguridad:

Nombre de la vista o rutina	Esquema	Descripción
"AUTH_LIST_AUTHORITIES_FOR_AUTHID" en la página 311	SYSPROC	Esta función 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 una función.
"Función de tabla AUTH_LIST_GROUPS_FOR_AUTHID – Recuperación lista de pertenencia al grupo para un ID de autorización determinado" en la página 315	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 - Devuelve la lista de funciones" en la página 316	SYSPROC	Esta función devuelve la lista de funciones de los que es miembro el ID de autorización determinado.
"Vista administrativa AUTHORIZATIONIDS – Recuperar ID y tipos de autorización" en la página 318	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.

Tabla 9. Vistas y rutinas administrativas SQL de la seguridad: (continuación)

Nombre de la vista o rutina	Esquema	Descripción
“Vista administrativa OBJECTOWNERS – Recuperar información sobre propiedad de objetos” en la página 319	SYSIBMADM	Esta vista administrativa contiene toda la información sobre la propiedad de objetos correspondiente a la base de datos conectada actualmente.
“Vista administrativa PRIVILEGES – Recuperar información sobre privilegios” en la página 320	SYSIBMADM	Esta vista administrativa contiene todos los privilegios explícitos correspondientes a la base de datos conectada actualmente.

Tabla 10. Vistas y rutinas administrativas SQL de instantáneas

Nombre de la vista o rutina	Esquema	Descripción
“Vista administrativa APPLICATIONS – Recuperar información de aplicación de base de datos conectada” en la página 323	SYSIBMADM	Esta vista administrativa devuelve información sobre las aplicaciones de base de datos conectadas.
“Vista administrativa APPL_PERFORMANCE – Recuperar porcentaje de filas seleccionadas para una aplicación” en la página 322	SYSIBMADM	Esta vista administrativa muestra información acerca del índice filas seleccionado en comparación con las filas leídas por aplicación.
“Vista administrativa BP_HITRATIO – Recuperar información sobre la tasa de aciertos de agrupación de almacenamientos intermedios” en la página 327	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 – Recuperar información de rendimiento de lectura de las agrupaciones de almacenamientos intermedios” en la página 329	SYSIBMADM	Esta vista administrativa devuelve información sobre el rendimiento de lectura de la agrupación de almacenamientos intermedios.
“Vista administrativa BP_WRITE_IO – Recuperar información de rendimiento de grabación de las agrupaciones de almacenamientos intermedios” en la página 331	SYSIBMADM	Esta vista administrativa devuelve información sobre el rendimiento de grabación de cada agrupación de almacenamientos intermedios.
“Vista administrativa CONTAINER_UTILIZATION – Recuperar información de utilización y contenedores de espacio de tablas” en la página 333	SYSIBMADM	Esta vista administrativa devuelve información acerca de los contenedores de espacios de tabla y los índices de utilización.
“Vista administrativa LOCKS_HELD – Recuperar información acerca de los bloqueos retenidos” en la página 335	SYSIBMADM	Esta vista administrativa devuelve información sobre los bloqueos actuales.
“Vista administrativa LOCKWAITS – Recuperar información de bloqueos actuales en espera de ser activados” en la página 337	SYSIBMADM	Esta vista administrativa devuelve información sobre los bloqueos que están a la espera de ser concedidos.
“Vista administrativa LOG_UTILIZATION – Recuperar información de utilización de las anotaciones cronológicas” en la página 341	SYSIBMADM	Esta vista administrativa devuelve información sobre la utilización de las anotaciones cronológicas para la base de datos conectada actualmente.
“LONG_RUNNING_SQL, vista administrativa” en la página 342	SYSIBMADM	Esta vista administrativa devuelve las sentencias SQL de ejecución más larga de la base de datos conectada actualmente.
“Vista administrativa QUERY_PREP_COST – Recuperar información de tiempo de preparación de sentencias” en la página 345	SYSIBMADM	Esta vista administrativa devuelve una lista de las sentencias con información acerca del tiempo necesario para preparar la sentencia.

Tabla 10. Vistas y rutinas administrativas SQL de instantáneas (continuación)

Nombre de la vista o rutina	Esquema	Descripción
“Procedimiento SNAP_WRITE_FILE” en la página 484	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 – Recuperar información de instantáneas de la aplicación del grupo de datos lógicos agent” en la página 346	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 – Recuperar información de instantáneas del grupo de datos lógicos memory_pool” en la página 349	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 – Recuperar información de instantáneas del grupo de datos lógicos appl” en la página 360	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 – Recuperar información de instantáneas del grupo de datos lógicos appl_info” en la página 352	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 – Recuperar información de instantáneas del grupo lógico bufferpool” en la página 369	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 – Recuperar información de instantáneas del grupo de datos lógicos bufferpool_nodeinfo” en la página 374	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.
“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 378	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 – Recuperar información de instantánea desde el grupo lógico dbase” en la página 382	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).



Tabla 10. Vistas y rutinas administrativas SQL de instantáneas (continuación)

Nombre de la vista o rutina	Esquema	Descripción
“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” en la página 395	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 – Recuperar la información de instantáneas del grupo de datos lógicos dbm” en la página 399	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 – Recuperar información sobre el uso de la memoria en el nivel del gestor de bases de datos” en la página 403	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 – Recuperar información de instantánea desde el grupo de datos lógicos detail_log” en la página 406	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 – Recuperar información de instantáneas del grupo de datos lógicos dynsql” en la página 409	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 – Recuperar la información de instantáneas del grupo de datos lógicos fcm” en la página 415	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 – Recuperar la información de instantáneas del grupo de datos lógicos fcm_node” en la página 417	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 – Recuperar información de instantáneas del grupo de datos lógicos hadr” en la página 419	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 – Recuperar información de instantáneas del grupo de datos lógicos lock” en la página 423	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.
“Vista administrativa SNAPLOCKWAIT y función de tabla SNAP_GET_LOCKWAIT – Recuperar información de instantáneas del grupo de datos lógicos lockwait” en la página 429	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.

Tabla 10. Vistas y rutinas administrativas SQL de instantáneas (continuación)

Nombre de la vista o rutina	Esquema	Descripción
“Vista administrativa SNAPSTMT y función de tabla SNAP_GET_STMT – Recuperar información de instantáneas de sentencias” en la página 435	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 – Recuperar información de vía de acceso del almacenamiento automático” en la página 442	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 – Recuperar información de instantáneas del grupo de supervisor lógico de la subsección” en la página 444	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 – Recuperar información acerca del estado del conmutador de instantánea de base de datos” en la página 448	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 – Recuperar información de instantáneas del grupo de datos lógicos table” en la página 451	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 - Recuperar información sobre instantáneas de reorganización de tabla” en la página 454	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 – Recuperar información de instantáneas del grupo de datos lógicos de espacio de tablas” en la página 459	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 – Recuperar información de instantáneas del grupo de datos lógicos tablespace_nodeinfo” en la página 466	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 – Recuperar información acerca de la instantánea de espacio de tablas de inmovilizador” en la página 471	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 – Recuperar información acerca de la instantánea de rango” en la página 474	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 – Recuperar información acerca de la instantánea de grupo de datos lógicos utility_info” en la página 478	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 10. 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 – Recuperar información acerca de la instantánea de grupo de datos lógicos de progreso” en la página 481	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 – Recuperar información de utilización y configuración de espacio de tablas” en la página 624	SYSIBMADM	Esta vista administrativa devuelve información de utilización y configuración de espacio de tablas.
“Vista administrativa TOP_DYNAMIC_SQL – Recuperar información acerca de las sentencias de SQL dinámico más frecuentes” en la página 628	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 11. Rutinas administrativas SQL de procedimientos SQL

Nombre de la rutina	Esquema	Descripción
“RGET_ROUTINE_OPTS” en la página 629	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” en la página 630	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.
“PUT_ROUTINE_SAR” en la página 631	SYSFUN	Este procedimiento pasa la información necesaria para crear y definir una rutina SQL en el servidor de bases de datos.
“REBIND_ROUTINE_PACKAGE” en la página 632	SYSPROC	Este procedimiento vuelve a vincular el paquete asociado con un procedimiento de SQL.
“RSET_ROUTINE_OPTS” en la página 633	SYSPROC	Este procedimiento establece las opciones que deben utilizarse para la creación de procedimientos SQL en la sesión actual.

Tabla 12. Rutinas administrativas SQL de redistribución paso a paso

Nombre de la rutina	Esquema	Descripción
“Procedimiento ANALYZE_LOG_SPACE - Recuperar información de análisis de espacio de anotaciones” en la página 633	SYSPROC	Este procedimiento devuelve información de análisis del espacio de anotaciones cronológicas.
“Procedimiento GENERATE_DISTFILE - Generar archivo de distribución de datos” en la página 636	SYSPROC	Este procedimiento genera un archivo de distribución de datos.
“Procedimiento GET_SWRD_SETTINGS - Recuperar información de redistribución” en la página 637	SYSPROC	Este procedimiento devuelve información de redistribución.

Tabla 12. Rutinas administrativas SQL de redistribución paso a paso (continuación)

Nombre de la rutina	Esquema	Descripción
“Procedimiento SET_SWRD_SETTINGS - Crear o cambiar registro de redistribución” en la página 639	SYSPROC	Este procedimiento crea o modifica el registro de redistribución.
“Procedimiento STEPWISE_REDISTRIBUTE_DBPG - Redistribuir parte del grupo de particiones de base de datos” en la página 642	SYSPROC	Este procedimiento redistribuye parte de un grupo de particiones de base de datos.

Tabla 13. Rutinas administrativas SQL de la herramienta de administración del almacenamiento

Nombre de la rutina	Esquema	Descripción
“Procedimiento CAPTURE_STORAGEMGMT_INFO – Recuperar información relacionada con el almacenamiento para un objeto raíz concreto” en la página 643	SYSPROC	Este procedimiento devuelve información relacionada con el almacenamiento de un objeto raíz concreto.
“Procedimiento CREATE_STORAGEMGMT_TABLES – Creación de tablas de gestión de almacenamiento” en la página 644	SYSPROC	Este procedimiento crea tablas de gestión de almacenamiento.
“Procedimiento DROP_STORAGEMGMT_TABLES – Eliminar todas las tablas de gestión de almacenamiento” en la página 645	SYSPROC	Este procedimiento inactiva las tablas de gestión de almacenamiento.

Tabla 14. Rutinas administrativas SQL de la herramienta de mantenimiento automático

Nombre de la rutina	Esquema	Descripción
“WLM_CANCEL_ACTIVITY - Cancelar una actividad” en la página 646	SYSPROC	Este procedimiento cancela la actividad dada.
“WLM_CAPTURE_ACTIVITY_IN_PROGRESS - Recopilar información para supervisor de sucesos de actividades” en la página 647	SYSPROC	Este procedimiento envía información de una actividad determinada al supervisor de eventos de actividades.
“WLM_COLLECT_STATS - Coleccionar y restaurar estadísticas de gestión de carga de trabajo” en la página 649	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.
“WLM_GET_ACTIVITY_DETAILS - Devolver información detallada sobre una actividad específica” en la página 650	SYSPROC	Esta función devuelve información detallada sobre una actividad específica identificada por su APPLICATION_HANDLE, UOW_ID y ACTIVITY_ID.
“Función de tabla WLM_GET_QUEUE_STATS - Devolver estadísticas de cola de umbral” en la página 657	SYSPROC	Esta función devuelve información estadística básica de una o más colas de umbral.
“WLM_GET_SERVICE_CLASS_AGENTS - Listar agentes en ejecución en una clase de servicio” en la página 661	SYSPROC	Esta función 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.

Tabla 14. Rutinas administrativas SQL de la herramienta de mantenimiento automático (continuación)

Nombre de la rutina	Esquema	Descripción
WLM_GET_SERVICE_CLASS_WORKLOAD_OCCURRENCES - Lista de ocurrencias de carga de trabajo	SYSPROC	Esta función devuelve la lista de todas las ocurrencias de carga de trabajo ejecutadas en una clase de servicio dada en una partición determinada.
“WLM_GET_SERVICE_SUBCLASS_STATS - Devolver estadísticas de subclases de servicio” en la página 671	SYSPROC	Esta función devuelve las estadísticas básicas de una o más subclases de servicio.
“WLM_GET_SERVICE_SUPERCLASS_STATS - Devolver estadísticas de superclases de servicio” en la página 676	SYSPROC	Esta función devuelve las estadísticas básicas de una o más superclases de servicio.
“WLM_GET_WORK_ACTION_SET_STATS - Devolver estadísticas de conjunto de acciones de trabajo” en la página 678	SYSPROC	Esta función devuelve las estadísticas básicas de clases de trabajo en un conjunto de acciones de trabajo.
WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES - Devuelve una lista de actividades	SYSPROC	Esta función devuelve la lista de todas las actividades sometidas mediante la aplicación dada en la partición especificada y que todavía no se han completado.
“WLM_GET_WORKLOAD_STATS - Devolver estadísticas de carga de trabajo” en la página 684	SYSPROC	Esta función devuelve las estadísticas básicas de una o más cargas de trabajo.

Tabla 15. Vistas y rutinas administrativas SQL varias

Nombre de la vista o rutina	Esquema	Descripción
“Procedimiento ADMIN_COPY_SCHEMA – Copiar un esquema específico y sus objetos” en la página 687	SYSPROC	Este procedimiento sirve para copiar un esquema específico y todos los objetos contenidos en el mismo.
“Procedimiento ADMIN_DROP_SCHEMA - Eliminar un esquema específico y sus objetos” en la página 692	SYSPROC	Este procedimiento sirve para desactivar un esquema específico y todos los objetos contenidos en el mismo.
“Vista administrativa ADMINTABINFO y función de tabla ADMIN_GET_TAB_INFO_V95 – Recuperar información de tamaño y estado para las tablas” en la página 220	SYSIBMADM (vista administrativa), SYSPROC (función de tabla)	Esta vista administrativa y función de tabla devuelven información de tamaño y estado para las tablas, las tablas de consultas materializadas (MQT) y las tablas de jerarquía.
“ALTOBJ” en la página 694	SYSPROC	Este procedimiento altera una tabla existente utilizando la sentencia CREATE TABLE de entrada como la definición de tabla de destino.
“RAPPLICATION_ID” en la página 696	SYSFUN	Esta función escalar devuelve el ID de aplicación de la conexión actual.
“Función de tabla COMPILATION_ENV – Recuperación de elementos del entorno de compilación” en la página 697	SYSPROC	Esta función de tabla devuelve los elementos de un entorno de compilación.
“Vista administrativa CONTACTGROUPS – Recuperar la lista de grupos de contactos” en la página 699	SYSIBMADM	Esta vista administrativa devuelve información acerca de la lista de contactos.
“Vista administrativa CONTACTS - Recuperar lista de contactos” en la página 700	SYSIBMADM	La vista administrativa devuelve la lista de contactos definidos en el servidor de bases de datos.

Tabla 15. Vistas y rutinas administrativas SQL varias (continuación)

Nombre de la vista o rutina	Esquema	Descripción
“Vista administrativa DB_HISTORY – Recuperar información de archivos históricos” en la página 701	SYSIBMADM	Esta administrativa devuelve información del archivo histórico asociado a la partición de base de datos conectada actualmente.
“Vista administrativa DBPATHS – Recuperar vías de acceso de bases de datos” en la página 706	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” en la página 710	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.
“EXPLAIN_GET_MSGS” en la página 714	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.
“RGET_DBSIZE_INFO” en la página 717	SYSPROC	Este procedimiento calcula el tamaño de base de datos y la capacidad máxima.
“PD_GET_DIAG_HIST - Devolver registros de un determinado recurso” en la página 720	SYSPROC	La función devuelve registros de anotación cronológica, registros de sucesos y registros de notificación de un recurso determinado.
“Vista administrativa NOTIFICATIONLIST – Recuperar lista de contactos para notificación de salud” en la página 719	SYSIBMADM	Esta vista administrativa devuelve la lista de contactos y grupos de contactos a los que se notifica sobre la salud de una instancia.
“Vista administrativa PDLOGMSGs_LAST24HOURS y función de tabla PD_GET_LOG_MSGS – Recuperar mensajes de determinación de problemas” en la página 727	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.
“Procedimiento REORGCHK_IX_STATS - Recuperar estadísticas de índice para evaluación de la reorganización” en la página 734	SYSPROC	Este procedimiento comprueba las estadísticas del índice para determinar si es o no es necesaria una reorganización.
“Procedimiento REORGCHK_TB_STATS - Recuperar estadísticas de tabla para evaluación de la reorganización” en la página 736	SYSPROC	Este procedimiento comprueba las estadísticas de la tabla para determinar si es o no es necesaria una reorganización.
“Funciones escalares SQLERRM – Recuperar información de mensajes de error” en la página 738	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” en la página 740	SYSPROC	El procedimiento crea o descarta los objetos de base de datos requeridos para una herramienta específica.

---

## 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 16. 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.
RECOM_NAME	VARCHAR(256)	El nombre o descripción abreviada de la recomendación.

Tabla 16. Información devuelta por la función de tabla  
AM\_BASE\_RPT\_RECOMS (continuación)

Nombre de columna	Tipo de datos	Descripción
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 17. 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

Eliminar 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.

### Parámetro 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 18. 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 de parám. de salida
```

```
-----
```

```
Nombre del parámetro : LOCK_CHAINS
```

```
Valor del 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 retorno = 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 19. 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 la aplicación
AUTH_ID	VARCHAR(128)	auth_id - ID de autorización
APPL_ID	VARCHAR(128)	appl_id - ID de aplicación

Tabla 19. 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"> <li>• BACKUP</li> <li>• COMMIT_ACT</li> <li>• COMP</li> <li>• CONNECTED</li> <li>• CONNECTPEND</li> <li>• CREATE_DB</li> <li>• DECOUPLED</li> <li>• DISCONNECTPEND</li> <li>• INTR</li> <li>• IOERROR_WAIT</li> <li>• LOAD</li> <li>• LOCKWAIT</li> <li>• QUIESCE_TABLESPACE</li> <li>• RECOMP</li> <li>• REMOTE_RQST</li> <li>• RESTART</li> <li>• RESTORE</li> <li>• ROLLBACK_ACT</li> <li>• ROLLBACK_TO_SAVEPOINT</li> <li>• TEND</li> <li>• THABRT</li> <li>• THCOMT</li> <li>• TPREP</li> <li>• UNLOAD</li> <li>• UOWEXEC</li> <li>• UOWWAIT</li> <li>• WAITFOR_REMOTE</li> </ul>
COORD_PARTITION_NUM	SMALLINT	coord_node - Coordinación del nodo
SEQUENCE_NO	VARCHAR(4)	sequence_no - Número de secuencia
CLIENT_PRDID	VARCHAR(128)	client_prdid - ID de producto/versión del cliente
CLIENT_PID	BIGINT	client_pid - ID de proceso de cliente

Tabla 19. 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"> <li>• AIX</li> <li>• AIX64</li> <li>• AS400_DRDA</li> <li>• DOS</li> <li>• DYNIX</li> <li>• HP</li> <li>• HP64</li> <li>• HPIA</li> <li>• HPIA64</li> <li>• LINUX</li> <li>• LINUX390</li> <li>• LINUXIA64</li> <li>• LINUXPPC</li> <li>• LINUXPPC64</li> <li>• LINUXX8664</li> <li>• LINUXZ64</li> <li>• MAC</li> <li>• MVS_DRDA</li> <li>• NT</li> <li>• NT64</li> <li>• OS2</li> <li>• OS390</li> <li>• SCO</li> <li>• SGI</li> <li>• SNI</li> <li>• SUN</li> <li>• SUN64</li> <li>• UNKNOWN</li> <li>• UNKNOWN_DRDA</li> <li>• VM_DRDA</li> <li>• VSE_DRDA</li> <li>• WINDOWS</li> <li>• WINDOWS95</li> </ul>

Tabla 19. 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"> <li>• CPIC</li> <li>• LOCAL</li> <li>• NETBIOS</li> <li>• NPIPE</li> <li>• TCPIP (para DB2 Universal Database o DB2 UDB)</li> <li>• TCPIP4</li> <li>• TCPIP6</li> </ul>
CLIENT_NNAME	VARCHAR(128)	El elemento del supervisor client_nname está 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 la 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 escalas de bloqueo
X_LOCK_ESCALS	BIGINT	x_lock_escals - Escalas de bloqueo exclusivas
DEADLOCKS	BIGINT	deadlocks - Puntos muertos detectados

Tabla 20. 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
TABSHEMA	VARCHAR(128)	table_schema - Nombre de esquema de tabla
TABNAME	VARCHAR(128)	table_name - Nombre de tabla

Tabla 20. 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"> <li>• AUTORESIZE_LOCK</li> <li>• AUTOSTORAGE_LOCK</li> <li>• BLOCK_LOCK</li> <li>• EOT_LOCK</li> <li>• INPLACE_REORG_LOCK</li> <li>• INTERNAL_LOCK</li> <li>• INTERNALB_LOCK</li> <li>• INTERNALC_LOCK</li> <li>• INTERNALJ_LOCK</li> <li>• INTERNALL_LOCK</li> <li>• INTERNALO_LOCK</li> <li>• INTERNALQ_LOCK</li> <li>• INTERNALP_LOCK</li> <li>• INTERNALS_LOCK</li> <li>• INTERNALT_LOCK</li> <li>• INTERNALV_LOCK</li> <li>• KEYVALUE_LOCK</li> <li>• ROW_LOCK</li> <li>• SYSBOOT_LOCK</li> <li>• TABLE_LOCK</li> <li>• TABLE_PART_LOCK</li> <li>• TABLESPACE_LOCK</li> <li>• XML_PATH_LOCK</li> </ul>



Tabla 20. Información retenida por los bloqueos devuelta por el procedimiento AM\_GET\_LOCK\_RPT (continuación)

Nombre de columna	Tipo de datos	Descripción
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"> <li>• IN</li> <li>• IS</li> <li>• IX</li> <li>• NON (si no existe bloqueo)</li> <li>• NS</li> <li>• NW</li> <li>• NX</li> <li>• S</li> <li>• SIX</li> <li>• U</li> <li>• W</li> <li>• X</li> <li>• Z</li> </ul>
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"> <li>• CONV</li> <li>• GRNT</li> </ul>
LOCK_ESCALATION	SMALLINT	lock_escalation - Escala de bloqueo
LOCK_NAME	VARCHAR(32)	lock_name - Nombre de bloqueo
DBPARTITIONNUM	SMALLINT	La partición de base de datos desde la que se han recuperado los datos para esta fila.

Tabla 21. 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 - Bloqueo de retención de ID de agente
APPL_ID_HOLDING_LK	VARCHAR(128)	appl_id_holding_lk - Bloqueo de retención de ID de aplicación
LOCK_WAIT_START_TIME	TIMESTAMP	lock_wait_start_time - Indicación de la hora de inicio de espera de bloqueo
DBPARTITIONNUM	SMALLINT	La 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

Tabla 21. Información de espera de los bloqueos devuelta por el procedimiento AM\_GET\_LOCK\_RPT (continuación)

Nombre de columna	Tipo de datos	Descripción
TABSHEMA	VARCHAR(128)	table_schema - Nombre de esquema de tabla
TABNAME	VARCHAR(128)	table_name - Nombre de tabla
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"> <li>• AUTORESIZE_LOCK</li> <li>• AUTOSTORAGE_LOCK</li> <li>• BLOCK_LOCK</li> <li>• EOT_LOCK</li> <li>• INPLACE_REORG_LOCK</li> <li>• INTERNAL_LOCK</li> <li>• INTERNALB_LOCK</li> <li>• INTERNALC_LOCK</li> <li>• INTERNALJ_LOCK</li> <li>• INTERNALL_LOCK</li> <li>• INTERNALO_LOCK</li> <li>• INTERNALQ_LOCK</li> <li>• INTERNALP_LOCK</li> <li>• INTERNALS_LOCK</li> <li>• INTERNALT_LOCK</li> <li>• INTERNALV_LOCK</li> <li>• KEYVALUE_LOCK</li> <li>• ROW_LOCK</li> <li>• SYSBOOT_LOCK</li> <li>• TABLE_LOCK</li> <li>• TABLE_PART_LOCK</li> <li>• TABLESPACE_LOCK</li> <li>• XML_PATH_LOCK</li> </ul>

Tabla 21. Información de espera de los bloqueos devuelta por el procedimiento AM\_GET\_LOCK\_RPT (continuación)

Nombre de columna	Tipo de datos	Descripción
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"> <li>• IN</li> <li>• IS</li> <li>• IX</li> <li>• NON (si no existe bloqueo)</li> <li>• NS</li> <li>• NW</li> <li>• NX</li> <li>• S</li> <li>• SIX</li> <li>• U</li> <li>• W</li> <li>• X</li> <li>• Z</li> </ul>
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"> <li>• IN</li> <li>• IS</li> <li>• IX</li> <li>• NON (si no existe bloqueo)</li> <li>• NS</li> <li>• NW</li> <li>• NX</li> <li>• S</li> <li>• SIX</li> <li>• U</li> <li>• W</li> <li>• X</li> <li>• Z</li> </ul>
LOCK_ESCALATION	SMALLINT	lock_escalation - Escala de bloqueo

## 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')
```

---

## 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 sobre uso

Recuperación de la información de ejecución del mandato:

- Puesto que el procedimiento ADMIN\_CMD se ejecuta en el servidor, 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  $\geq 0$ , 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 22. 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 $\geq 0$ : Información adicional (conjuntos de resultados) devuelta, si la hay.	El SQLCODE devuelto es $\geq 0$ : Información adicional (conjuntos de resultados) devuelta, si la hay, pero las columnas MSG_RETRIEVAL y MSG_REMOVAL son NULL.

Tabla 22. *SQLCODE* e información devueltos por el procedimiento *ADMIN\_CMD* (continuación)

Estado de ejecución del mandato administrativo	Opción <b>MESSAGES ON SERVER</b> especificada	Opción <b>MESSAGES ON SERVER</b> no especificada
Error	El <b>SQLCODE</b> ha devuelto 20397: Información adicional (conjuntos de resultados) devuelta, pero sólo las columnas <b>MSG_RETRIEVAL</b> y <b>MSG_REMOVAL</b> están llenas.	El <b>SQLCODE</b> 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.

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.

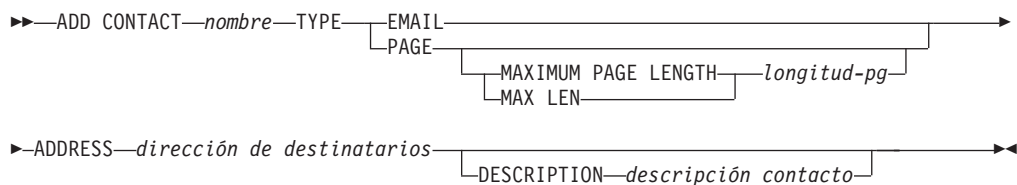
### Autorización

Ninguna.

### Conexión necesaria

Base de datos. El DAS debe estar en ejecución.

### Sintaxis del mandato



### Parámetros del mandato

#### **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 contactos es una lista de usuarios y grupos a los que procesos de supervisión, 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.

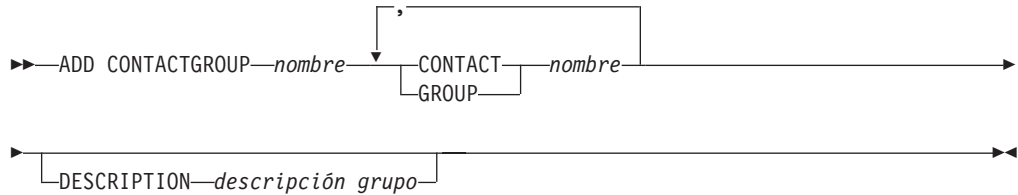
**Autorización**

Ninguna

## Conexión necesaria

Base de datos. El DAS debe estar en ejecución.

## Sintaxis del mandato



## Parámetros del mandato

### **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 añadirlo 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.

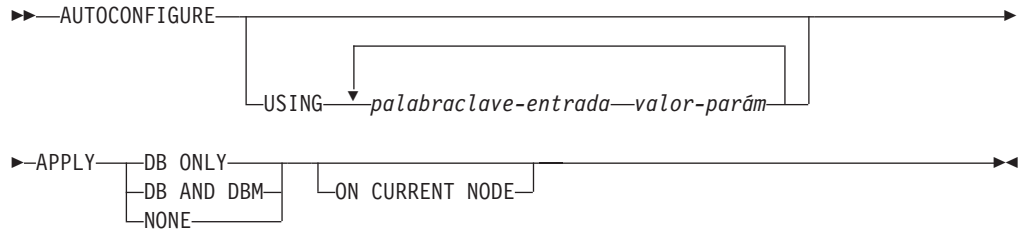
## Autorización

*sysadm.*

## Conexión necesaria

Base de datos.

## Sintaxis del mandato



## Parámetros del mandato

### USING palabraclave-entrada valor-parám

Tabla 23. Palabras claves de entrada y valores de parámetro válidos

Palabra clave	Valores válidos	Valor por omisión	Explicación
mem_percent	1-100	25	Porcentaje de memoria que se va a dedicar. Si otras aplicaciones (aparte del sistema operativo) se están ejecutando en este servidor, establezca este valor en menos de 100.
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

Tabla 23. Palabras claves de entrada y valores de parámetro válidos (continuación)

Palabra clave	Valores válidos	Valor por omisión	Explicación
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
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 comprometida). 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 24. 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"> <li>• BP para nivel de agrupación de almacenamientos intermedios</li> <li>• DBM para nivel de gestor de bases de datos</li> <li>• DB para nivel de base de datos</li> </ul>
NAME	VARCHAR(128)	<ul style="list-style-type: none"> <li>• Si LEVEL es DB o DBM, este valor contiene la palabra clave de parámetro de configuración.</li> <li>• Si LEVEL es BP, este valor contiene el nombre de agrupación de almacenamientos intermedios.</li> </ul>

Tabla 24. Conjunto de resultados devuelto por el mandato AUTOCONFIGURE (continuación)

Nombre de columna	Tipo de datos	Descripción
VALUE	VARCHAR(256)	<ul style="list-style-type: none"> <li>• 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.</li> <li>• 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.</li> </ul>
RECOMMENDED_VALUE	VARCHAR(256)	<ul style="list-style-type: none"> <li>• Si LEVEL es DB o DBM, esta columna contiene el valor recomendado (o aplicado) del parámetro de configuración identificado en la columna del parámetro.</li> <li>• 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.</li> </ul>
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 específica 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`

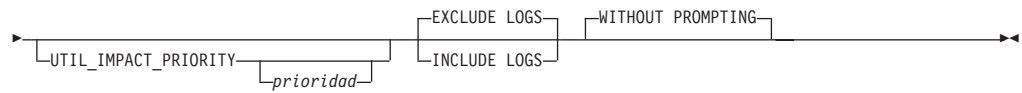
## Conexión necesaria

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

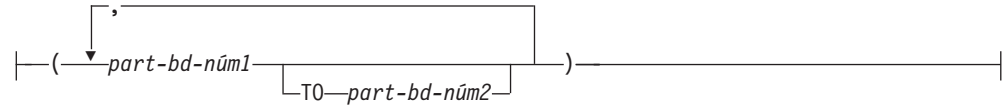
## Sintaxis del mandato







### 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 número1-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-tabla*

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-almacins* 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

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 solicitud 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-almacints* **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:

`bufferpages <= ST_MAX_BUFFERS * 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. Éste 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.

## 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 25. 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 la 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

Este mandato:

- Visualiza información de salida acerca de una sentencia SELECT, CALL o XQuery
- Visualiza columnas de una tabla o una vista
- Visualiza índices de una tabla o una vista
- Visualiza particiones de datos de una tabla o una vista

## Autorización

Para visualizar información de salida acerca de una sentencia SELECT, se necesita uno de los privilegios o una de las autorizaciones que se listan más abajo para cada tabla o vista a la que se hace referencia en la sentencia SELECT.

Para visualizar las columnas, índices o particiones de datos de una tabla o una vista, se necesita el privilegio SELECT, el privilegio CONTROL, la autorización *sysadm* o la autorización *dbadm* sobre los siguientes catálogos del sistema:

- SYSCAT.COLUMNS (DESCRIBE TABLE), SYSCAT.DATAPARTITIONEXPRESSION (con SHOW DETAIL)
- Privilegio de ejecutar SYSCAT.INDEXES (DESCRIBE INDEXES FOR TABLE) en GET\_INDEX\_COLNAMES() UDF (con SHOW DETAIL)
- SYSCAT.DATAPARTITIONS (DESCRIBE DATA PARTITIONS FOR TABLE)

Dado que PUBLIC tiene todos los privilegios sobre tablas temporales globales declaradas, un usuario puede utilizar el mandato para visualizar información acerca de cualquier tabla temporal global declarada que exista en su conexión.

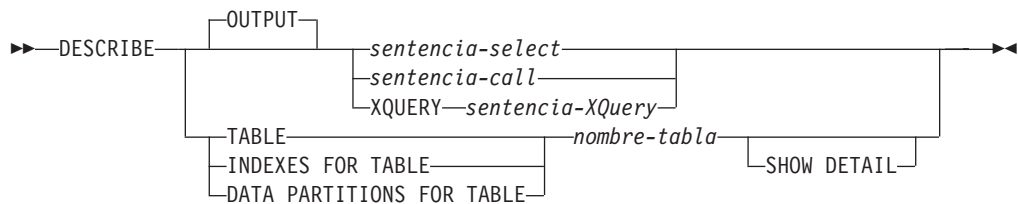
Para visualizar información de salida acerca de una sentencia CALL, se necesita uno de los privilegios o una de las autorizaciones que se listan más abajo:

- Privilegio EXECUTE para el procedimiento almacenado
- Autorización *sysadm* o *dbadm*

## Conexión necesaria

Base de datos.

## Sintaxis del mandato



## 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. El *esquema* es el nombre de usuario bajo el que se ha creado la tabla o la vista. 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. 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.

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

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

Para el mandato DESCRIBE INDEXES FOR TABLE, el nombre de índice se trunca cuanto tiene más de 18 bytes. La salida incluye la siguiente información adicional:

- Nombres de columna

#### **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)

## 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 retorno = 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 retorno = 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.

```
CALL SYSPROC.ADMIN_CMD('describe indexes for table t1')
```

El siguiente ejemplo muestra una salida de esta sentencia CALL.

Conjunto de resultados 1

```

-----
INDSCHEMA      INDNAME          UNIQUE_RULE      NUMBER_OF_COLUMNS COLNAMES
-----...-...-...-...-...-...-...-...-...-...-...-...-...
SYSIBM         SQL050117181625680 PRIMARY_INDEX      1 +PK
TXU           T1_INDEX1        DUPLICATES_ALLOWED 1 +C1

```

2 registro(s) seleccionado(s).

Estado de retorno = 0

### 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 retorno = 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 retorno = 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 de 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:

- Mandatos Tabla 26: DESCRIBE *sentencia-select*, DESCRIBE *sentencia-call* y DESCRIBE XQUERY *sentencia-XQuery*
- Tabla 27 en la página 60: Conjunto de resultados 1 del mandato DESCRIBE TABLE
- Tabla 28 en la página 61: Conjunto de resultados 2 del mandato DESCRIBE TABLE
- Tabla 29 en la página 61: Mandato DESCRIBE INDEXES FOR TABLE
- Tabla 30 en la página 62: Conjunto de resultados 1 del mandato DESCRIBE DATA PARTITIONS FOR TABLE
- Tabla 31 en la página 62: Conjunto de resultados 2 del mandato DESCRIBE DATA PARTITIONS FOR TABLE

Tabla 26. Conjunto de resultados devuelto por los mandatos DESCRIBE *sentencia-select*, DESCRIBE *sentencia-call* y DESCRIBE XQUERY *sentencia-XQuery*

Nombre de columna	Tipo de datos	Solo LOB <sup>1</sup>	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.
SQLLEN	INTEGER	No	Atributo longitud de la columna, tal como aparece en el campo SQLLEN del SQLDA.

Tabla 26. Conjunto de resultados devuelto por los mandatos DESCRIBE sentencia-select, DESCRIBE sentencia-call y DESCRIBE XQUERY sentencia-XQuery (continuación)

Nombre de columna	Tipo de datos	Solo LOB <sup>1</sup>	Descripción
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_TYPESHEMA	VARCHAR (128)	Sí	Nombre del esquema de tipo de datos.
SQLDATA_TYPENAME	VARCHAR (128)	Sí	Nombre del tipo de datos.

**Nota:** <sup>1</sup>: el valor Sí (Yes) indica que solo se devuelven valores no-nulos cuando se describen datos LOB.

Tabla 27. Conjunto de resultados 1 devuelto por el mandato DESCRIBE TABLE

Nombre de columna	Tipo de datos	Detalle <sup>2</sup>	Descripción
COLNAME	VARCHAR (128)	No	Nombre de la columna.
YPESHEMA	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.
NULLABLE	CHAR(1)	No	Uno de los siguientes: <ul style="list-style-type: none"> <li>• 'Y' si la columna tiene posibilidad de nulos</li> <li>• 'N' si la columna no tiene posibilidad de nulos</li> </ul>
COLNO	SMALLINT	Sí	Ordinal de la columna.

Tabla 27. Conjunto de resultados 1 devuelto por el mandato DESCRIBE TABLE (continuación)

Nombre de columna	Tipo de datos	Detalle <sup>2</sup>	Descripción
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"> <li>• Valor de la página de códigos de base de datos de columnas no definidas con FOR BIT DATA.</li> <li>• Valor de la página de códigos DBCS para columnas gráficas.</li> <li>• 0, en caso contrario.</li> </ul>
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:** <sup>2</sup>: el valor Sí (Yes) indica que solo se devuelven valores no-nulos cuando se utiliza la cláusula SHOW DETAIL.

Tabla 28. 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 29. Conjunto de resultados devuelto por el mandato DESCRIBE INDEXES FOR TABLE

Nombre de columna	Tipo de datos	Detalle <sup>2</sup>	Descripción
INDSCHEMA	VARCHAR (128)	No	Nombre del esquema de índice.
INDNAME	VARCHAR (128)	No	Nombre del índice.

Tabla 29. Conjunto de resultados devuelto por el mandato DESCRIBE INDEXES FOR TABLE (continuación)

Nombre de columna	Tipo de datos	Detalle <sup>2</sup>	Descripción
UNIQUE_RULE	VARCHAR (30)	No	Uno de los siguientes: <ul style="list-style-type: none"> <li>• DUPLICATES_ALLOWED</li> <li>• PRIMARY_INDEX</li> <li>• UNIQUE_ENTRIES_ONLY</li> </ul>
COLCOUNT	SMALLINT	No	Número de columnas de la clave, más el número de columnas de inclusión, si hay alguna.
COLNAMES	VARCHAR (2048)	Sí	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:** <sup>2</sup>: el valor Sí (Yes) indica que solo se devuelven valores no-nulos cuando se utiliza la cláusula SHOW DETAIL.

Tabla 30. Conjunto de resultados 1 devuelto por el mandato DESCRIBE DATA PARTITIONS FOR TABLE

Nombre de columna	Tipo de datos	Detalle <sup>2</sup>	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:** <sup>2</sup>: el valor Sí (Yes) indica que solo se devuelven valores no-nulos cuando se utiliza la cláusula SHOW DETAIL.

Tabla 31. 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.



Tabla 31. Conjunto de resultados 2 devuelto por el mandato *DESCRIBE DATA PARTITIONS FOR TABLE* cuando se utiliza la cláusula *SHOW DETAIL*. (continuación)

Nombre de columna	Tipo de datos	Descripción
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.
ACCESSMODE	VARCHAR (20)	Define la accesibilidad de la partición de datos y uno de estos valores: <ul style="list-style-type: none"> <li>• FULL_ACCESS</li> <li>• NO_ACCESS</li> <li>• NO_DATA_MOVEMENT</li> <li>• READ_ONLY</li> </ul>
STATUS	VARCHAR(64)	Estado de la partición de datos y puede ser uno de estos valores: <ul style="list-style-type: none"> <li>• NEWLY_ATTACHED</li> <li>• NEWLY_DETACHED: se necesita mantenimiento de MQT.</li> <li>• INDEX_CLEANUP_PENDING: partición de datos desconectada cuya tupla de SYSDATAPARTITIONS sólo se mantiene para hacer limpieza de índices. Esta tupla se elimina cuando se hayan suprimido todos los registros de índices que hagan referencia a la partición de datos desenlazada.</li> </ul> <p>De lo contrario, la columna está en blanco.</p>

## 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 71.

### Autorización

Una de las siguientes:

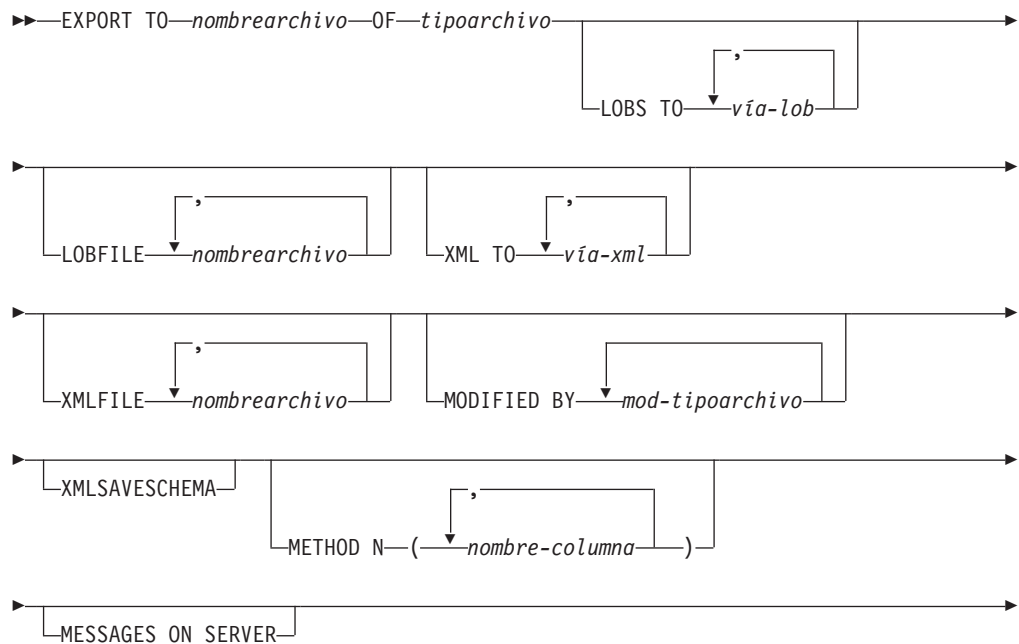
- *sysadm*
- *dbadm*

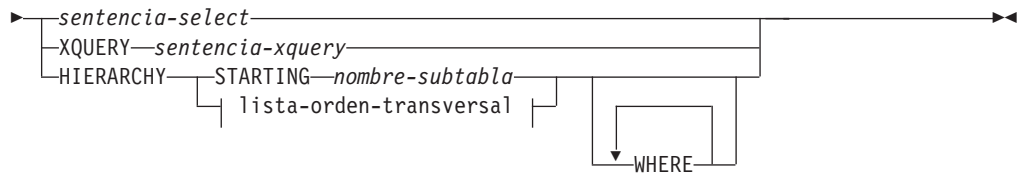
o privilegio CONTROL o SELECT en cada tabla o vista participante.

### 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. El número máximo de nombres de archivo que se puede especificar es 999. 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 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.

### **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 SQL necesaria para recuperar todos los mensajes de aviso y de error producidos durante esta operación, y `MSG_REMOVAL`, que es la sentencia 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 71.

**OF** *tipoarchivo*

Especifica el formato de los datos del archivo de salida:

- DEL (formato ASCII delimitado), utilizado por diversos programas 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.

- 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** *nombreadarchivo*

Especifica el nombre del archivo al que hay que exportar datos en el servidor. Debe ser una vía de acceso totalmente calificada y que ya exista 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** *nombreadarchivo*

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 COMMIT después de cerrar todos los cursores abiertos WITH HOLD o emitiendo un 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 la 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 32. Conjunto de resultados devuelto por el mandato EXPORT*

Nombre de columna	Tipo de datos	Descripción
ROWS_EXPORTED	BIGINT	Número total de filas exportadas.
MSG_RETRIEVAL	VARCHAR(512)	Sentencia SQL que sirve para recuperar los mensajes creados por este programa de utilidad. Por ejemplo: SELECT SQLCODE, MSG FROM TABLE (SYSPROC.ADMIN_GET_MSGS ( '3203498_txu' )) AS MSG



Tabla 32. Conjunto de resultados devuelto por el mandato EXPORT (continuación)

Nombre de columna	Tipo de datos	Descripción
MSG_REMOVAL	VARCHAR(512)	Sentencia SQL que sirve para hacer limpieza de los mensajes creados por este programa de utilidad. Por ejemplo: CALL SYSPROC.ADMIN_REMOVE_MSGS ('3203498_txu')

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

Tabla 33. 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 db2exp.001.123.456/ en el archivo de datos, el LOB está ubicado en el desplazamiento 123 dentro del archivo db2exp.001 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 db2exp.001.7.-1/.</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.
xmlgraphic	Si se especifica el modificador xmlgraphic 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 codepage.

Tabla 34. Modificadores de tipo de archivo válidos para el programa de utilidad de exportación: Formato 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 delimitar una serie.<sup>2</sup> 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>x 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 x00 a x3F, inclusive. El modificador codepage no se puede utilizar con el modificador lobsinfile.</p>
coldelx	<p>x 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.<sup>2</sup></p> <p>En el ejemplo siguiente, coldel; 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	<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 (.). Se utiliza el carácter especificado como carácter de coma decimal.<sup>2</sup></p>
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 chardelx o nodoubledel. Son opciones que se excluyen mutuamente.</p>
nodoubledel	<p>Suprime el reconocimiento de los delimitadores de caracteres dobles.<sup>2</sup></p>



Tabla 34. 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 la hora en el archivo fuente.<sup>4</sup> Los elementos de indicación de la hora son:</p> <ul style="list-style-type: none"> <li>AAAA - Año (cuatro dígitos entre 0000 y 9999)</li> <li>M - Mes (uno o dos dígitos entre 1 y 12)</li> <li>MM - Mes (dos dígitos entre 01 y 12; se excluye mutuamente con M y MMM)</li> <li>MMM - Mes (abreviación de tres letras no sensible a mayúsculas y minúsculas para el nombre del mes; se excluye mutuamente con M y MM)</li> <li>D - Día (uno o dos dígitos entre 1 y 31)</li> <li>DD - Día (dos dígitos entre 1 y 31; se excluye mutuamente con D)</li> <li>DDD - Día del año (tres dígitos entre 001 y 366; se excluye mutuamente con otros elementos de día o mes)</li> <li>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)</li> <li>HH - Hora (dos dígitos entre 0 y 12, para un sistema de 12 horas y 0 y 24 para uno de 24 horas; se excluye mutuamente con H)</li> <li>M - Minuto (uno o dos dígitos entre 0 y 59)</li> <li>MM - Minuto (dos dígitos entre 0 y 59; se excluye mutuamente con M, minuto)</li> <li>S - Segundo (uno o dos dígitos entre 0 y 59)</li> <li>SS - Segundo (dos dígitos entre 0 y 59; se excluye mutuamente con S)</li> <li>SSSSS - Segundo del día después de medianoche (5 dígitos entre 00000 y 86399; se excluye mutuamente con otros elementos de hora)</li> <li>UUUUUU - Microsegundo (6 dígitos entre 000000 y 999999; se excluye mutuamente con todos los otros elementos de microsegundo)</li> <li>UUUUU - Microsegundo (5 dígitos entre 00000 y 99999, se correlaciona con un rango de 000000 a 999990; se excluye mutuamente con todos los otros elementos de microsegundo)</li> <li>UUUU - Microsegundo (cuatro dígitos entre 0000 y 9999, se correlaciona con un rango de 000000 a 999900; se excluye mutuamente con todos los otros elementos de microsegundo)</li> <li>UUU - Microsegundo (3 dígitos entre 000 y 999 se correlaciona con un rango de 000000 a 999000; se excluye mutuamente con todos los otros elementos de microsegundo)</li> <li>UU - Microsegundo (2 dígitos entre 00 y 99, se correlaciona con un rango de 000000 a 990000; se excluye mutuamente con todos los otros elementos de microsegundo)</li> <li>U - Microsegundo (1 dígito entre 0 y 9, se correlaciona con un rango de 000000 a 900000; se excluye mutuamente con todos los otros elementos de microsegundo)</li> <li>TT - Indicador de meridiano (AM o PM)</li> </ul> <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 la 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 35. 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 36. Modificadores de tipo de archivo válidos para el programa de utilidad de exportación: Formato de archivo WSF

Modificador	Descripción
1	Crea un archivo WSF compatible con Lotus 1-2-3 Release 1 o Lotus 1-2-3 Release 1a. <sup>5</sup> Es el valor por omisión.
2	Crea un archivo WSF compatible con Lotus Symphony Release 1.0. <sup>5</sup>
3	Crea un archivo WSF compatible con Lotus 1-2-3 Versión 2 o Lotus Symphony Release 1.1. <sup>5</sup>
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 la hora en formato "AAAA-MM-DD-HH.MM.SS.uuuuuu"

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 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 la 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.
6. El formato de archivo WSF no está soportado para columnas de XML.
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 3, 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 bien  
.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 a los usuarios locales o remotos a salir del sistema para permitir el 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/sqllib/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

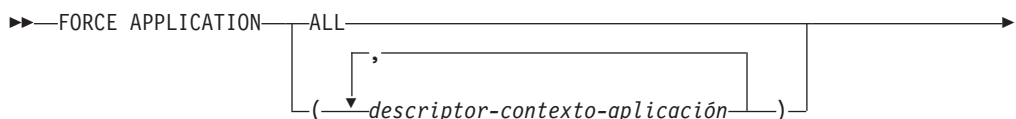
Una de las siguientes:

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

### Conexión necesaria

Base de datos.

### Sintaxis del mandato



## Parámetros del mandato

### 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-contexto-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 de 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.

No se puede forzar a salir a los usuarios que están creando una base de datos.

Después de haber emitido un FORCE, 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 STTM (Self Tuning Memory Manager) preferido por el usuario y el número de partición de base de datos de ajuste STMM actual.



## Autorización

Autorización SYSADM o DBADM

## 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 retorno = 0

## Notas sobre 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 ejecución del mandato se devuelve en la SQLCA resultante de la sentencia CALL. Si la ejecución es correcta, el mandato devuelve información adicional en el conjunto de resultados siguiente:

Tabla 37. 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.

Tabla 37. Conjunto de resultados devuelto por el mandato GET STMM TUNING DBPARTITIONNUM (continuación)

Nombre de columna	Tipo de datos	Descripción
CURRENT_NUMBER	INTEGER	El 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 95.

### Autorización

- IMPORT utilizando la opción INSERT requiere una de las autorizaciones siguientes:
  - *sysadm*
  - *dbadm*
  - 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:
  - *sysadm*
  - *dbadm*
  - 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:
  - *sysadm*
  - *dbadm*
  - Privilegio CONTROL sobre la tabla o la 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:
  - *sysadm*
  - *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:
  - *sysadm*
  - *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:
  - *sysadm*
  - *dbadm*
  - 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

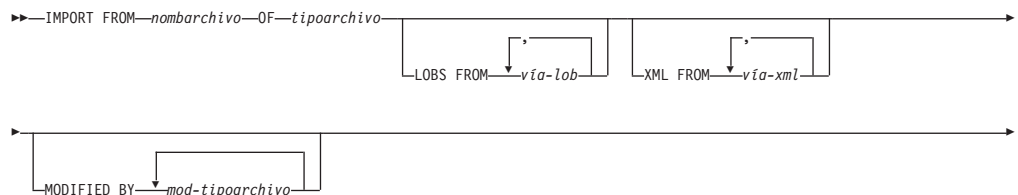
La etiqueta de la fila que se debe insertar, las credenciales LBAC del usuario, la definición de política de seguridad y las normas de LBAC determinan la etiqueta en la fila.

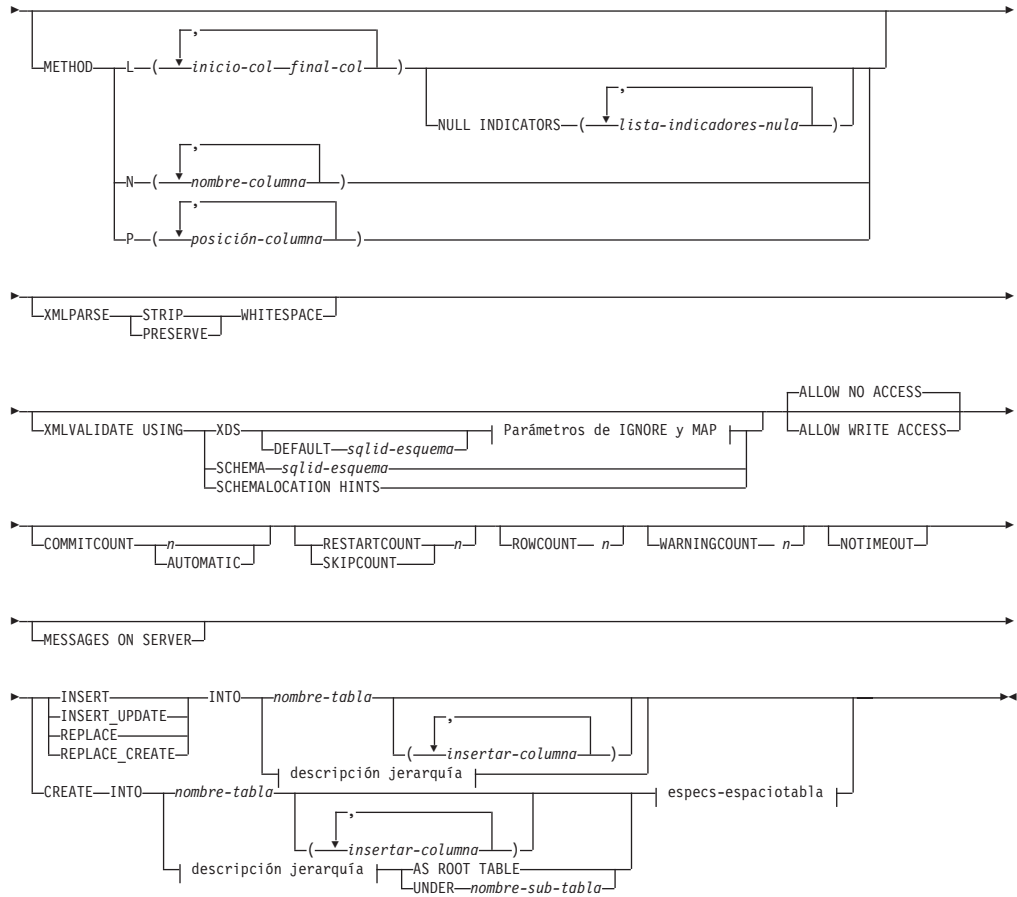
- Si se especifica la opción REPLACE o REPLACE\_CREATE, el ID de autorización de la sesión debe tener la autorización para descartar la tabla.

### Conexión necesaria

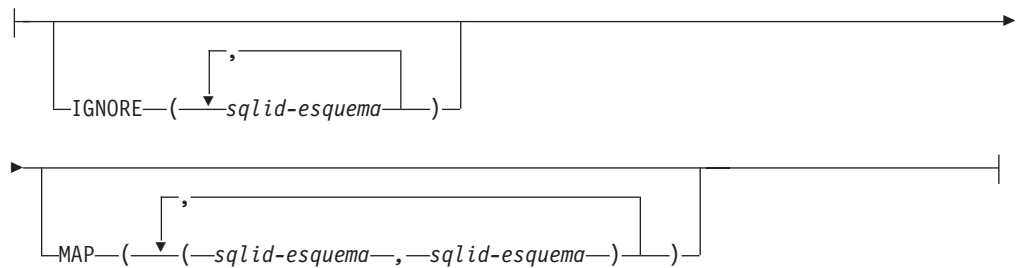
Base de datos. El acceso del programa de utilidad a servidores de bases de datos Linux, UNIX o Windows desde clientes Linux, UNIX o Windows debe ser una conexión directa a través del motor y no a través de un entorno de pasarela o de bucle de retorno de DB2 Connect.

### Sintaxis del mandato

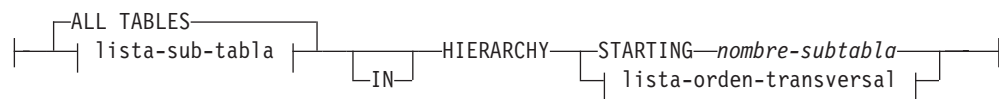




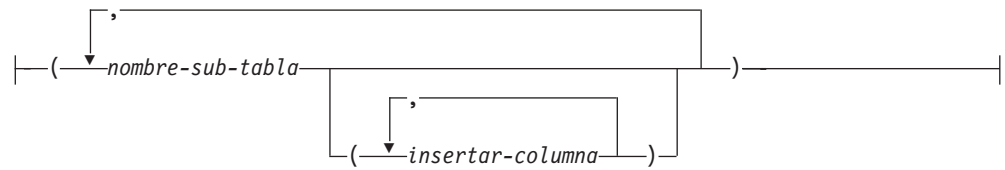
**Parámetros de IGNORE y de MAP:**



**descripción jerarquía:**



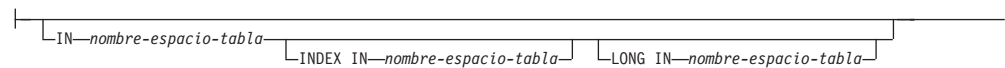
### lista-subtablas:



### lista-orden-transversal:



### especs-espaciotabla:



## Parámetros del mandato

### ALL TABLES

Palabra clave implícita solo para jerarquía. Cuando se importa una jerarquía, el valor por omisión es importar todas las tablas especificadas en el orden transversal.

### ALLOW NO ACCESS

Ejecuta la importación en modalidad fuera de línea. Antes de que se inserten filas, se adquiere un bloqueo exclusivo (X) sobre la tabla de destino. Esto evita que varias aplicaciones accedan simultáneamente a los datos de la tabla. Este es el comportamiento por omisión de la importación.

### ALLOW WRITE ACCESS

Ejecuta la importación en modalidad en línea. Cuando se inserta la primera fila, se adquiere un bloqueo de intento exclusivo (IX) sobre la tabla de destino. Esto permite que haya varios lectores y grabadores que accedan simultáneamente a los datos de la tabla. La modalidad en línea no es compatible con las opciones de importación REPLACE, CREATE o REPLACE\_CREATE. La modalidad en línea no se puede utilizar conjuntamente con las inserciones en almacenamiento intermedio. La operación de importación confirmará periódicamente datos insertados para evitar la escala de bloqueo en un bloqueo de tabla y para evitar el agotamiento del espacio de anotación cronológica activa. Estas confirmaciones se realizarán aunque no se haya utilizado la opción COMMITCOUNT. Durante cada confirmación, la importación perderá su bloqueo de tabla IX e intentará readquirirlo tras la confirmación. Este 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.

Si el mandato IMPORT encuentra un SQL0964C (archivo de anotaciones de transacción lleno) mientras inserta o actualiza un registro, y se ha especificado COMMITCOUNT  $n$ , el mandato IMPORT intentará resolver esta cuestión realizando una confirmación no condicional y luego reintentando insertar o actualizar el registro. Si así no se resuelve la condición de archivo de anotaciones lleno (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  $n$ . Se puede usar la opción RESTARTCOUNT o SKIPCOUNT para evitar que se procesen las filas ya confirmadas.

## CREATE

**Nota:** El parámetro CREATE ha quedado obsoleto y se puede eliminar en un futuro release. Para más detalles, consulte *Las opciones del programa de utilidad de importación 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** *nombarchivo*

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 base 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-tabla*

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 declarada temporal 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-tabla*

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 alternativas 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 correlación 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 SQL necesaria para recuperar todos los mensajes de aviso y de error producidos durante esta operación, y MSG\_REMOVAL, que es la sentencia 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 95.

## 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-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 de indicador NULL especifica que los datos de columna no son NULL y que se importarán 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 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.

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, solo 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 más detalles, consulte *Las opciones del programa de utilidad de importación 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 (XDS0 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 COMMIT después de cerrar todos los cursores abiertos WITH HOLD o emitiendo un 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 retroacción (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 tiene otros dependientes distintos de la propia tabla o una vista de objeto de 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 datos; 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 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/IX en clientes de DB2 en el sistema operativo AIX.

Para objetos de tabla en una página de 8 KB 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.

No se pueden importar datos a una tabla de sistema, a una tabla declarada temporal o a 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 utilizar 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 38. Conjunto de resultados devuelto por el mandato *IMPORT*

Nombre de columna	Tipo de datos	Descripción
ROWS_READ	BIGINT	Número de registros leídos en el archivo durante la importación.
ROWS_SKIPPED	BIGINT	Número de registros que se han saltado antes de que se empiece a insertar o actualizar.
ROWS_INSERTED	BIGINT	Número de filas insertadas en la tabla de destino.
ROWS_UPDATED	BIGINT	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).
ROWS_REJECTED	BIGINT	Número de registros que no se han podido importar.
ROWS_COMMITTED	BIGINT	Número de registros importados satisfactoriamente y confirmados en la base de datos.
MSG_RETRIEVAL	VARCHAR(512)	Sentencia 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>
MSG_REMOVAL	VARCHAR(512)	Sentencia 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 39. 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: <code>usedefaults</code>, <code>identitymissing</code>, <code>identityignore</code>, <code>generatedmissing</code> y <code>generatedignore</code>.</p>

Tabla 39. Modificadores de tipo de archivo válidos para el programa de utilidad de importación: Todos los formatos de archivo (continuación)

Modificador	Descripción
generatedignore	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.
generatedmissing	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.
identityignore	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.
identitymissing	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.
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 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>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 db2exp.001.7.-1/.</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.

Tabla 39. Modificadores de tipo de archivo válidos para el programa de utilidad de importación: Todos los formatos de archivo (continuación)

Modificador	Descripción
nodefaults	<p>Si no se especifica explícitamente una columna de origen 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 de origen para una de las columnas de tabla de destino, se produce una de las siguientes acciones:</p> <ul style="list-style-type: none"> <li>• Si se puede especificar un valor por omisión para una columna, se carga el valor por omisión</li> <li>• Si la columna es anulable y no se puede especificar un valor por omisión para dicha columna, se carga un NULL</li> <li>• 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.</li> </ul>
norowwarnings	Suprime todos los avisos sobre filas rechazadas.
rowchangetimestampignore	<p>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:</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.</p>
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><b>Nota:</b> 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>

Tabla 39. Modificadores de tipo de archivo válidos para el programa de utilidad de importación: Todos los formatos de archivo (continuación)

Modificador	Descripción
usedefaults	<p>Si se ha especificado una columna de origen para una columna de tabla de destino, pero dicha columna de origen 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"> <li>• 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.</li> <li>• Para archivos DEL/ASC/WSF: una fila que no tiene suficientes columnas o que no es suficientemente larga para la especificación original.</li> </ul> <p><b>Nota:</b> 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 la 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 de origen no contiene datos para una instancia de fila, se produce una de las acciones siguientes:</p> <ul style="list-style-type: none"> <li>• 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.</li> </ul>

Tabla 40. 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"> <li>• Para DBCS (gráfico) puro, DBCS mixto y EUC, los delimitadores están restringidos al rango de x00 a x3F, inclusive.</li> <li>• 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.</li> </ul> <p><b>Nota:</b></p> <ol style="list-style-type: none"> <li>1. El modificador codepage no se puede utilizar con el modificador lobsinfile.</li> <li>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.</li> </ol>

Tabla 40. Modificadores de tipo de archivo válidos para el programa de utilidad de importación: Formatos de archivo ASCII (ASC/DEL) (continuación)

Modificador	Descripción
dateformat="x"	<p>x es el formato de la fecha en el archivo fuente.<sup>2</sup> Los elementos de fecha válidos son:</p> <ul style="list-style-type: none"> <li>AAAA - Año (cuatro dígitos entre 0000 y 9999)</li> <li>M - Mes (uno o dos dígitos entre 1 y 12)</li> <li>MM - Mes (dos dígitos entre 1 y 12; se excluye mutuamente con M)</li> <li>D - Día (uno o dos dígitos entre 1 y 31)</li> <li>DD - Día (dos dígitos entre 1 y 31; se excluye mutuamente con D)</li> <li>DDD - Día del año (tres dígitos entre 001 y 366; se excluye mutuamente con otros elementos de día o mes)</li> </ul> <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>
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>
timeformat="x"	<p>x es el formato de la hora en el archivo fuente.<sup>2</sup> Los elementos de hora válidos son:</p> <ul style="list-style-type: none"> <li>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)</li> <li>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)</li> <li>M - Minuto (uno o dos dígitos entre 0 y 59)</li> <li>MM - Minuto (dos dígitos entre 0 y 59; se excluye mutuamente con M)</li> <li>S - Segundo (uno o dos dígitos entre 0 y 59)</li> <li>SS - Segundo (dos dígitos entre 0 y 59; se excluye mutuamente con S)</li> <li>SSSS - Segundo del día después de medianoche (5 dígitos entre 00000 y 86399; se excluye mutuamente con otros elementos de hora)</li> <li>TT - Indicador de meridiano (AM o PM)</li> </ul> <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> <pre>"HH:MM:SS" "HH.MM TT" "SSSS"</pre>

Tabla 40. 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.<sup>2</sup> Los elementos válidos de la indicación de fecha y hora son:</p> <ul style="list-style-type: none"> <li>AAAA - Año (cuatro dígitos entre 0000 y 9999)</li> <li>M - Mes (uno o dos dígitos entre 1 y 12)</li> <li>MM - Mes (dos dígitos entre 01 y 12; se excluye mutuamente con M y MMM)</li> <li>MMM - Mes (abreviación de tres letras no sensible a mayúsculas y minúsculas para el nombre del mes; se excluye mutuamente con M y MM)</li> <li>D - Día (uno o dos dígitos entre 1 y 31)</li> <li>DD - Día (dos dígitos entre 1 y 31; se excluye mutuamente con D)</li> <li>DDD - Día del año (tres dígitos entre 001 y 366; se excluye mutuamente con otros elementos de día o mes)</li> <li>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)</li> <li>HH - Hora (dos dígitos entre 0 y 12, para un sistema de 12 horas y 0 y 24 para uno de 24 horas; se excluye mutuamente con H)</li> <li>M - Minuto (uno o dos dígitos entre 0 y 59)</li> <li>MM - Minuto (dos dígitos entre 0 y 59; se excluye mutuamente con M, minuto)</li> <li>S - Segundo (uno o dos dígitos entre 0 y 59)</li> <li>SS - Segundo (dos dígitos entre 0 y 59; se excluye mutuamente con S)</li> <li>SSSSS - Segundo del día después de medianoche (5 dígitos entre 00000 y 86399; se excluye mutuamente con otros elementos de hora)</li> <li>UUUUUU - Microsegundo (6 dígitos entre 000000 y 999999; se excluye mutuamente con todos los otros elementos de microsegundo)</li> <li>UUUUU - Microsegundo (5 dígitos entre 00000 y 99999, se correlaciona con un rango de 000000 a 999990; se excluye mutuamente con todos los otros elementos de microsegundo)</li> <li>UUUU - Microsegundo (cuatro dígitos entre 0000 y 9999, se correlaciona con un rango de 000000 a 999900; se excluye mutuamente con todos los otros elementos de microsegundo)</li> <li>UUU - Microsegundo (3 dígitos entre 000 y 999 se correlaciona con un rango de 000000 a 999000; se excluye mutuamente con todos los otros elementos de microsegundo)</li> <li>UU - Microsegundo (2 dígitos entre 00 y 99, se correlaciona con un rango de 000000 a 990000; se excluye mutuamente con todos los otros elementos de microsegundo)</li> <li>U - Microsegundo (1 dígito entre 0 y 9, se correlaciona con un rango de 000000 a 900000; se excluye mutuamente con todos los otros elementos de microsegundo)</li> <li>TT - Indicador de meridiano (AM o PM)</li> </ul> <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="padding-left: 40px;">"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 style="padding-left: 40px;">db2 import from delfile2 of del modified by timestampformat="aaa.mm.dd hh:mm tt" insert into schedule</pre>

Tabla 40. 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><b>Restricciones</b></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><b>Nota:</b> 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 41. 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 el origen 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 $\backslash$ . <sup>3</sup>  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 32767. 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 42. 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. <sup>34</sup> 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 42. 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.<sup>34</sup></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.<sup>34</sup></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&lt;delimitador de fila&gt; "Vincent,&lt;delimitador de fila&gt;, is a manager", ... ... 4005,44.37&lt;delimitador de fila&gt;</pre> <p>Si se especifica el modificador delprioritychar, sólo habrá dos filas en este archivo de datos. El segundo &lt;delimitador de fila&gt; se interpretará como parte de la primera columna de datos de la segunda fila, mientras que el primero y tercero &lt;delimitadores de fila&gt; 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 &lt;delimitador de fila&gt;.</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 42. 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 43. 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 el origen 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 44. 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 44. 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.  <b>Aviso:</b> 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.  <b>Aviso:</b> 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 supone que los datos de caracteres están en la página de códigos N. Se supone que los datos gráficos están en la página de códigos gráfica de 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.  <b>Aviso:</b> 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 la 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:
  - `indexixf`
  - `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.
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 SYSADM o 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' )
```

## 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 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 para la tabla se define en el valor YES, los datos cargados estarán sujetos a la compresión en cada dato y partición de base de datos para los que ya existe un diccionario en la tabla.

Enlace rápido con “Modificadores de tipo de archivo para el programa de utilidad de carga” en la página 134.

## 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:

- *sysadm*
- *dbadm*
- Autorización de carga 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.

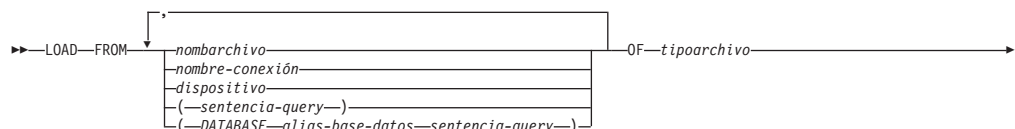
Dado que todos los procesos de carga (y todos los procesos del servidor DB2 en general) son propiedad del propietario de la instancia y todos estos procesos utilizan la identificación del propietario de la instancia para acceder a los archivos necesarios, el propietario de la instancia debe tener acceso 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.

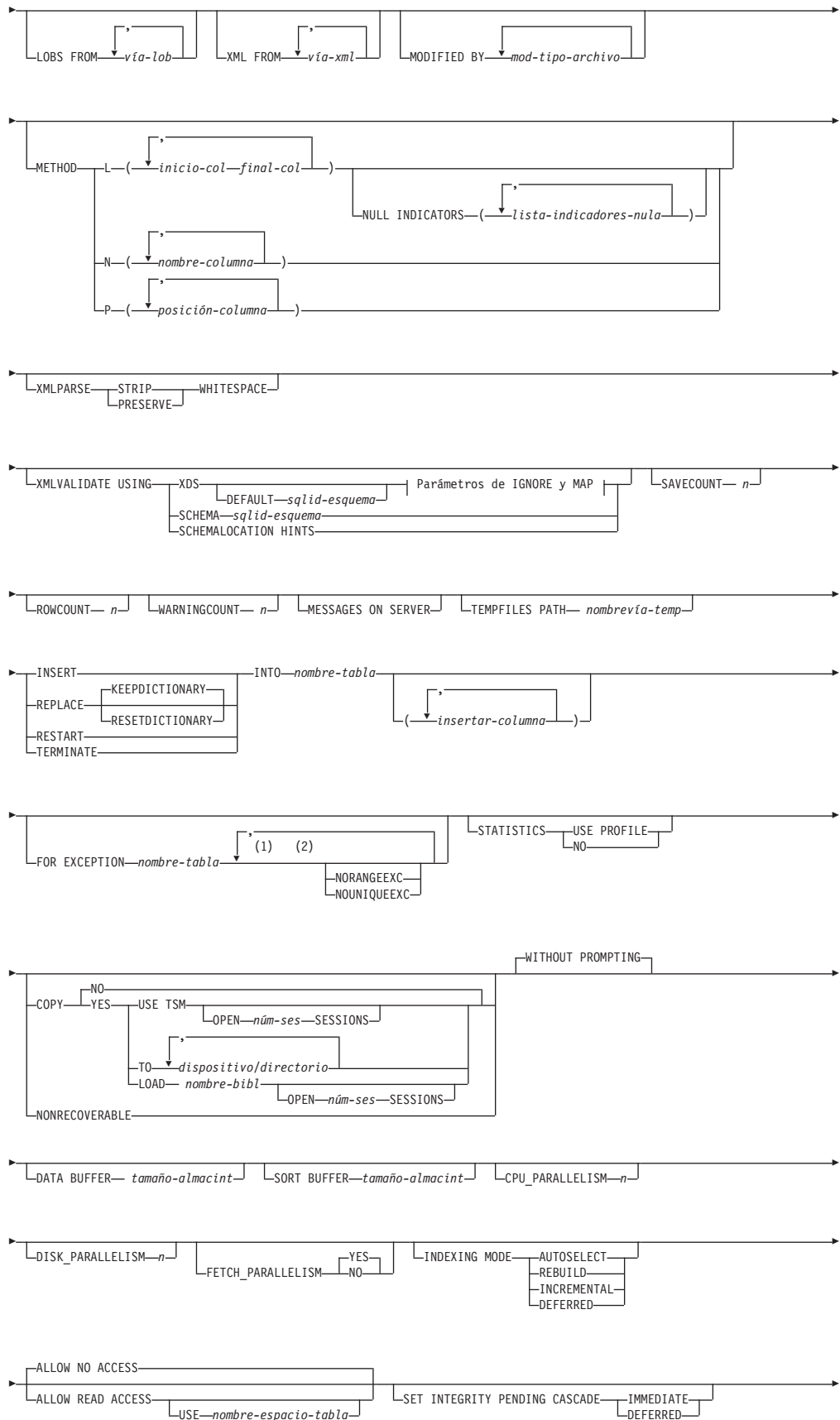
### Conexión necesaria

Base de datos.

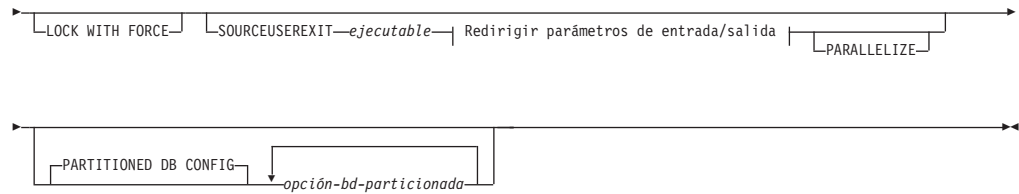
Instancia. No se necesita una conexión explícita. Si se ha establecido una conexión a la base de datos, se intenta una conexión implícita a la instancia local.

### Sintaxis del mandato

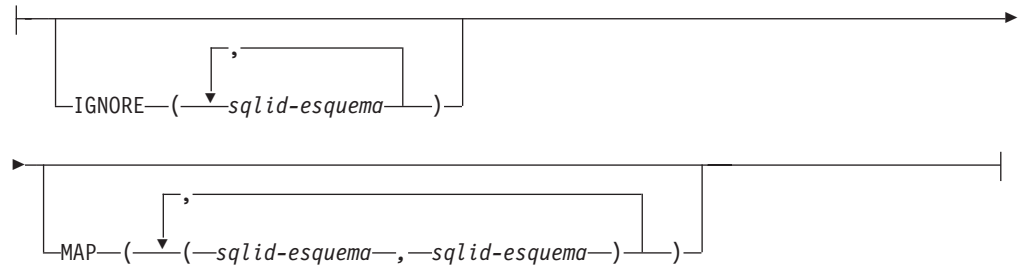




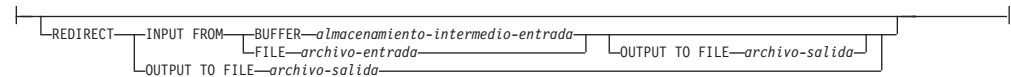




### Parámetros de IGNORE y de MAP:



### 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 hacen referencia a una sentencia SQL que contiene los datos que se están cargando o a la propia sentencia SQL y la base de datos fuente opcional para cargar 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 procedimientos 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.)

**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).

**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 134.

**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 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 correlación 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 se ignora cuando se especifica junto con el tipo de archivo CURSOR.

**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 SQL necesaria para recuperar todos los mensajes de aviso y de error producidos durante esta operación, y MSG\_REMOVAL, que es la sentencia 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 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 45. LOAD REPLACE KEEPDICTIONARY

Comprimir	Diccionario existe	Resultados
S	S	Conservar diccionario; todas las filas de entrada están sujetas a la compresión con el diccionario existente.
S	N	Insertar diccionario nuevo en la tabla únicamente si existen suficientes datos de usuario; las filas restantes están sujetas a la compresión tras crear el diccionario.
N	S	Conservar diccionario; no se comprimen todas las filas de entrada.
N	N	Sin efecto; no se comprimen todas las filas.

## 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 46. LOAD REPLACE RESETDICTIONARY

Comprimir	Diccionario existe	Resultados
S	S	Crear un diccionario nuevo*; las filas restantes que se van a cargar están sujetas a la compresión tras crear el diccionario.
S	N	Crear un diccionario nuevo; las filas restantes están sujetas a la compresión tras crear el diccionario.
N	S	Eliminar el diccionario; no se comprimen todas las filas.
N	N	Sin efecto; no se comprimen todas las filas.

\* Si hay un diccionario y el atributo de compresión está habilitado, pero no hay registros para cargar en la partición de tabla, no se puede crear un diccionario nuevo 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 tabla.

### 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 o una tabla declarada temporal. 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.

#### **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. Éste 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 se inhabilita la recuperación en avance (*logretain* y *userexit* están desactivados).

#### **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 compromiso 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.

Éste 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 paralelismo entre particiones. 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 haya restricciones en la tabla, el estado de la tabla se definirá en Establecer integridad pendiente. La sentencia SET INTEGRITY se debe utilizar para sacar la tabla fuera del estado Establecer integridad pendiente.

## **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 Establecer integridad pendiente, así como Carga en proceso y Acceso de lectura. Al final de la carga, se eliminará el estado de la tabla Carga en curso, pero los estados de la tabla Establecer integridad pendiente y Acceso de lectura permanecerán. La sentencia SET INTEGRITY debe utilizarse para sacar la tabla fuera del estado Establecer integridad pendiente. Mientras la tabla está en los estados Establecer integridad pendiente 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-tabla***

Si se vuelven a crear los índices, se crea una copia duplicada del índice en el espacio de tablas *nombre-espacio-tabla* 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 un estado Establecer integridad pendiente, la opción SET INTEGRITY PENDING CASCADE permite al usuario especificar si el estado Establecer integridad pendiente de la tabla cargada se transmite inmediatamente en cascada a todos los descendientes (incluidas las tablas de clave foránea descendientes y las tablas de consulta materializadas inmediatas descendientes y las tablas por fases inmediatas descendientes).

**IMMEDIATE**

Indica que el estado Establecer integridad pendiente se extiende inmediatamente a todas las tablas de clave foránea descendientes, las tablas de consulta materializadas inmediatas y las tablas por fases descendientes. Para una operación LOAD INSERT, el estado Establecer integridad pendiente no se extiende a las tablas de clave foránea aunque se especifique la opción IMMEDIATE.

Cuando, posteriormente, se comprueba si hay violaciones de restricciones (utilizando la opción IMMEDIATE CHECKED de la sentencia SET INTEGRITY), las tablas de clave foránea descendientes que se pusieron en estado Establecer integridad pendiente con acceso de lectura se pondrán en estado Establecer integridad pendiente sin acceso.

## DEFERRED

Indica que sólo la tabla cargada se colocará en el estado Establecer integridad pendiente. 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.

Las tablas de clave foránea descendientes podrían ponerse más tarde implícitamente en estado Establecer integridad pendiente cuando se comprueba si hay violaciones de restricciones en las tablas padre (utilizando la opción IMMEDIATE CHECKED de la sentencia SET INTEGRITY). Las tablas de consulta materializadas inmediatas y las tablas por fases inmediatas descendientes se colocarán implícitamente en el estado Establecer integridad pendiente cuando se comprueba si hay violaciones de integridad en una de sus tablas subyacentes. Se emitirá un aviso (SQLSTATE 01586) para indicar que las tablas dependientes se han colocado en estado Establecer integridad pendiente. Vea la sección de Notas de la sentencia SET INTEGRITY en la publicación Consulta de SQL para averiguar cuándo se pondrán las tablas descendientes en el estado Establecer integridad pendiente.

Si no se especifica la opción SET INTEGRITY PENDING CASCADE:

- Sólo la tabla cargada se pondrá en el estado Establecer integridad pendiente. El estado de las tablas de clave foránea descendientes, las tablas de consulta materializadas inmediatas descendientes y las tablas por fases inmediatas descendientes permanecerán sin cambios y, posteriormente, se pueden poner implícitamente en el estado Establecer integridad pendiente cuando se comprueba si hay violaciones de restricciones en la tabla cargada.

Si LOAD no pone la tabla de destino en el estado Establecer integridad pendiente, se omite 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 retorno = 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 retorno = 0

## Ejemplos de carga de datos de documentos XML

### Carga de datos de un documento 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 sobre 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 de referencia, 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 referencia o de comprobación se colocan en estado Establecer integridad pendiente. Las tablas de resumen que se definen con *REFRESH IMMEDIATE* y que dependen de tablas que se están cargando, también se colocan en estado Establecer integridad pendiente. Emita la sentencia *SET INTEGRITY* para sacar la tabla del estado Establecer integridad pendiente. 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.

## 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 47. Gestión de diccionario LOAD TERMINATE

Atributo COMPRESS (comprimir) tabla	¿El diccionario existe antes de LOAD?	TERMINATE: LOAD REPLACE KEEPDICTIONARY o LOAD INSERT	TERMINATE: LOAD REPLACE RESETDICTIONARY
YES	YES	Conservar diccionario existente.	No se ha conservado nada.
YES	NO	No se ha conservado nada.	No se ha conservado nada.
NO	YES	Conservar diccionario existente.	No se ha conservado nada.
NO	NO	No hacer nada.	No hacer nada.

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 48. Gestión de diccionario LOAD RESTART

Atributo COMPRESS (comprimir) tabla	¿El diccionario existe antes de LOAD (cargar) punto de coherencia?	RESTART: LOAD REPLACE KEEPDICTIONARY o LOAD INSERT	RESTART: LOAD REPLACE RESETDICTIONARY
YES	YES	Conservar diccionario existente.	Conservar diccionario existente.
YES	NO	Crear diccionario sujeto a ADC.	Crear diccionario.
NO	YES	Conservar diccionario existente.	Eliminar diccionario existente.
NO	NO	No hacer nada.	No hacer nada.

### 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 49: Conjunto de resultados para la operación de carga.
- Tabla 50 en la página 132: El conjunto de resultados 2 contiene información para cada partición de base de datos en una operación de carga de múltiples particiones.

Tabla 49. 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 49. 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 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 <code>MESSAGES ON SERVER</code>.</p>
MSG_REMOVAL	VARCHAR(512)	Sentencia 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 <code>MESSAGES ON SERVER</code>.</p>

Tabla 50. 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 50. 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. Es uno de los siguientes:</p> <ul style="list-style-type: none"> <li>• <b>LOADPENDING:</b> 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.</li> <li>• <b>NORMAL:</b> 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 Establecer integridad pendiente si es necesario realizar más proceso de restricciones, pero esta interfaz no comunica este estado.</li> <li>• <b>UNCHANGED:</b> 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.</li> </ul> <p><b>Nota:</b> 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"> <li>• FILE_TRANSFER</li> <li>• LOAD</li> <li>• LOAD_TO_FILE</li> <li>• PARTITIONING</li> <li>• PRE_PARTITIONING</li> </ul>

## Modificadores de tipo de archivo para el programa de utilidad de carga

Tabla 51. 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 <code>SAVECOUNT &gt; 0</code> , 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 <code>generatedmissing</code> o el modificador <code>generatedoverride</code> .
generatedmissing	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 <code>generatedignore</code> o el modificador <code>generatedoverride</code> .
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 reglas normales para estos tipos de columnas). Esto es útil al migrar datos desde otro sistema de bases de datos o al cargar una tabla de datos que se ha recuperado utilizando la opción <code>RECOVER DROPPED TABLE</code> en el mandato <code>ROLLFORWARD DATABASE</code>. Cuando se utilice este modificador, se rechazarán las filas sin datos o con datos NULL para una columna no anulable generada (<code>SQL3116W</code>). Cuando se utiliza este modificador, la tabla se colocará en estado Establecer integridad pendiente. Para sacar la tabla del estado Establecer integridad pendiente sin verificar los valores suministrados por el usuario, emita el siguiente mandato después de la operación de carga:</p> <pre>SET INTEGRITY FOR &lt; nombre-tabla &gt; GENERATED COLUMN IMMEDIATE UNCHECKED</pre> <p>Para sacar la tabla fuera del estado Establecer integridad pendiente y forzar la verificación de los valores suministrados por el usuario, emita el siguiente mandato después de la operación de carga:</p> <pre>SET INTEGRITY FOR &lt; nombre-tabla &gt; 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 <code>LOAD</code> convertirá automáticamente el modificador a <code>generatedignore</code> 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 <code>generatedmissing</code> o <code>generatedignore</code>.</p>
identityignore	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 <code>GENERATED ALWAYS</code> y <code>GENERATED BY DEFAULT</code> . Esto significa que, para las columnas <code>GENERATED ALWAYS</code> , no se rechazará ninguna fila. Este modificador no se puede utilizar con el modificador <code>identitymissing</code> o <code>identityoverride</code> .



Tabla 51. Modificadores de tipo de archivo válidos para el programa de utilidad de carga: Todos los formatos de archivo (continuación)

Modificador	Descripción
identitymissing	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.
identityoverride	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 bases 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.
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>

Tabla 51. Modificadores de tipo de archivo válidos para el programa de utilidad de carga: Todos los formatos de archivo (continuación)

Modificador	Descripción
lobsinfile	<p><i>vía-acceso-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 LOBSINFILE. 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>
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><i>x</i> 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 <i>x</i> 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>

Tabla 51. Modificadores de tipo de archivo válidos para el programa de utilidad de carga: Todos los formatos de archivo (continuación)

Modificador	Descripción
rowchangetimestampignore	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.
rowchangetimestampmissing	Si se especifica este modificador, el programa de utilidad presupone que el archivo de datos de entrada no contiene ningún datos 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.
rowchangetimestampoverride	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 bases 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.
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>

Tabla 51. Modificadores de tipo de archivo válidos para el programa de utilidad de carga: Todos los formatos de archivo (continuación)

Modificador	Descripción
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><b>Nota:</b> 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>
usedefaults	<p>Si se ha especificado una columna de origen para una columna de tabla de destino, pero dicha columna de origen 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"> <li>• 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.</li> <li>• 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 la hora o utilizando NULL INDICATOR para una columna de cualquier tipo para indicar que la columna es NULL.</li> </ul> <p>Sin esta opción, si la columna de origen no contiene datos para una instancia de fila, se produce una de las acciones siguientes:</p> <ul style="list-style-type: none"> <li>• 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.</li> </ul>

Tabla 52. Modificadores de tipo de archivo válidos para el programa de carga: 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 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"> <li>• Para DBCS (gráfico) puro, DBCS mixto y EUC, los delimitadores están restringidos al rango de x00 a x3F, inclusive.</li> <li>• 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.</li> <li>• 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.</li> </ul> <p>Esta opción no está soportada conjuntamente con el tipo de archivo CURSOR.</p>
dateformat=" <i>x</i> "	<p><i>x</i> es el formato de la fecha en el archivo fuente.<sup>1</sup> 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"  "AAAADD"</p>

Tabla 52. Modificadores de tipo de archivo válidos para el programa de carga: Formatos de archivo ASCII (ASC/DEL) (continuación)

Modificador	Descripción
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 data-bbox="570 415 1143 491">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><b>Nota:</b></p> <ol data-bbox="537 646 1425 982" style="list-style-type: none"> <li>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.</li> <li>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 comprometidos en disco no se puede saber con certeza y no se puede garantizar la coherencia después de un 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.</li> <li>3. Si el archivo especificado ya existe, no se volverá a crear, sino que se añadirá.</li> </ol>
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 data-bbox="537 1083 1425 1213" style="list-style-type: none"> <li>1. se utiliza junto con el modificador de tipo de archivo dumpfile</li> <li>2. el usuario dispone de privilegio SELECT sobre la tabla de destino de carga</li> <li>3. se emite en una partición de base de datos del servidor DB2 que reside en un sistema operativo UNIX.</li> </ol> <p>Si el archivo especificado ya existe, los permisos no se modificarán.</p>
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&gt;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>

Tabla 52. Modificadores de tipo de archivo válidos para el programa de carga: Formatos de archivo ASCII (ASC/DEL) (continuación)

Modificador	Descripción
timeformat="x"	<p>x es el formato de la hora en el archivo fuente.<sup>1</sup> Los elementos de hora válidos son:</p> <ul style="list-style-type: none"> <li>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)</li> <li>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)</li> <li>M - Minuto (uno o dos dígitos entre 0 y 59)</li> <li>MM - Minuto (dos dígitos entre 0 y 59; se excluye mutuamente con M)</li> <li>S - Segundo (uno o dos dígitos entre 0 y 59)</li> <li>SS - Segundo (dos dígitos entre 0 y 59; se excluye mutuamente con S)</li> <li>SSSSS - Segundo del día después de medianoche (5 dígitos entre 00000 y 86399; se excluye mutuamente con otros elementos de hora)</li> <li>TT - Indicador de meridiano (AM o PM)</li> </ul> <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 52. 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.<sup>1</sup> Los elementos válidos de la indicación de fecha y hora son:</p> <p>AAAA - Año (cuatro dígitos entre 0000 y 9999)</p> <p>M - Mes (uno o dos dígitos entre 1 y 12)</p> <p>MM - Mes (dos dígitos entre 01 y 12; se excluye mutuamente con M y MMM)</p> <p>MMM - Mes (abreviación de tres letras no sensible a mayúsculas y minúsculas para el nombre del mes; se excluye mutuamente con M y MM)</p> <p>D - Día (uno o dos dígitos entre 1 y 31)</p> <p>DD - Día (dos dígitos entre 1 y 31; se excluye mutuamente con D)</p> <p>DDD - Día del año (tres dígitos entre 001 y 366; se excluye mutuamente con otros elementos de día o mes)</p> <p>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)</p> <p>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)</p> <p>M - Minuto (uno o dos dígitos entre 0 y 59)</p> <p>MM - Minuto (dos dígitos entre 0 y 59; se excluye mutuamente con M, minuto)</p> <p>S - Segundo (uno o dos dígitos entre 0 y 59)</p> <p>SS - Segundo (dos dígitos entre 0 y 59; se excluye mutuamente con S)</p> <p>SSSSS - Segundo del día después de medianoche (5 dígitos entre 00000 y 86399; se excluye mutuamente con otros elementos de hora)</p> <p>UUUUUU - Microsegundo (6 dígitos entre 000000 y 999999; se excluye mutuamente con todos los otros elementos de microsegundo)</p> <p>UUUUU - Microsegundo (5 dígitos entre 00000 y 99999, se correlaciona con un rango de 000000 a 999990; se excluye mutuamente con todos los otros elementos de microsegundo)</p> <p>UUUU - Microsegundo (cuatro dígitos entre 0000 y 9999, se correlaciona con un rango de 000000 a 999900; se excluye mutuamente con todos los otros elementos de microsegundo)</p> <p>UUU - Microsegundo (3 dígitos entre 000 y 999 se correlaciona con un rango de 000000 a 999000; se excluye mutuamente con todos los otros elementos de microsegundo)</p> <p>UU - Microsegundo (2 dígitos entre 00 y 99 se correlaciona con un rango de 000000 a 990000; se excluye mutuamente con todos los otros elementos de microsegundo)</p> <p>U - Microsegundo (1 dígito entre 0 y 9 se correlaciona con un rango de 000000 a 900000; se excluye mutuamente con todos los otros elementos de microsegundo)</p> <p>TT - Indicador de meridiano (AM o PM)</p>



Tabla 52. 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> <pre> "AAAA/MM/DD HH:MM:SS.UUUUUU" </pre> <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> <pre> AAAA-MM-DD-HH.MM.SS AAAA-MM-DD HH:MM:SS </pre> <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 dos formatos, si incluye el campo de microsegundos (UUUUU), el programa de utilidad de carga prevé 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><b>Restricciones</b></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 52. 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 53. 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"> <li>• No se realiza ninguna conversión entre tipos de datos, a excepción de BIGINT, INTEGER y SMALLINT.</li> <li>• Las longitudes de los datos deben coincidir con las definiciones de las columnas de destino.</li> <li>• Los FLOAT deben estar en formato de Coma flotante IEEE.</li> <li>• 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.</li> </ul> <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 53. Modificadores de tipo de archivo válidos para el programa de carga: Formatos de archivo ASC (ASCII no delimitado) (continuación)

Modificador	Descripción
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 el origen 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 $\backslash$ . <sup>2</sup>  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, entonces n se reconoce también como un indicador NULL.
packeddecimal	Carga directamente datos decimales empaquetados, dado 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 reclen.  Los valores soportados para la porción de signo son: + = 0xC 0xA 0xE 0xF - = 0xD 0xB  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.  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.  Este modificador no se puede utilizar con el modificador implieddecimal.
reclen=x	$x$ 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.
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.  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 53. 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> <p>+ = 0xC 0xA 0xE 0xF - = 0xD 0xB</p> <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 54. 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.<sup>23</sup> Si desea especificar explícitamente las comillas dobles (") como delimitador de serie de caracteres, debe especificarse de la manera siguiente:</p> <p>modified by chardel""</p> <p>También se pueden especificar las comillas simples (') como delimitador de serie de caracteres del modo siguiente:</p> <p>modified by chardel''</p>
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.<sup>23</sup></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.<sup>23</sup></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> <p>db2 load ... modified by delprioritychar ...</p> <p>Por ejemplo, dado el archivo de datos DEL siguiente:</p> <p>"Smith, Joshua",4000,34.98&lt;delimitador de fila&gt; "Vincent,&lt;delimitador de fila&gt;, is a manager", ... ... 4005,44.37&lt;delimitador de fila&gt;</p> <p>Si se especifica el modificador delprioritychar, sólo habrá dos filas en este archivo de datos. El segundo &lt;delimitador de fila&gt; se interpretará como parte de la primera columna de datos de la segunda fila, mientras que el primero y tercero &lt;delimitadores de fila&gt; 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 &lt;delimitador de fila&gt;.</p>

Tabla 54. 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 55. 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	<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 el origen 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>

**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 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 la 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, \).

- 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 56. 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.  <b>Aviso:</b> 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 56. 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><b>Aviso:</b> Los datos gráficos se corromperán cuando se carguen en una base de datos de un solo byte.</p>
Presente	Presente	<p>Se supone que los datos de caracteres están en la página de códigos N. Se supone que los datos gráficos están en la página de códigos gráfica de 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><b>Aviso:</b> 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

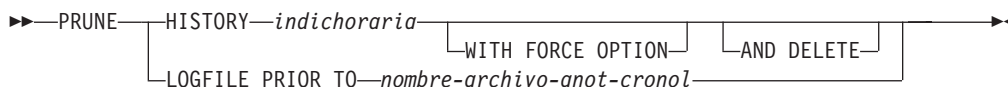
Una de las siguientes:

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

### Conexión necesaria

Base de datos

### Sintaxis del mandato



## Parámetros del mandato

### HISTORY *indicahoraria*

Identifica un rango de entradas del archivo histórico de recuperación que se suprimirán. Se puede especificar una indicación horaria completa (en el formato *aaaammddhhmmss*) o un prefijo inicial (mínimo *aaaa*). Todas las entradas con indicaciones de la hora iguales o menores que la indicación horaria proporcionada se suprimirán del archivo histórico de recuperación.

### WITH FORCE OPTION

Especifica que se podarán las entradas de acuerdo con la indicación horaria 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 necesiten. 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:

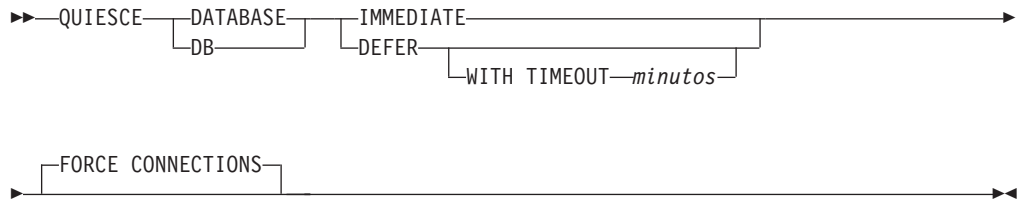
Para inmovilización a nivel de base de datos:

- *sysadm*
- *dbadm*
- *sysadm*
- *sysctrl*

## Conexión necesaria

Base de datos

## Sintaxis del mandato



## Parámetros del mandato

### DEFER

Espera a las aplicaciones hasta que comprometen la unidad de trabajo actual.

### WITH TIMEOUT

Especifica un período de tiempo, en minutos, que se deberá esperar para que las aplicaciones comprometan 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 comprometan 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 que todos los usuarios queden desconectados de la base de datos si se suministra la opción **FORCE CONNECTION**. **FORCE CONNECTION** es el comportamiento por omisión; el parámetro está permitido en el mandato por motivos de compatibilidad.
- El mandato se sincronizará con **FORCE** y solo se completará una vez que **FORCE** 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 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:

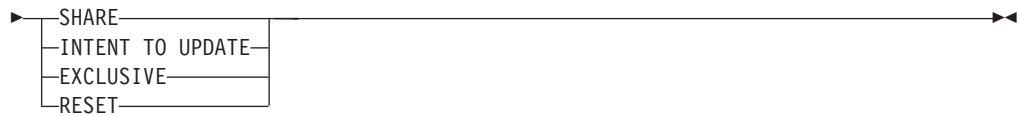
- *sysadm*
- *sysctrl*
- *sysmaint*
- *dbadm*
- *load*

## Conexión necesaria

Base de datos

## Sintaxis del mandato

```
►► QUIESCE TABLESPACES FOR TABLE nombretabla
                                     └─esquema.nombretabla─┘
```



## Parámetros del mandato

### TABLE

#### **nombretabla**

Especifica el nombre de tabla no calificado. La tabla no puede ser una tabla de catálogos del sistema.

#### **esquema.nombretabla**

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 compromete 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 perdura las anomalías de las transacciones, las anomalías de conexión e incluso las anomalías del sistema (por ejemplo anomalía de alimentación o rearranque).

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. Se puede especificar la distribución actual de datos, tanto si es uniforme como si está desviada. El algoritmo de redistribución selecciona las particiones que se deben mover basándose en la distribución actual de datos.

### Ámbito

Este mandato afecta a todas las particiones de base de datos del grupo de particiones de base de datos.

### Autorización

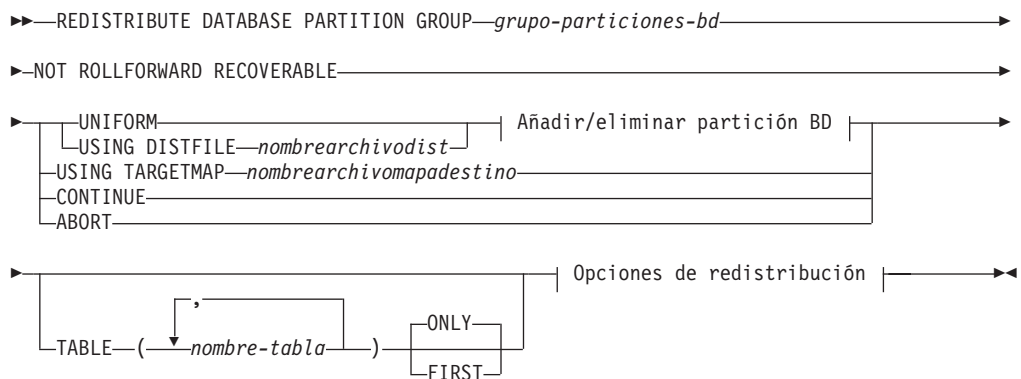
Una de las siguientes:

- *sysadm*
- *sysctrl*
- *dbadm*

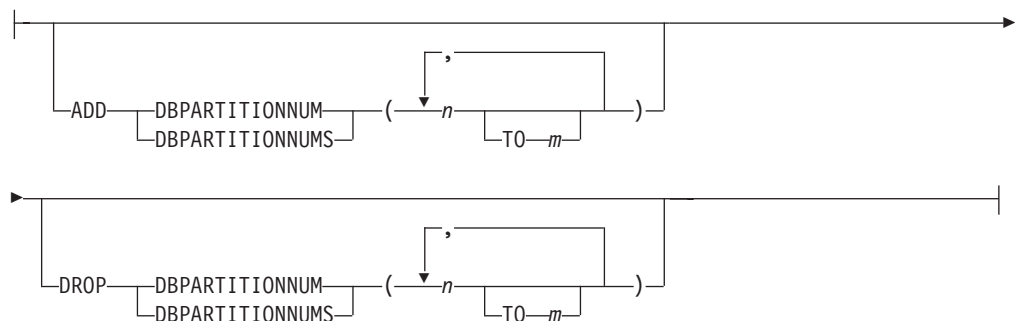
### Conexión necesaria

Conexión a la partición del catálogo.

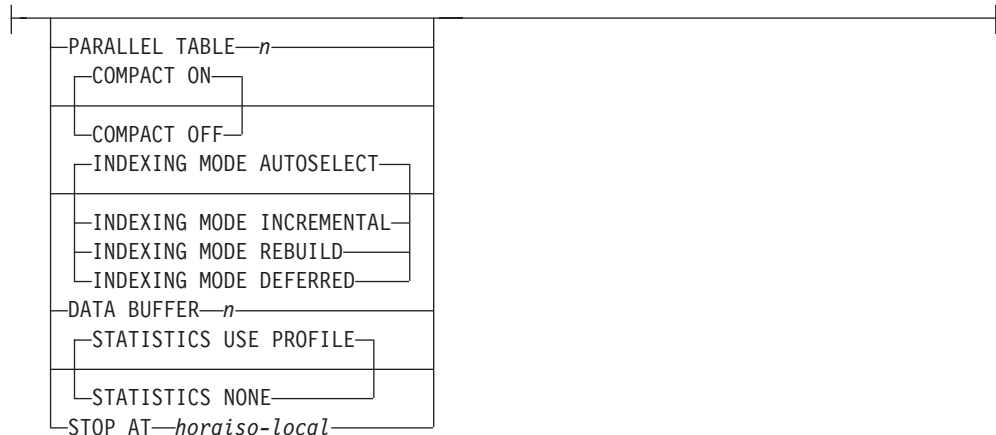
### Sintaxis del mandato



### 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 IBMCATGROUP y IBMTEMPGROUP no se pueden redistribuir.

### NOT ROLLFORWARD RECOVERABLE

Recordatorio importante que indica que el mandato REDISTRIBUTE DATABASE PARTITION GROUP no es recuperable en avance.

### 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 *nombreachivodist*

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 *nombreachivodist* para indicar la distribución actual de los datos entre las 4.096 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 *nombreachivodist*, 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 *nombearchivodist* debe contener 4.096 valores enteros positivos en formato carácter. La suma de los valores debe ser igual o menor que 4 294 967 295.

Hay que incluir el nombre de vía de acceso completo del *nombearchivodist*, y el *nombearchivodist* debe existir en el servidor y ser accesible desde la partición conectada.

#### **USING TARGETMAP** *nombearchivomapadestino*

El archivo especificado en *nombearchivomapadestino* 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 *nombearchivomapadestino* y el *nombearchivomapadestino* 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. Éste 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.

**PARALLEL TABLE** *n*

Especifica el número de tablas a redistribuir en paralelo. Hasta ulterior aviso, cualquier número entrado se omitirá y se utilizará el valor por omisión de 1.

**INDEXING MODE**

Este parámetro especifica cómo se mantienen los índices durante la redistribución. Los valores válidos son:

**AUTOSELECT**

El mandato Redistribute seleccionará automáticamente entre utilizar las modalidades INCREMENTAL y REBUILD basándose en la cantidad de datos que han de moverse. Ésta es la modalidad por omisión.

**INCREMENTAL**

Los índices se ampliarán incrementalmente. Por lo tanto, los índices deben ser válidos.

**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 mínimo arquitectónico, 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 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.

### COMPACT

Esta opción especifica si el programa de utilidad debería rellenar los huecos de las páginas de datos encontrados durante el proceso.

**ON** Para cualquier tabla específica, las particiones de base de datos que contienen datos de la tabla antes de que empiece la redistribución de los datos, tendrá huecos en las páginas rellenos a medida que la operación de redistribución prosigue. Para tablas normales, se tomarán registros del final lógico de la tabla, y la tabla se truncará al final para liberar espacio. Para tablas MDC, los registros se tomarán de la misma celda para rellenar los huecos y se liberarán los bloques vacíos para que sean reutilizados. Para tablas particionadas por rangos, se tomarán registros del final lógico de un rango y se llevará a cabo un truncamiento en cada rango para liberar espacio. El valor por omisión es ON.

**OFF** El programa de utilidad no rellenará huecos en páginas de datos. Se puede utilizar el mandato `REORG INDEXES/TABLE` para reclamar el espacio después de la redistribución.

### STOP AT *horaiso-local*

Cuando se especifica esta opción, el programa de utilidad comparará la *horaiso-local* con la indicación de fecha y hora local actual antes de empezar a trabajar en cada tabla. Si la *horaiso-local* es igual o menor que la indicación de fecha y hora local actual, el programa de utilidad se detendrá con un mensaje de aviso. Como resultado, todas las tablas hechas anteriormente utilizarán el mapa de distribución de reequilibrio y las tablas restantes continuarán usando el mapa de distribución definido en el grupo de particiones de bases de datos. Las tablas restantes se pueden redistribuir utilizando la opción `CONTINUE`. Este valor de *hora-local* se especifica como indicación horaria, 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. Esto es el valor por omisión.

### NONE

No se recopilarán estadísticas para la tablas.

## La operación de redistribución no es recuperable en avance

El mandato REDISTRIBUTE DATABASE PARTITION GROUP utiliza una estrategia mínima de registro cronológico que evita, siempre que ello sea posible, grabar registros 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 bases de datos recuperables, el programa de utilidad REDISTRIBUTE DATABASE PARTITION GROUP coloca todos los espacios de tablas afectados en el estado BACKUP PENDING, obligando al usuario a realizar una 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 un aspecto muy importante como consecuencia 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 fuera del 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/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 de datos actual mediante un archivos de distribución de datos, `distfile_for_dbpg_1`, y moviendo 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 sobre uso

- La primera vez que el programa de utilidad accede a un espacio de tablas, se coloca en el 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:
  - El directorio `homeinst/sqllib/redistribute` en sistemas basados en Linux y UNIX, utilizando el siguiente formato para subdirectorio y nombre de archivo: *nombre-basedatos.nombre-grupo-particiones-basedatos.indichoraria*.
  - El directorio `DB2INSTPROF\instancia\redistribute` en los sistemas operativos Windows (donde DB2INSTPROF es el valor de la variable de registro DB2INSTPROF), utilizando el siguiente formato para subdirectorios y nombre de archivo: *nombre-basedatos.nombre-grupo-particiones-basedatos.indichoraria*.
  - El valor de indicación de la hora es la hora en la que se ha emitido el mandato.

Para obtener más información sobre el registro de sucesos de redistribución, consulte el tema *Recuperación de errores 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 en el grupo de particiones de base de datos.
- La opción PARALLEL TABLE permite que la redistribución funcione simultáneamente en varias tablas del mismo grupo de particiones de bases de datos. No obstante, no se da soporte al lanzamiento de varias peticiones de redistribución simultáneamente en el mismo grupo de particiones de bases de datos y se devuelve -6056.
- 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 *nombreachivomapadestino*, 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. Si encuentra datos en la partición de la base de datos equivocada como resultado de una operación de redistribución, tiene dos opciones. Puede:
  1. descargar la tabla, descartarla y, a continuación, volver a cargarla, o
  2. utilizar una correlación de destino nueva para volver a redistribuir el grupo de particiones de base de datos.

## Compatibilidades

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 se puede utilizar para los índices en tablas temporales declaradas (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

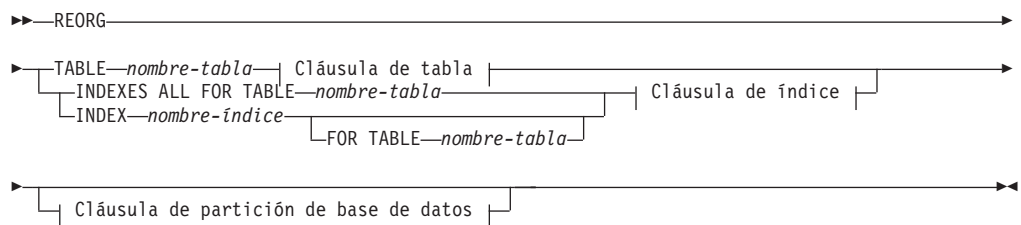
Una de las siguientes:

- *sysadm*
- *sysctrl*
- *sysmaint*
- *dbadm*
- Privilegio CONTROL en la tabla.

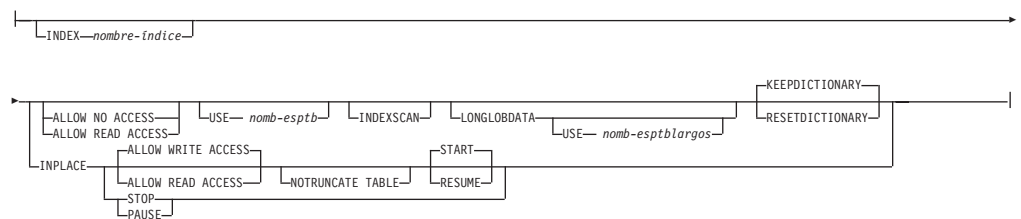
## Conexión necesaria

Base de datos

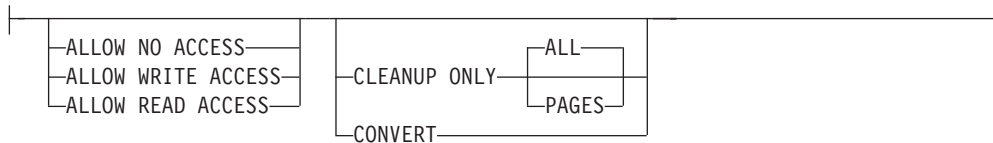
## Sintaxis del mandato



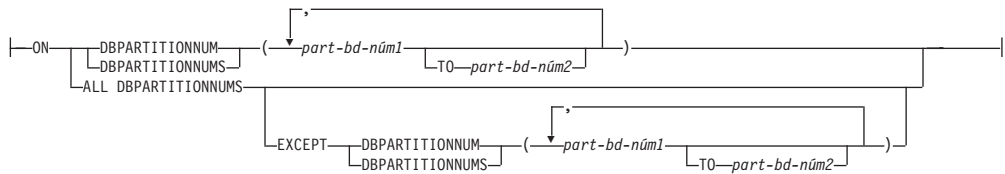
## Cláusula de tabla:



## Cláusula de índice:



### 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 índices individuales SOLO se puede usar 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.

#### ALLOW NO ACCESS

Especifica que ningún otro usuario puede acceder a la tabla mientras los índices se están reorganizando.

#### ALLOW READ ACCESS

Especifica que otros usuarios pueden tener acceso solo de lectura a la tabla mientras se están reorganizando los índices. Este nivel de acceso no está soportado para los REORG INDEXES de una tabla particionada a menos que se especifique la opción CLEANUP ONLY.

#### ALLOW WRITE ACCESS

Especifica que otros usuarios pueden leer y grabar en la tabla mientras se están reorganizando los índices. Este nivel de acceso no está soportado para tablas de clústeres de múltiples dimensiones (MDC), índices ampliados o tablas que contienen una columna con el tipo de datos XML a menos que se especifique la opción CLEANUP ONLY.

Cuando no se especifica ninguna modalidad ACCESS, se le elegirá una del siguiente modo:



Tabla 57. 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	Parámetros adicionales especificados para la cláusula de índice	Modalidad de acceso por omisión
REORG INDEXES	tabla no particionada	cualquiera	ALLOW READ ACCESS
REORG INDEXES	tabla particionada	no se ha especificado ninguno	ALLOW NO ACCESS
REORG INDEXES	tabla particionada	se ha especificado CLEANUP ONLY	ALLOW READ ACCESS
REORG INDEX	tabla particionada	cualquiera	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 comprometidas. Una página pseudovacia comprometida es aquella en que todas las claves de la página están marcadas como suprimidas y se sabe que todas estas supresiones están comprometidas. El número de páginas pseudovacías en 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 el compromiso.

La opción CLEANUP ONLY ALL liberará las páginas pseudovacías comprometidas, y eliminará las claves pseudosuprimidas comprometidas 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 en un índice, excluidas las que están en páginas pseudovacías, puede determinarse ejecutando runstats y, a continuación, seleccionando NUMRIDS DELETED de SYSCAT.INDEXES. La opción ALL limitará NUMRIDS DELETED y NUM EMPTY LEAFs si están determinadas para el compromiso.

**ALL** Especifica que los índices deben limpiarse eliminando las claves pseudosuprimidas comprometidas y las páginas pseudovacías comprometidas.

### PAGES

Especifica que las páginas pseudovacías comprometidas 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

Si no está seguro de si la tabla en la que está operando tiene un índice de tipo 1 o 2, pero desea índices de tipo 2, puede utilizar la opción CONVERT. Si el índice es de tipo 1, esta opción lo convertirá en tipo 2. Si el índice ya es de tipo 2, esta opción no tiene ningún efecto.

Todos los índices creados por DB2 antes de la Versión 8 son índices de tipo 1. Todos los índices creados por la Versión 8 son índices de tipo 2, salvo cuando se crea un índice en una tabla que ya tiene un índice de tipo 1. En este caso, el nuevo índice también será de tipo 1.

Utilizar el mandato INSPECT para determinar el tipo de índice puede ser una tarea lenta. CONVERT le permite asegurarse de que el nuevo índice será de Tipo 2, sin que sea necesario determinar su tipo original.

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*

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 se 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. Este es el comportamiento por

omisión cuando se reorganiza una tabla particionada. La reorganización de una tabla particionada se produce fuera de línea.

#### **ALLOW READ ACCESS**

Permite solo el acceso de lectura a la tabla durante la reorganización. Este es el valor por omisión para una tabla no particionada.

#### **INPLACE**

Reorganiza la tabla al tiempo que permite el acceso de los usuarios.

En la reorganización de tablas in situ sólo se permite en tablas no particionadas y no tablas MDC con índices de tipo 2, pero sin índices ampliados y sin índices definidos en 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. Éste 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 tabla, 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.

En el caso de las tablas particionadas, el espacio de tablas se usa como almacenamiento temporal para la reorganización de todas las particiones de datos de la tabla. Al reorganizar una tabla particionada, las particiones de datos se reorganizan de una en una. 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.

#### **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 de 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 reorganizada tendrán un formato no comprimido. No es posible comprimir objetos largos, LOB, de índice o XML.

Tabla 58. REORG KEEPDICTIONARY

Comprimir	Diccionario existe	Resultado; consecuencia
Y	Y	Conservar diccionario; filas comprimidas
Y	N	Crear diccionario; filas comprimidas
N	Y	Conservar diccionario; todas las filas sin comprimir
N	N	Sin efecto; todas las filas comprimidas

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 tiene un diccionario de compresión existente, el proceso de reorganización eliminará el diccionario, y todas las filas de la tabla que se acabada de reorganizar tendrán un formato no comprimido. No es posible comprimir objetos largos, LOB, de índice o XML.

Tabla 59. REORG RESETDICTIONARY

Comprimir	Diccionario existe	Resultado; consecuencia
Y	Y	Crear diccionario nuevo*; filas comprimidos
Y	N	Crear diccionario nuevo; filas comprimidos
N	Y	Eliminar diccionario; todas las filas sin comprimir
N	N	Sin efecto; todas las filas comprimidas

\* - 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.

### 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; esta sentencia, en el caso de las 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 puede usar para tablas temporales declaradas.
- El mandato REORG TABLE no se puede usar en las vistas.
- La reorganización de una tabla no es compatible con las tablas agrupadas en clústeres 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:

- REORG está soportado a nivel de tabla. La reorganización de una partición de datos individual se puede conseguir desenlazando la partición de datos, reorganizando la tabla no particionada obtenida y luego volviendo a enlazar la partición de datos.
- La tabla debe tener una ACCESS\_MODE en SYSCAT.TABLES igual a Acceso Completo.
- La reorganización se salta las particiones de datos que estén en estado restringido debido a una operación de enlazar o desenlazar.
- Si se produce un error, los índices no particionados de la tabla se marcarán como no válidos y se volverán a crear la próxima vez que se acceda a la tabla.
- Si falla una operación de reorganización, algunas particiones de datos habrán quedado en estado reorganizado y otras no. Cuando se vuelve a emitir el mandato REORG TABLE, se reorganizarán todas las particiones de datos, sea cual sea el estado de reorganización que tengan.
- Al reorganizar índices en tablas particionadas, conviene que realice una operación runstats al final de una limpieza asíncrona de índices para generar estadísticas de índices precisas en presencia de las particiones de datos desenlazadas. Para determinar si en la tabla hay o no particiones de datos desenlazadas, puede consultar el campo de estado contenido en SYSDATAPARTITIONS y buscar el valor "I" (limpieza de índices) o el valor "D" (desenlazado con MQT dependiente).

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 REORGCHK como ayuda para determinar si un índice necesita una reorganización. 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 COMMIT después de cerrar todos los cursores abiertos WITH HOLD o emitiendo un ROLLBACK.

Es posible que los índices no sean óptimos tras una operación REORG TABLE in situ, dado que sólo se reorganiza el objeto de datos y no los índices. Se recomienda ejecutar un mandato REORG INDEXES después de una operación REORG TABLE

in situ. No obstante, los índices se reconstruyen por completo durante la última fase de un REORG TABLE clásico, en cuyo caso no es necesario reorganizar índices.

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 REORGCHK para determinar si una tabla se tiene que reorganizar. 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 mandato COMMIT después de cerrar todos los cursores abiertos WITH HOLD o emitiendo un mandato 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.

### Autorización

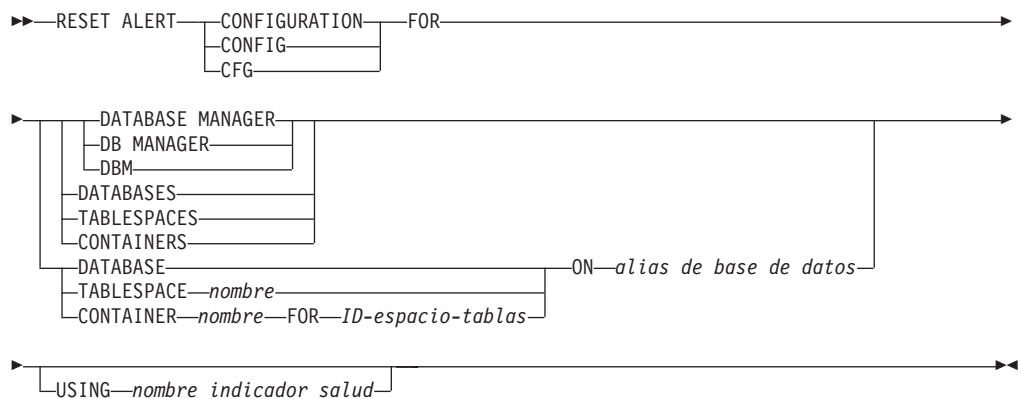
Una de las siguientes:

- sysadm
- sysmaint
- sysctrl

### Conexión necesaria

Base de datos.

### Sintaxis del mandato



### Parámetros del mandato

#### DATABASE MANAGER

Restablece los valores de alerta para el gestor de bases de datos.

#### 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 mediante la cláusula `DATABASE ON alias basedatos`.

#### 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 mediante la cláusula `"CONTAINER nombre ON alias basedatos"`.

#### CONTAINER nombre FOR ID-espacio-tablas FOR ID-espacio-tablas ON alias basedatos

Restablece los valores de alerta para el contenedor de espacio de tablas

denominado *nombre*, para el espacio de tablas especificado utilizando la cláusula "FOR ID-espacio-tablas", 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.

#### **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 mediante la cláusula "TABLESPACE *nombre* ON *alias basedatos*".

#### **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* ON *alias basedatos***

Restablece 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, 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

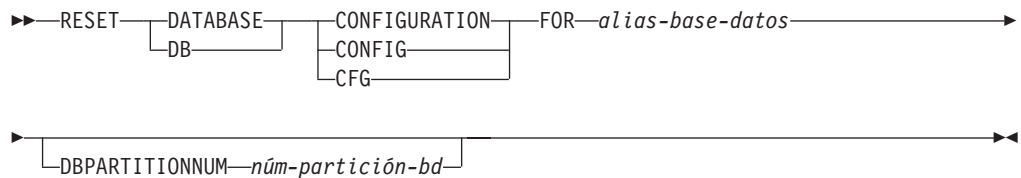
Una de las siguientes:

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

## 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 anteriores de la base de datos, donde el valor de `AUTO_RUNSTATS` será `ON`. `SELF_TUNING_MEMORY` 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 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

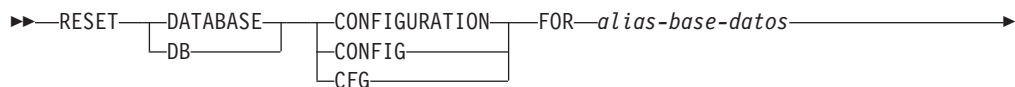
Una de las siguientes:

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

### 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 anteriores de la base de datos, donde el valor de AUTO\_RUNSTATS será ON. SELF\_TUNING\_MEMORY 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 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' )
```

### 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 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*
- 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*
- 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:

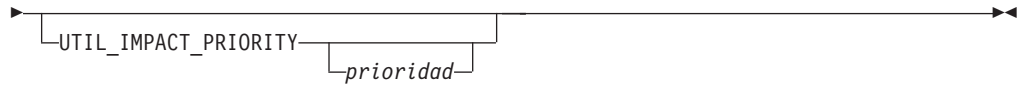
- *sysadm* o *dbadm*
- CONTROL
- SELECT

## Conexión necesaria

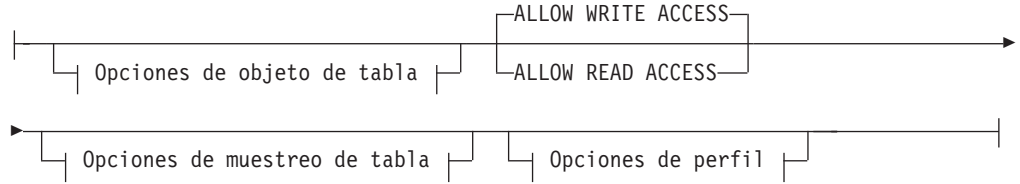
Base de datos

## Sintaxis del mandato

```
►►—RUNSTATS—ON TABLE—nombre-objeto—  
|—USE PROFILE—  
|—UNSET PROFILE—  
| Opciones de estadísticas |
```



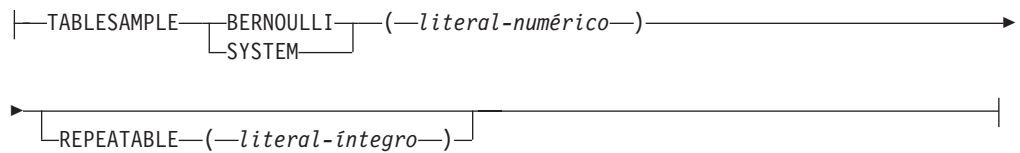
**Opciones de estadísticas:**



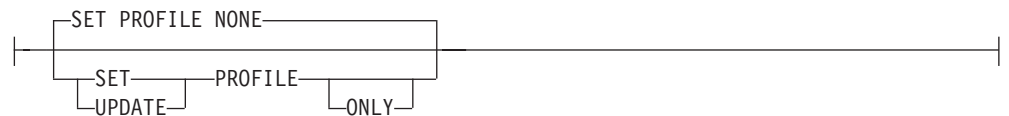
**Opciones de objeto de tabla:**



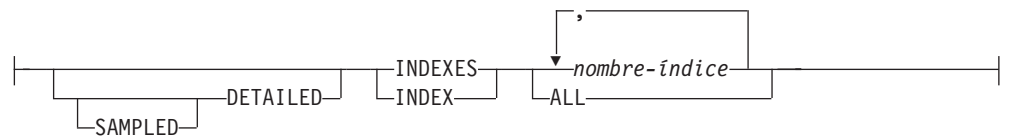
**Opciones de muestreo de tabla:**



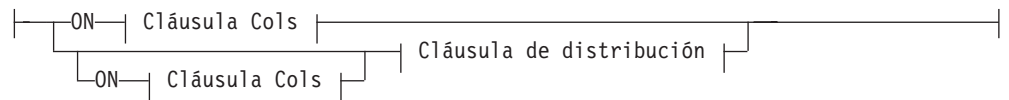
**Opciones de perfil:**



**Cláusula Index:**

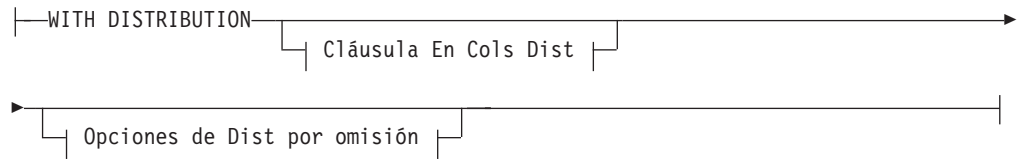


**Cláusula de estadísticas de columna:**

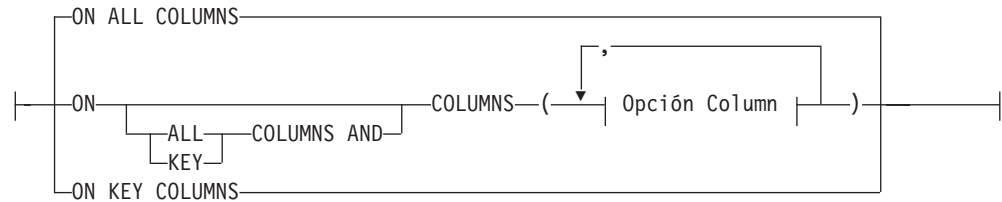


**Cláusula de distribución:**

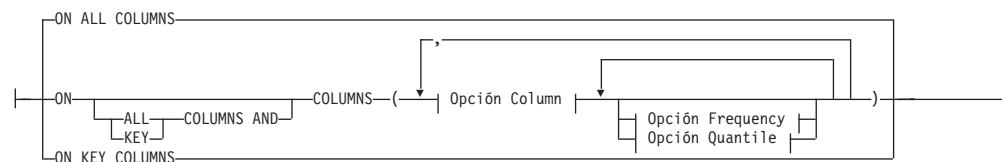




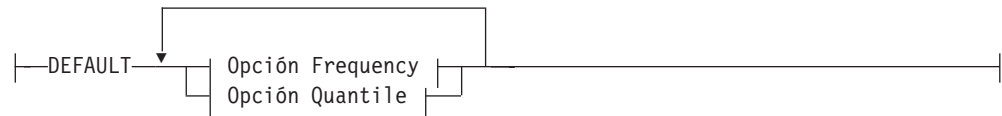
**Cláusula En cols:**



**Cláusula En Cols Dist:**



**Opción de Dist por omisión:**



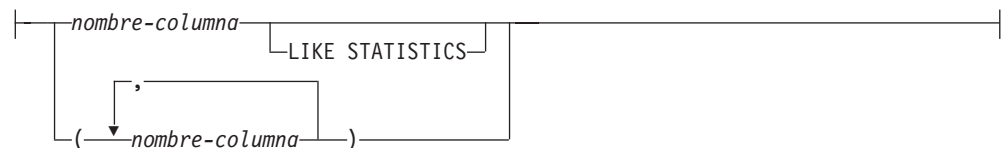
**Opción de frecuencia:**



**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 totalmente calificado o 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**

La recopilación de estadísticas puede realizarse en algunas columnas, pero no en otras. Columnas tales como LONG VARCHAR o CLOB no son elegibles. Si se desea reunir estadísticas sobre todas las columnas elegibles, se puede utilizar 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. Solo se recopilan para las columnas de tipo serie, y el optimizador de consulta las utiliza 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 las vistas estadísticas, el muestreo de `SYSTEM` está restringido a una clase específica de vistas. Son vistas que acceden a una sola tabla base o a un solo apodo, o que acceden a múltiples tablas base unidas por medio de relaciones de integridad referencial. En ambos casos, no debe haber predicados locales en la definición de la vista. Si se especifica el muestreo de `SYSTEM` en una vista que no puede soportar dicho muestreo, se emite un error `SQL20288N`.

#### **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. Asimismo, el método con el que se obtuvo la muestra del modo especificado por la palabra clave BERNOULLI o SYSTEM, también debe ser el mismo para asegurar 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).

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 harán que se eliminen las estadísticas de distribución recopiladas 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 recopiladas 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 clústeres 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 clústeres 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 estadísticas de captación previa que no pueden calcularse al trabajar en modalidad DMS. Al mirar en las estadísticas de índice de 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 solo se realiza con páginas de datos de tablas y no 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'.

### **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' )
```

## 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 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.

## 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 ALERT CONFIGURATION utilizando el procedimiento ADMIN\_CMD

Actualiza los valores de configuración de alertas para los indicadores de salud.

## Autorización

Una de las siguientes:

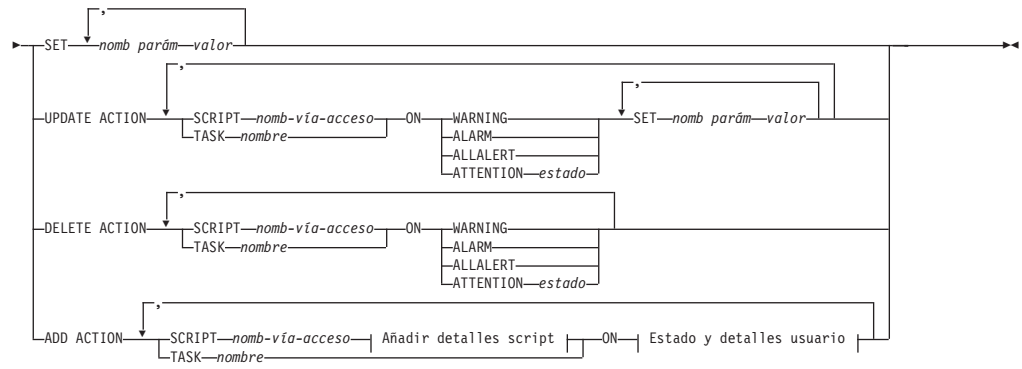
- *sysadm*
- *sysmaint*
- *sysctrl*

## Conexión necesaria

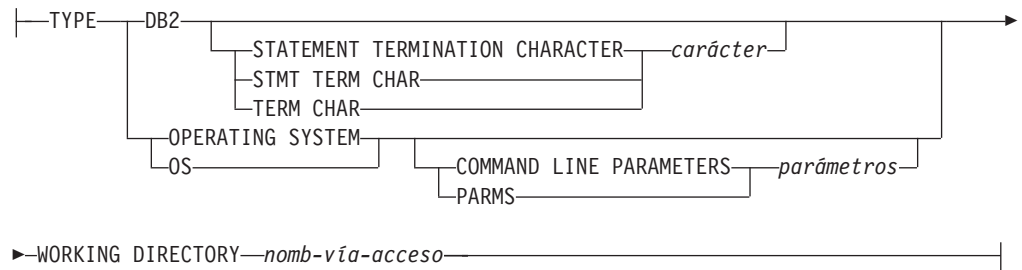
Base de datos.

## Sintaxis del mandato

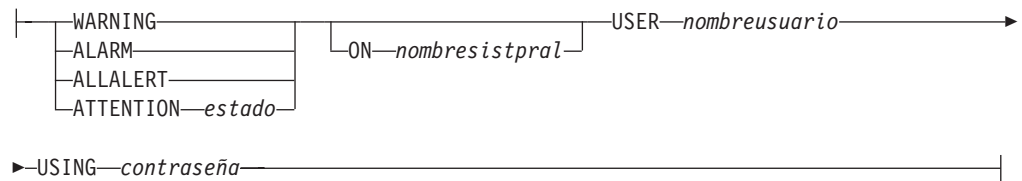
```
►►—UPDATE ALERT—CONFIGURATION—FOR—  
|—CONFIG—  
|—CFG—  
|—DATABASE MANAGER—  
|—DB MANAGER—  
|—DBM—  
|—DATABASES—  
|—CONTAINERS—  
|—TABLESPACES—  
|—DATABASE—ON—alias basedatos—  
|—TABLESPACE—nombre—  
|—CONTAINER—nombre—FOR—ID-espacio-tablas—  
►►—USING—nombre indicador salud—
```



**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* 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 ON alias basedatos`.

**DATABASE ON *alias basedatos***

Actualiza los valores de alerta de 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* FOR *ID-espacio-tablas* ON *alias basedatos***

Actualiza los valores de alerta del contenedor del espacio de tablas denominado *nombre*, para el espacio de tablas especificado utilizando la cláusula `FOR id-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* ON *alias basedatos***

Actualiza los valores de alerta para el espacio de tablas llamado *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

Los tipos válidos son OS o DB2.

- WORKINGDIR
- TERMCHAR
- CMDLINEPARMS

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* 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* 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. Cuando se utiliza la opción `ADD ACTION`, el *nombusuario* y la *contraseña* podrían quedar expuestos en la red (donde *nombusuario* y *contraseña* se envíen sin cifrar), el archivo `db2diag.log`, los archivos de rastreo, el archivo de vuelco, el supervisor de instantáneas (instantánea SQL dinámica), las instantáneas del supervisor del sistema, algunos supervisores de sucesos (como los de sentencias y puntos muertos), Query Patroller, las tablas de explicación, la salida de `db2pd` (como los mecanismos de antememoria de paquete y de 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 mypasswvdv' )
```

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 sobre 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)
- db2diag.log
- 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 explicación
- salida de db2pd (mecanismos de antememoria de paquete y tiempo de espera excedido de bloqueo, entre otros)
- Registro de anotaciones 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 del servidor y debe ser local con respecto al 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**

##### **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 buscapersonas.

## MAXPAGELEN

Número máximo de caracteres que el buscaperonas 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.

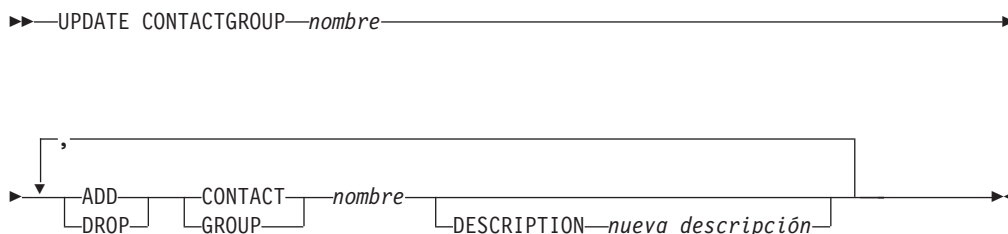
## Autorización

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**

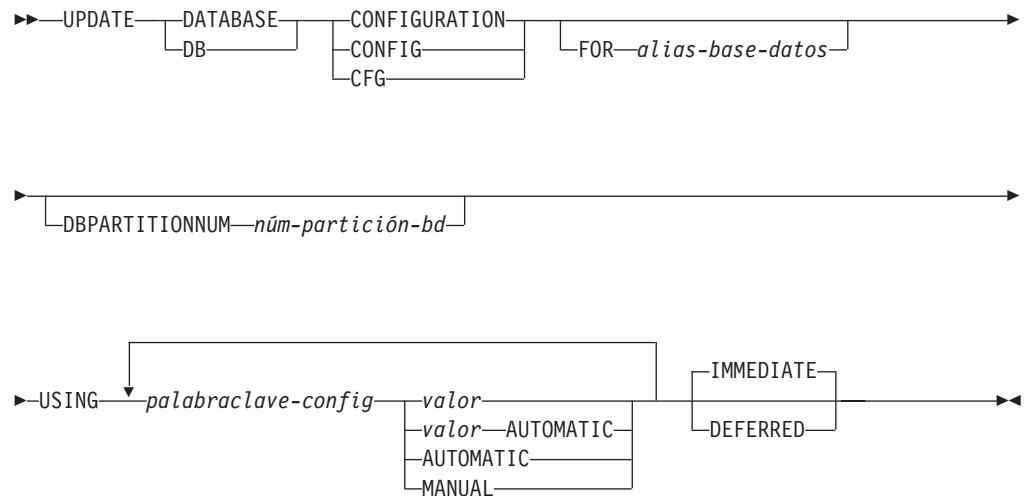
Una de las siguientes:

- *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 la 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.

### **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 de gestor de bases de datos para la instancia que contiene la base de datos conectada actualmente..

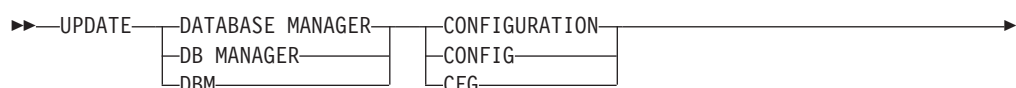
### **Autorización**

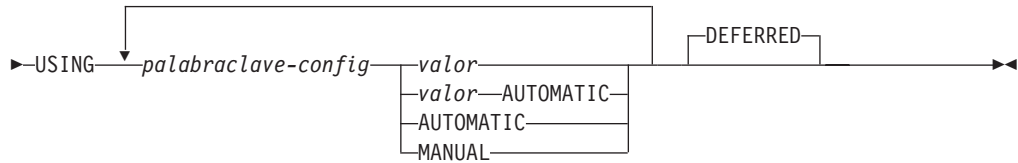
*sysadm*

### **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:** Observe que el parámetro de configuración del gestor de bases de datos *federated\_async* sólo se puede establecer en *AUTOMATIC* utilizando el procesador de la 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. Éste es el valor por omisión.

### 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**

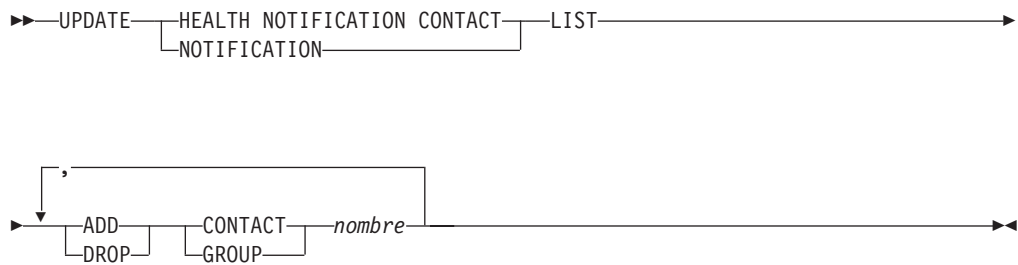
Una de las siguientes:

- 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

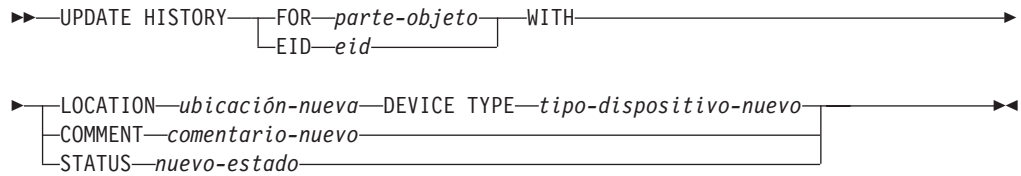
Una de las siguientes:

- *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 la 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 interconexión 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

Autorización SYSADM o DBADM

### 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, la partición de base de datos de ajuste STMM utilizará la partición de base de datos por omisión como se ha definido para la característica STMM. 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 sobre 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 ejecución del mandato se devuelve en la SQLCA resultante de la sentencia CALL.

Este mandato confirma sus cambios en el procedimiento ADMIN\_CMD.

## 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-partición-bd* (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-partición-bd*, 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

*núm-partición-bd*

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 60. 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

```
DB2 "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.)

```
DB2 "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

```
DB2 "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).

## **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 subdirectorio 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.del".
-              -          SQL3105N El programa de utilidad de exportación
                          ha finalizado la exportación de "8"
                          filas.
```

2 registro(s) seleccionado(s).

## Notas sobre 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 61. 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.
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"><li>• 'LOAD': para agente de carga</li><li>• 'PART': para agente de particionamiento</li><li>• 'PREP': para agente de preparticionamiento</li><li>• Nulo: no hay información de tipo de agente disponible</li></ul>
SQLCODE	VARCHAR(9)	SQLCODE del mensaje que se devuelve.
MSG	VARCHAR(1024)	Mensaje de error abreviado correspondiente al SQLCODE.



## Vista ADMINTABCOMPRESSINFO y ADMIN\_GET\_TAB\_COMPRESS\_INFO

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

Privilegio SELECT o CONTROL en la vista administrativa ADMINTABCOMPRESSINFO y privilegio EXECUTE en la tabla de función ADMIN\_GET\_TAB\_\_COMPRESS\_INFO.

### 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:

TABSHEMA	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 TABSHEMA, 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	SYSCOLUMNS	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 AADMINTABCOMPRESSINFO y los metadatos de la función de tabla ADMIN\_GET\_TAB\_COMPRESS\_INFO encontrará una lista completa de la información que puede devolverse.

### 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. Este 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,
       DATAPARTITIONNUM, 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 migradas de una versión previa de DB2.

```
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
-----

```

### Utilización de Notas

- 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 deben 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 62. Vista administrativa ADMINTABCOMPRESSINFO y metadatos de la función de tabla ADMIN\_GET\_TAB\_COMPRESS\_INFO

Nombre de columna	Tipo de datos	Descripción
TABSHEMA	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
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"> <li>• 'Y' = Compresión de filas establecida en sí</li> <li>• 'N' = Compresión de filas establecida en no</li> </ul>

Tabla 62. Vista administrativa ADMINTABCOMPRESSINFO y metadatos de la función de tabla ADMIN\_GET\_TAB\_COMPRESS\_INFO (continuación)

Nombre de columna	Tipo de datos	Descripción
DICT_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"> <li>• 'INSPECT' = INSPECT ROWCOMPESTIMATE</li> <li>• 'LOAD' = LOAD INSERT/REPLACE</li> <li>• 'NOT BUILT' = ningún diccionario disponible</li> <li>• 'REDISTRIBUTE' = REDISTRIBUTE</li> <li>• 'REORG' = REORG RESETDICTIONARY</li> <li>• 'TABLE FUNCTION' = ADMIN_GET_TAB_COMRESS_INFO</li> <li>• 'TABLE GROWTH' = INSERT</li> </ul>
DICT_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.
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.

## 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 sobre 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.

## Vista administrativa ADMINTABINFO y función de tabla ADMIN\_GET\_TAB\_INFO\_V95 – Recuperar información de tamaño y estado para las tablas

La función de tabla ADMINTABINFO y la función de tabla ADMIN\_GET\_TAB\_INFO\_V95 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 los metadatos de la función de tabla ADMIN\_GET\_TAB\_INFO\_V95 encontrará una lista completa de la información que puede devolverse.

## Autorización

Privilegio SELECT o CONTROL para la vista administrativa ADMINTABINFO y privilegio EXECUTE en la tabla de función ADMIN\_GET\_TAB\_INFO\_V95.

## 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
```

## Función de tabla ADMIN\_GET\_TAB\_INFO\_V95

La función de tabla ADMIN\_GET\_TAB\_INFO\_V95 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 los metadatos de la función de tabla ADMIN\_GET\_TAB\_INFO\_V95 encontrará una lista completa de la información que puede devolverse.

## Sintaxis

```
►►—ADMIN_GET_TAB_INFO_V95—(—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\_V95.

## Ejemplos

*Ejemplo 1:* Recuperar información de tamaño y estado para la tabla DBUSER1.EMPLOYEE.

```
SELECT * FROM TABLE (SYSPROC.ADMIN_GET_TAB_INFO_V95('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_V95( '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_V95( 'DBUSER1', 'EMPLOYEE' )) AS T
```

*Ejemplo 3:* Compruebe el tipo actual de información estadística reunida para la tabla T1

```
db2 => select substr(tabschema, 1, 10) as tbschema, substr(tabname, 1, 10)
as tname, statstype from SYSIBMADM.ADMINTABINFO where tabname = 'T1';
```

TBSHEMA	TBNAME	STATSTYPE
-----	-----	-----
DB2USER1	T1	U

1 registro(s) seleccionado(s).

## Notas sobre 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 deben 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\_V95 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. redistribución realizada satisfactoriamente y confirmada para la tabla. N

### Vista administrativa ADMINTABINFO y los metadatos de función de tabla ADMIN\_GET\_TAB\_INFO\_V95

Tabla 63. Vista administrativa ADMINTABINFO y los metadatos de función de tabla ADMIN\_GET\_TAB\_INFO\_V95

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"> <li>• 'H' = tabla de jerarquía</li> <li>• 'S' = tabla de consultas materializadas</li> <li>• 'T' = tabla</li> </ul>
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"> <li>• '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.</li> <li>• 'Y' = la tabla está disponible.</li> </ul> <p><b>Nota:</b> La realización de una recuperación en avance en una carga no recuperable establecerá una tabla en estado de no disponibilidad.</p>

Tabla 63. Vista administrativa ADMINTABINFO y los metadatos de función de tabla ADMIN\_GET\_TAB\_INFO\_V95 (continuación)

Nombre de columna	Tipo de datos	Descripción
DATA_OBJECT_L_SIZE	BIGINT	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.
DATA_OBJECT_P_SIZE	BIGINT	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.
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.

Tabla 63. Vista administrativa ADMINTABINFO y los metadatos de función de tabla ADMIN\_GET\_TAB\_INFO\_V95 (continuación)

Nombre de columna	Tipo de datos	Descripción
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.
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.
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.

Tabla 63. Vista administrativa ADMINTABINFO y los metadatos de función de tabla ADMIN\_GET\_TAB\_INFO\_V95 (continuación)

Nombre de columna	Tipo de datos	Descripción
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"> <li>• 1 si están utilizándose índices de tipo 1.</li> <li>• 2 si están utilizándose índices de tipo 2.</li> </ul>
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"> <li>• ABORTED (en estado PAUSED, pero incapaz de RESUME; se necesita STOP)</li> <li>• EXECUTING</li> <li>• NULL (si no se ha realizado ninguna reorganización in situ en la tabla)</li> <li>• PAUSED</li> </ul>
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"> <li>• IN_PROGRESS</li> <li>• NULL (si no existe ninguna carga en progreso para la tabla y si la tabla no está en estado de carga pendiente)</li> <li>• PENDING</li> </ul>
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)	'Y' si debe volver a crearse cualquiera de los índices definidos en la tabla; de lo contrario, 'N'.

Tabla 63. Vista administrativa ADMINTABINFO y los metadatos de función de tabla ADMIN\_GET\_TAB\_INFO\_V95 (continuación)

Nombre de columna	Tipo de datos	Descripción
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.
REDISTRIBUTE_PENDING	CHAR(1)	Una "Y" indicaría que actualmente se está ejecutando una operación de redistribución o que está pendiente en la tabla. De lo contrario, se devuelve una "N".
BLOCKS_PENDING_CLEANUP	BIGINT	Para tablas MDC, el número de bloqueos pendientes de limpieza. Para tablas no MDC este valor siempre será cero.

Tabla 63. Vista administrativa ADMINTABINFO y los metadatos de función de tabla ADMIN\_GET\_TAB\_INFO\_V95 (continuación)

Nombre de columna	Tipo de datos	Descripción
STATSTYPE	CHAR(1)	<ul style="list-style-type: none"> <li>• '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.</li> <li>• '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.</li> <li>• '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.</li> <li>• '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.</li> <li>• NULL = tipo desconocido</li> </ul>
REDISTRIBUTE_PENDING	CHAR(1)	Una "Y" indicaría que actualmente se está ejecutando una operación de redistribución o que está pendiente en la tabla. De lo contrario, se devuelve una "N".

## 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*—,—*núm-partición-bd*—)————►►

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 registro de comprobación. 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.

*núm-partición-bd*

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 64. 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.
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 - realiza una extracción al 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 registro de comprobación 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 registro de comprobación 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

Ejecutar todos los privilegios en las funciones  
SYSPROC.AUDIT\_DELIM\_EXTRACT y SYSPROC.AUDIT\_LIST\_LOGS

## Ejemplos

**Nota:** Los archivos de registro de comprobación 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 registro de comprobación 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 registro de comprobación 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 SYSPROC.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 65. 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.

---

## 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 las 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 el 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 política denominado 'policy.xml', si db2set da DB2INSTPROF=C:\DB2PROF y %DB2INSTANCE%=db2, entonces 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 la 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.

### 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.

### Ejemplo

Para establecer los valores actuales del mantenimiento automático para operaciones runstats:

```
db2 "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."

Existen archivos de entrada XML de ejemplo en el directorio SQLLIB/samples/automaintcfg que puede modificar para cumplir sus requisitos y pasar el contenido

XML dentro de la función escalar BLOB() tal como se muestra en el ejemplo anterior.

## 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 subdirectorio tmp del directorio 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 directorio y el archivo deberían tener permiso de lectura.

Por ejemplo:

En UNIX, si el directorio 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 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 política denominado 'automaint\policy.xml', si db2set da DB2INSTPROF=C:\DB2PROF y %DB2INSTANCE%=db2, entonces 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 ubicada en el directorio tmp bajo el directorio de instancias de DB2.

Existen archivos de entrada XML de ejemplo en el directorio SQLLIB/samples/automaintcfg y que pueden utilizarse como referencia para crear un archivo xml de política.

---

## 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 registros(s) seleccionado(s).

## Información devuelta

Tabla 66. 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.
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

Privilegio SELECT o CONTROL sobre la vista administrativa DBCFG y privilegio EXECUTE sobre la función de tabla DB\_GET\_CFG.

### 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_tbl_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_full    NO                   0
catalogcache_sz     -1                   0
...
```

## Información devuelta

*Tabla 67. 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"> <li>• NONE - no hay información adicional</li> <li>• AUTOMATIC - el parámetro de configuración se ha establecido como automático</li> </ul>
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 67. 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"> <li>• NONE - no hay información adicional</li> <li>• AUTOMATIC - el parámetro de configuración se ha establecido como automático</li> </ul>
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

Privilegio SELECT o CONTROL sobre la vista administrativa DBMCFG y privilegio EXECUTE sobre la función de tabla DBM\_GET\_CFG.

### 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
aslheapsz                             15
audit_buf_sz                          0
authentication                       SERVER
catalog_noauth                       YES
cInt_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
...

```

*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	...
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
...	15	NONE	BIGINT
...	0	NONE	BIGINT
...	SERVER	NONE	VARCHAR(32)
...	YES	NONE	VARCHAR(3)
...		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 68. 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"> <li>• NONE - no hay información adicional</li> <li>• AUTOMATIC - el parámetro de configuración se ha establecido como automático</li> </ul>
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.
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"> <li>• NONE - no hay información adicional</li> <li>• AUTOMATIC - el parámetro de configuración se ha establecido como automático</li> </ul>
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

Privilegio SELECT o CONTROL sobre la vista administrativa REG\_VARIABLES y privilegio EXECUTE sobre la función de tabla REG\_LIST\_VARIABLES.

## 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 69. 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 del 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) .

Tabla 69. Información devuelta por la vista administrativa REG\_VARIABLES (continuación)

Nombre de columna	Tipo de datos	Descripción
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"> <li>• I = instancia</li> <li>• G = global</li> <li>• N = partición de base de datos</li> <li>• E = entorno</li> </ul>

## Vistas del entorno

### 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

Privilegio SELECT o CONTROL sobre la vista administrativa ENV\_INST\_INFO y privilegio EXECUTE sobre la función de tabla ENV\_GET\_INST\_INFO.

#### 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                                     0
...

```

## Información devuelta

Tabla 70. 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

Privilegio SELECT o CONTROL sobre la vista administrativa ENV\_PROD\_INFO y privilegio EXECUTE sobre la función de tabla ENV\_GET\_PROD\_INFO\_V95

### 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 71. 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 71. 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"> <li>• 12_MONTHS_LICENSE_AND_SUBSCRIPTION</li> <li>• AUTHORIZED_USER</li> <li>• AUTHORIZED_USER_OPTION</li> <li>• CLIENT_DEVICE</li> <li>• CPU</li> <li>• CPU_OPTION</li> <li>• HOST_SERVER_AND_MSU</li> <li>• LICENSE_NOT_REGISTERED</li> <li>• MANAGED_PROCESSOR</li> <li>• N/A</li> <li>• RESTRICTED</li> <li>• TRIAL</li> <li>• UNWARRANTED</li> <li>• USER</li> </ul>

## 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 características existe información sobre si se ha instalado o no una licencia válida para la característica.

El esquema es SYSIBMADM.

### Autorización

Privilegio SELECT o CONTROL en la vista administrativa ENV\_FEATURE\_INFO y privilegio EXECUTE en la función de tabla ENV\_GET\_FEATURE\_INFO.

### 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  ...
HFESE        IBM_HOMOGENEOUS_FEDERATION_FEATURE_FOR_ESE  ...
XMLESE       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 72. 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"> <li>• ESE - DB2 Enterprise Server Edition</li> <li>• WSE - DB2 Workgroup Server Edition</li> <li>• EXP - DB2 Express Edition</li> </ul>
FEATURE_USE_STATUS	VARCHAR (30)	Indica el estado de cumplimiento de la licencia. Esta valor indica el estado de uso de la característica. Hay tres valores posibles: <ul style="list-style-type: none"> <li>• IN_COMPLIANCE: La característica se ha utilizado al menos una vez y dicha característica tiene una licencia válida.</li> <li>• IN_VIOLATION: La característica se ha utilizado al menos una vez y dicha característica tiene ninguna licencia válida.</li> <li>• NOT_USED: La característica no se ha utilizado.</li> </ul>

## 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

Privilegio SELECT o CONTROL sobre la vista administrativa ENV\_SYS\_INFO y privilegio EXECUTE sobre la función de tabla ENV\_GET\_SYS\_INFO.

## 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 73. 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

Privilegio SELECT o CONTROL en la vista administrativa ENV\_SYS\_RESOURCES y privilegio EXECUTE en la función de tabla ENV\_GET\_SYS\_RESOURCES.

### 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_USAGE_TOTAL	7	SMALLINT	0

4 registro(s) seleccionado(s).

## Metadatos de la vista administrativa ENV\_SYS\_RESOURCES

Tabla 74. Metadatos de la vista administrativa ENV\_SYS\_RESOURCES

Nombre de columna	Tipo de datos	Descripción
NAME	VARCHAR(128)	Nombre del atributo. Consulte la Tabla 75 en la página 250 para ver los valores posibles. <b>Nota:</b> 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 utilizad 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 75. 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 75. 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 CPUs.	Todos	NULL
	CPU_ONLINE	BIGINT	Número de CPUs en línea.	Todos	NULL
	CPU_CONFIGURED	BIGINT	Número de CPUs 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 75. 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

## 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 tabla desde una instantánea de los espacios de tabla de una base de datos.

#### Sintaxis

▶▶—HEALTH\_CONT\_HI—(—nombre-bd—,—número-partición-bd—)▶▶

El esquema es SYSPROC.

#### Parámetros de la función de tabla

*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.

*núm-partición-bd*

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. 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 76. 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 del contenedor
NODE_NUMBER	INTEGER	node_number - Número de nodo

Tabla 76. Información devuelta por la función de tabla HEALTH\_CONT\_HI (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
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

La función de tabla HEALTH\_CONT\_HI\_HIS devuelve información histórica del indicador de salud sobre contenedores desde una instantánea de la salud de una base de datos.

### Sintaxis

►►—HEALTH\_CONT\_HI\_HIS—(—nombre-bd—,—núm-partición-bd—)—————►►

El esquema es SYSPROC.

### Parámetros de la función de tabla

*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.

*núm-partición-bd*

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. 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 77. 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 del contenedor
NODE_NUMBER	INTEGER	node_number - Número de nodo

Tabla 77. Información devuelta por la función de tabla HEALTH\_CONT\_HI\_HIS (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
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.

### Sintaxis

▶▶—HEALTH\_CONT\_INFO—(—nombre-bd—,—núm-partición-bd—)—————▶▶

El esquema es SYSPROC.

### Parámetros de la función de tabla

*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.

*núm-partición-bd*

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. 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 78. 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 del 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.

## Sintaxis

►►—HEALTH\_DB\_HI—(—*nombre-bd*—,—*núm-partición-bd*—)—————►►

El esquema es SYSPROC.

## Parámetros de la función de tabla

### *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 todas las bases de datos bajo la instancia de la base de datos.

### *núm-partición-bd*

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. Si se especifica el valor nulo, se establece -1 de forma implícita.

## Autorización

Privilegio EXECUTE para la función de tabla HEALTH\_DB\_HI.

## Ejemplo

```
SELECT * FROM TABLE(HEALTH_DB_HI('',-1)) AS T
```

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

SNAPSHOT_TIMESTAMP	HI_ID	DB_NAME	HI_VALUE	...
2006-02-13-12.30.23.949888	1001	SAMPLE	0	...
2006-02-13-12.30.23.949888	1002	SAMPLE	0	...
2006-02-13-12.30.23.949888	1003	SAMPLE	0	...
2006-02-13-12.30.23.949888	1005	SAMPLE	6	...
2006-02-13-12.30.23.949888	1006	SAMPLE	53	...
2006-02-13-12.30.23.949888	1008	SAMPLE	3	...
2006-02-13-12.30.23.949888	1010	SAMPLE	0	...
2006-02-13-12.30.23.949888	1014	SAMPLE	74	...
2006-02-13-12.30.23.949888	1015	SAMPLE	1	...
2006-02-13-12.30.23.949888	1018	SAMPLE	1	...
2006-02-13-12.30.23.949888	1022	SAMPLE	1	...

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	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	1	Normal	...

```

... 2006-02-13-12.30.25.640000      2 Attention      ...
... 2006-02-13-12.30.25.640000      2 Attention      ...
... 2006-02-13-12.29.25.281000      2 Attention      ...

```

Datos de salida de esta consulta (continuación).

```

... HI_FORMULA                      ...
... -----...                      ...
... 0                                ...
... ((0 / 5000) * 100)               ...
...                                  ...
...                                  ...
...                                  ...
...                                  ...
...                                  ...
...                                  ...
...                                  ...
... ((0 - 0) / ((118 - 0) + 1)) * 100 ...
...                                  ...
...                                  ...
...                                  ...
...                                  ...
... ((1170384 / (1170384 + 19229616)) * 100) ...
...                                  ...
...                                  ...
...                                  ...
...                                  ...
... ((11155116032 / 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 79. 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 la 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.

Tabla 79. Información devuelta por la función de tabla HEALTH\_DB\_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\_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.

### Sintaxis

►►—HEALTH\_DB\_HI\_HIS—(—nombre-bd—,—núm-partición-bd—)—————►►

El esquema es SYSPROC.

### Parámetros de la función de tabla

*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 todas las bases de datos bajo la instancia de la base de datos.

*núm-partición-bd*

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. 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 ...
```







```

... 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 80. 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 la 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.

### Sintaxis

►►—HEALTH\_DB\_HIC—(—nombre-bd—,—número-partición-bd—)————◄◄

El esquema es SYSPROC.

## Parámetros de la función de tabla

### *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 todas las bases de datos bajo la instancia de la base de datos.

### *núm-partición-bd*

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. 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 81. 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.

Tabla 81. Información devuelta por la función de tabla HEALTH\_DB\_HIC (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
DB_NAME	VARCHAR(128)	db_name - Nombre de la base de datos
HI_OBJ_NAME	VARCHAR(256)	Un nombre que identifica exclusivamente a un objeto de la colección.
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"> <li>• NORMAL (1). No se necesita acción alguna en este objeto.</li> <li>• ATTENTION (2). No se ha habilitado la automatización para este indicador de salud; la acción debe ser manual.</li> <li>• AUTOMATED (5). Se ha habilitado la automatización para este indicador de salud; la acción se iniciará automáticamente.</li> <li>• 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.</li> </ul>
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

La función de tabla HEALTH\_DB\_HIC\_HIS devuelve información histórica del indicador de salud de la colección desde una instantánea de salud de una base de datos.

### Sintaxis

►►—HEALTH\_DB\_HIC\_HIS—(—nombre-bd—,—núm-partición-bd—)————◄◄

El esquema es SYSPROC.

## Parámetros de la función de tabla

*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 todas las bases de datos bajo la instancia de la base de datos.

*núm-partición-bd*

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. 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 82. 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 la 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"> <li>• NORMAL (1). No se necesita acción alguna en este objeto.</li> <li>• ATTENTION (2). No se ha habilitado la automatización para este indicador de salud; la acción debe ser manual.</li> <li>• AUTOMATED (5). Se ha habilitado la automatización para este indicador de salud; la acción se iniciará automáticamente.</li> <li>• 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.</li> </ul>
HI_OBJ_STATE_DETAIL	VARCHAR(20)	Versión convertida de la serie del valor de la columna HI_OBJ_STATE.

Tabla 82. 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_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.

### Sintaxis

►► HEALTH\_DB\_INFO (—nombre-bd—, —número-partición-bd—) ◀◀

El esquema es SYSPROC.

### Parámetros de la función de tabla

*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 todas las bases de datos bajo la instancia de la base de datos.

*número-partición-bd*

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. 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 83. 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 la 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 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.

## 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.

### Sintaxis

```

▶▶—HEALTH_DBM_HI—(—núm-partición-bd—)————▶▶

```

El esquema es SYSPROC.

### Parámetro de la función de tabla

*núm-partición-bd*

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. 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 84. 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.
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.

## Sintaxis

►—HEALTH\_DBM\_HI\_HIS—(*núm-partición-bd*)—►

El esquema es SYSPROC.

## Parámetro de la función de tabla

*núm-partición-bd*

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. 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	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                          -
... 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)    -
... ((196608 / 327680) * 100)    -

```

### Información devuelta

Tabla 85. 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.

### Sintaxis

►►—HEALTH\_DBM\_INFO—(—núm-partición-bd—)—————►►

El esquema es SYSPROC.

### Parámetro de la función de tabla

*núm-partición-bd*

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. 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 86. 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

Tabla 86. Información devuelta por la función de tabla HEALTH\_DBM\_INFO (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
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 - Iniciar indicación de la hora del gestor de bases de datos
LAST_RESET	TIMESTAMP	last_reset - Indicación de la hora de la última restauración
NUM_NODES_IN_DB2_INSTANCE	INTEGER	num_nodes_in_db2_instance - Número de nodos en la partición de base de datos

## Función de tabla HEALTH\_GET\_ALERT\_ACTION\_CFG - Recuperar valores de configuración de acciones de alerta de salud

La función de tabla 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 tabla y contenedor de espacio de tabla) y a diversos niveles de configuración (valor por omisión de instalación, instancia, global y objeto).

### Sintaxis

```
►►—HEALTH_GET_ALERT_ACTION_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 tabla
- 'TSC' para contenedor de espacio de tabla

**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 *tipoobjeto* es 'TS' o 'TSC' y *nivel\_cfg* es 'O'. Para las demás combinaciones de *tipoobjeto* 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','O','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 sobre 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 87. 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.

Tabla 87. Información devuelta por la función de tabla HEALTH\_GET\_ALERT\_ACTION\_CFG (continuación)

Nombre de columna	Tipo de datos	Descripción
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.

## Función de tabla HEALTH\_GET\_ALERT\_CFG – Recuperar valores de configuración de alertas de salud

La función de tabla HEALTH\_GET\_ALERT\_CFG devuelve valores de configuración de alertas de salud correspondientes a diversos tipos de objetos (gestor de bases de datos, base de datos, espacio de tabla, contenedor de espacio de tabla) y a diversos niveles de configuración (valor por omisión de instalación, global y objeto).

### 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





...	ID	IS_DEFAULT	WARNING_THRESHOLD	...
...	1001	0	0	...
...	1018	0	0	...
...	1015	0	0	...
...	1022	0	0	...
...	1002	1	95	...
...	1003	1	30	...
...	1004	1	60	...
...	1005	1	75	...
...	1006	1	75	...
...	1007	1	5	...
...	1008	1	75	...
...	1009	1	5	...
...	1010	1	50	...
...	1011	1	80	...

Datos de salida de esta consulta (continuación).

...	ALARM_THRESHOLD	SENSITIVITY	EVALUATE	ACTION_ENABLED	...
...	0	0	0	0	...
...	0	0	1	0	...
...	0	0	1	0	...
...	0	0	1	0	...
...	100	0	0	0	...
...	50	0	1	0	...
...	30	0	1	0	...
...	85	0	1	0	...
...	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 tabla 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 registro(s) seleccionado(s) con 1 mensaje(s) de aviso impreso(s).

### Notas sobre 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 tabla (ts.ts\_util) para el espacio de tabla 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 registro(s) seleccionado(s) con 1 mensaje(s) de aviso impreso(s).

## Información devuelta

Tabla 88. 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.
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.

## Función de tabla HEALTH\_GET\_IND\_DEFINITION – Recuperar definiciones de indicador de salud

La función de tabla HEALTH\_GET\_IND\_DEFINITION devuelve las definiciones del indicador de salud.

## 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 89. 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.
TYPE	VARCHAR(16)	Tipo de indicador de salud. Los valores posibles son: <ul style="list-style-type: none"> <li>• 'THRESHOLD_UPPER': indicadores de salud basados en umbral superior.</li> <li>• 'THRESHOLD_LOWER': indicadores de salud basados en umbral inferior.</li> <li>• 'STATE': indicadores de salud basados en estado.</li> <li>• 'COLLECTION_STATE': indicadores de salud basados en estado de recopilación.</li> </ul>
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

```

▶—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.

El procedimiento HEALTH\_HI\_REC recupera un conjunto de recomendaciones aplicadas a un indicador de salud en estado de alerta sobre un objeto de 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. 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).

*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, que se ubica 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, que se ubica en el subdirectorio include del directorio sqllib.

*nombre-bd*

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 tabla o contenedor de espacio de tabla 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 tabla, el nombre de objeto se especifica como <nombre de espacio de tablas>.<nombre del contenedor>.

*núm-partición-bd*

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 los siguientes: valores comprendidos entre 0 y 999, -1 (que especifica la partición de base de datos conectada actualmente) y -2 (que especifica todas las particiones de base de datos).

*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 indica 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 recomendaciones (XML), formateado de acuerdo con la definición de esquema DB2 Health Recommendation (consulte el esquema XML DB2RecommendationSchema.xsd, ubicado en el subdirectorio misc del directorio sqllib). 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

La función de tabla HEALTH\_TBS\_HI devuelve información del indicador de salud correspondiente a espacios de tabla desde una instantánea de la salud de los espacios de tabla de una base de datos.

### Sintaxis

►►—HEALTH\_TBS\_HI—(—nombre-bd—,—núm-partición-bd—)—————►►

El esquema es SYSPROC.

### Parámetros de la función de tabla

*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.

*núm-partición-bd*

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. 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". The 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)
"N/A", maximum size (bytes) "-1". The
current table space size (bytes) is
"1048576".
... 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. 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".

```

## Información devuelta

Tabla 90. 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 tabla a partir de una instantánea de la salud de una base de datos.

## Sintaxis

►►—HEALTH\_TBS\_HI\_HIS—(—nombre-bd—,—núm-partición-bd—)————►►

El esquema es SYSPROC.

## Parámetros de la función de tabla

### *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.

### *núm-partición-bd*

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. 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	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". The 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". The

```

... 0
... 0
... ((156 / 252) * 100)
... ((156 / 252) * 100)
... 0
... 0
... 0
... 0
... 0
... 0
... ((1504 / 1504) * 100)
... ((1504 / 1504) * 100)

```

current table space size (bytes) is "38797312".

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".

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".

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 "1048576".

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 "1048576".

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".

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"

... 0 respectively. The table space is defined with automatic storage set to "YES" and automatic resize enabled set to "YES". 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 91. 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

La función de tabla HEALTH\_TBS\_INFO devuelve información de espacio de tabla desde una instantánea de la salud de una base de datos.

### Sintaxis

►►—HEALTH\_TBS\_INFO—(—nombre-bd—,—núm-partición-bd—)—◄◄

El esquema es SYSPROC.

## Parámetros de la función de tabla

*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.

*núm-partición-bd*

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. 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 92. 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.

## Rutinas de MQSeries

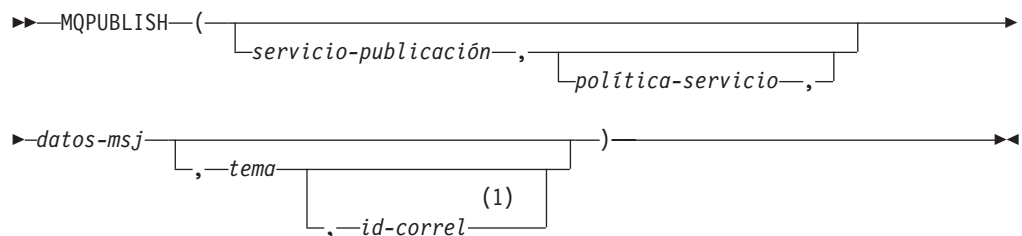
### MQPUBLISH

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')
```

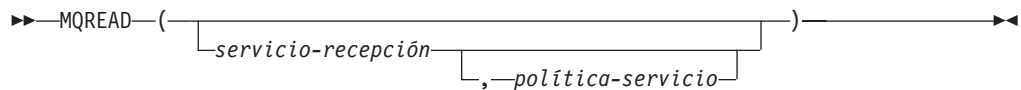
## MQREAD

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')
```

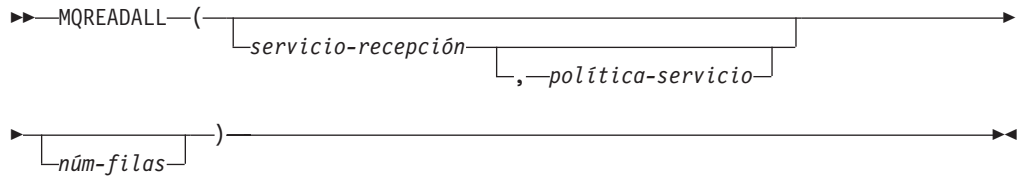
Ejemplo 3: En este ejemplo se lee el mensaje situado en la cabecera de la cola especificada por el servicio "MYSERVICE", utilizando la política "MYPOLICY".

```
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 93. Información devuelta por la función de tabla MQREADALL

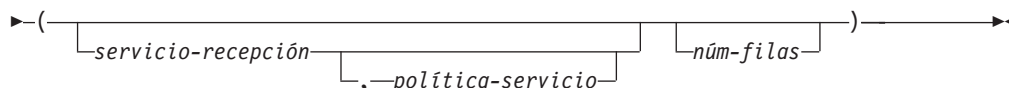
Nombre de columna	Tipo de datos	Descripción
MSG	VARCHAR(32000)	Incluye el contenido del mensaje MQSeries.
CORRELID	VARCHAR(24)	Contiene un ID de correlación que puede utilizarse para identificar mensajes. Puede seleccionar un mensaje de la cola utilizando este identificador. En el caso de un escenario 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

►►—MQREADALLCLOB—►



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 94. 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

```

  ►► MQREADCLOB ( ( servicio-recepción , política-servicio ) )
  
```

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')
```

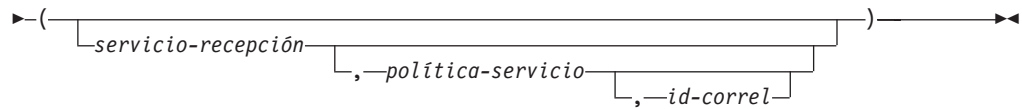
## MQRECEIVE

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

```
➡—MQRECEIVE—→
```



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')
```

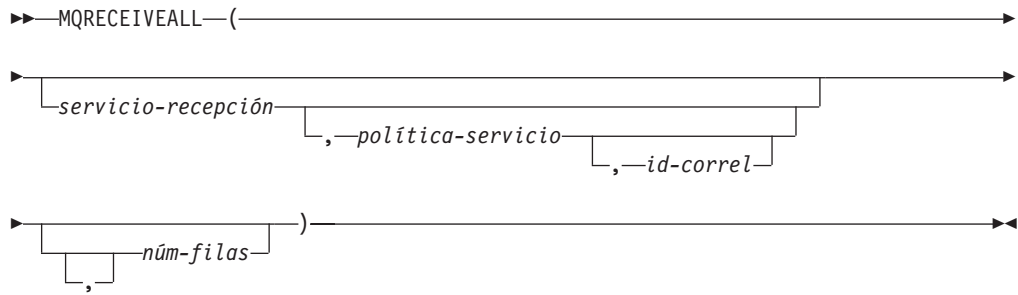
Ejemplo 4: En este ejemplo se recibe el primer mensaje con un id de correlación que coincide con '1234' de la cabecera de la cola especificada por el servicio "MYSERVICE", utilizando la política "MYPOLICY".

```
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 95. Información devuelta por la función de tabla MQRECEIVEALL

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.

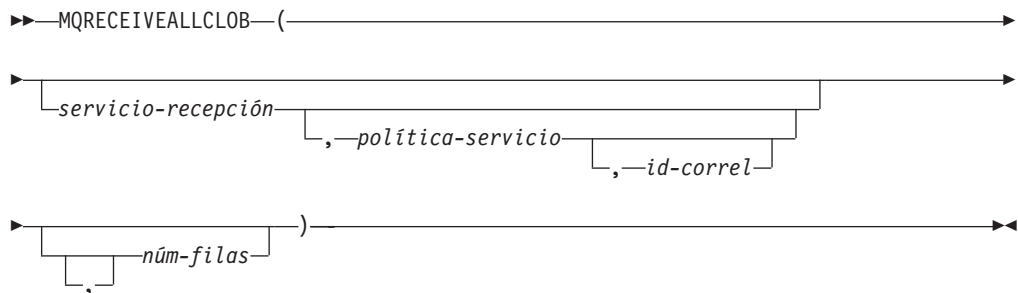
Tabla 95. Información devuelta por la función de tabla MQRECEIVEALL (continuación)

Nombre de columna	Tipo de datos	Descripción
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 96. 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

```
MQRECEIVECLOB (
  servicio-recepción
  [, política-servicio
  [, id-correl
  ]
)
```

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".

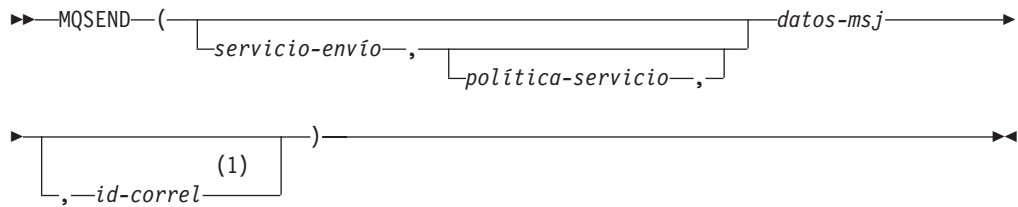
```
VALUES MQRECEIVECLOB('MYSERVICE', 'MYPOLICY', '1234')
```

## 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

```
►► MQSUBSCRIBE ( servicio-suscripción , política-servicio , tema )
```

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

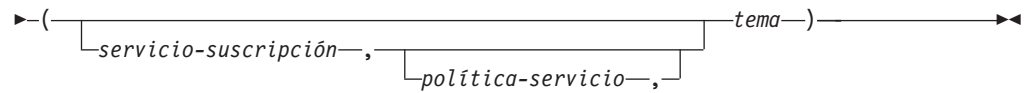
La función MQUNSUBSCRIBE se utiliza para eliminar el registro de una suscripción de mensajes existente. El *servicio-suscripción*, la *política-servicio* y el *tema* se utilizan para identificar la suscripción que se debe cancelar. La ejecución satisfactoria de esta función causa que el servidor de publicación 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

►►—MQUNSUBSCRIBE—►►





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')
```

---

## 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 una función.

## Sintaxis

►►—AUTH\_LIST\_AUTHORITIES\_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 función. 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: Función
- U: Usuario

## Autorización

Privilegio EXECUTE sobre la función de tabla AUTH\_LIST\_AUTHORITIES\_FOR\_AUTHID.

## Información devuelta

Tabla 97. 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 una función (R), entonces el valor no se aplicará (*'). <ul style="list-style-type: none"><li>• N = No se mantiene</li><li>• Y = Se mantiene</li><li>• * = No se aplica</li></ul>
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 una función (R), entonces el valor no se aplicará (*'). <ul style="list-style-type: none"><li>• N = No se mantiene</li><li>• Y = Se mantiene</li><li>• * = No se aplica</li></ul>
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 una función (R), entonces el valor no se aplicará (*'). <ul style="list-style-type: none"><li>• N = No se mantiene</li><li>• Y = Se mantiene</li><li>• * = No se aplica</li></ul>

Tabla 97. La información devuelta para AUTH\_LIST\_AUTHORITIES\_FOR\_AUTHID (continuación)

Nombre de columna	Tipo de datos	Descripción
ROLE_USER	CHAR(1)	Autorización otorgada directamente a una función con el <i>idaut</i> , cuando el <i>tipoidaut</i> es un usuario (U). Si el <i>tipoidaut</i> es un grupo (G) o una función (R), entonces el valor no se aplicará (*). La función podría formar parte de una jerarquía de funciones. <ul style="list-style-type: none"> <li>• N = No se mantiene</li> <li>• Y = Se mantiene</li> <li>• * = No se aplica</li> </ul>
ROLE_GROUP	CHAR(1)	Autorización otorgada directamente a una función 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 función (R), entonces el valor no se aplicará (*). La función podría formar parte de una jerarquía de funciones. <ul style="list-style-type: none"> <li>• N = No se mantiene</li> <li>• Y = Se mantiene</li> <li>• * = No se aplica</li> </ul>
ROLE_PUBLIC	CHAR(1)	Autorización otorgada directamente a una función 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 una función (R), entonces el valor no se aplicará (*). La función podría formar parte de una jerarquía de funciones. <ul style="list-style-type: none"> <li>• N = No se mantiene</li> <li>• Y = Se mantiene</li> <li>• * = No se aplica</li> </ul>
D_ROLE	CHAR(1)	Autorización otorgada a una función o a una función otorgada a la función. Si el <i>tipoidaut</i> es un usuario (U) o un grupo (G), entonces el valor no se aplicará (*). La función podría formar parte de una jerarquía de funciones. <ul style="list-style-type: none"> <li>• N = No se mantiene</li> <li>• Y = Se mantiene</li> <li>• * = No se aplica</li> </ul>

## 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 otorgado la función R1 a ALICE. Se ha otorgado la autorización LOAD a la función R1. Se ha otorgado la función R2 al grupo ADMIN1. La autorización CREATE\_NOT\_FENCED\_ROUTINE se ha otorgado a la función R2.

Ejemplo 1: Recuperar todas las autorizaciones otorgadas directamente al usuario ALICE o indirectamente a través de un grupo, PUBLIC o una función.

```
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
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	*
DBADM	Y	Y	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	*
SYSADM	*	Y	*	*	*	*	*
SYSCTRL	*	Y	*	*	*	*	*
SYSMAINT	*	Y	*	*	*	*	*
SYSMON	*	N	*	*	*	*	*

Ejemplo 2: Recuperar todas las autorizaciones otorgadas directamente al grupo ADMIN1 o indirectamente a través de PUBLIC o una función.

```
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
-----	-----	-----	-----	-----	-----	-----	-----
BINDADD	*	N	*	*	N	*	*
CONNECT	*	N	*	*	N	*	*
CREATE_EXTERNAL_ROUTINE	*	N	*	*	N	*	*
CREATE_NOT_FENCED_ROUTINE	*	N	*	*	Y	*	*
CREATETAB	*	N	*	*	N	*	*
DBADM	*	N	*	*	N	*	*
IMPLICIT_SCHEMA	*	N	*	*	N	*	*
LOAD	*	N	*	*	N	*	*
QUIESCE_CONNECT	*	N	*	*	N	*	*
SECADM	*	N	*	*	N	*	*
SYSADM	*	Y	*	*	*	*	*
SYSCTRL	*	Y	*	*	*	*	*
SYSMAINT	*	Y	*	*	*	*	*
SYSMON	*	N	*	*	*	*	*

Ejemplo 3: Recuperar todas las autorizaciones otorgadas directamente al grupo especial PUBLIC o indirectamente a través de una función.

```
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
-----	-----	-----	-----	-----	-----	-----	-----
BINDADD	*	*	Y	*	*	N	*
CONNECT	*	*	Y	*	*	N	*
CREATE_EXTERNAL_ROUTINE	*	*	N	*	*	N	*
CREATE_NOT_FENCED_ROUTINE	*	*	N	*	*	N	*
CREATETAB	*	*	Y	*	*	N	*
DBADM	*	*	N	*	*	N	*
IMPLICIT_SCHEMA	*	*	Y	*	*	N	*
LOAD	*	*	N	*	*	N	*
QUIESCE_CONNECT	*	*	N	*	*	N	*
SECADM	*	*	N	*	*	N	*
SYSADM	*	*	*	*	*	*	*
SYSCTRL	*	*	*	*	*	*	*
SYSMAINT	*	*	*	*	*	*	*
SYSMON	*	*	*	*	*	*	*

Ejemplo 4: Recuperar todas las autorizaciones otorgadas directamente a la función R1 o indirectamente a través de una función. Tenga en cuenta que en este caso, también se ha otorgado la función R2 a la función 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
-----	-----	-----	-----	-----	-----	-----	-----
BINDADD	*	*	*	*	*	*	N
CONNECT	*	*	*	*	*	*	N
CREATE_EXTERNAL_ROUTINE	*	*	*	*	*	*	N
CREATE_NOT_FENCED_ROUTINE	*	*	*	*	*	*	Y
CREATETAB	*	*	*	*	*	*	N
DBADM	*	*	*	*	*	*	N

IMPLICIT_SCHEMA	*	*	*	*	*	*	N
LOAD	*	*	*	*	*	*	Y
QUIESCE_CONNECT	*	*	*	*	*	*	N
SECADM	*	*	*	*	*	*	N
SYSADM	*	*	*	*	*	*	*
SYSCTRL	*	*	*	*	*	*	*
SYSMAINT	*	*	*	*	*	*	*
SYSMON	*	*	*	*	*	*	*

## Utilización de Notas

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 funciones otorgadas al grupo) a quien pertenece el *idaut*
- otorgada a cualquier función (o funciones otorgadas al grupo) otorgada a *idaut*
- otorgada a PUBLIC (o funciones otorgadas 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

►►—AUTH\_LIST\_GROUPS\_FOR\_AUTHID—(—*idaut*—)—————►►

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 ejemplo siguiente muestra los datos de salida de la consulta.

```
GROUP
-----,.,-----
BUILD
PDXDB2
```

2 registro(s) seleccionado(s).

## Notas sobre 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 98. 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 funciones

La función AUTH\_LIST\_ROLES\_FOR\_AUTHID devuelve la lista de funciones 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 función. 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: Función
- U: Usuario

### Autorización

Privilegio EXECUTE sobre la función de tabla AUTH\_LIST\_ROLES\_FOR\_AUTHID.

### Información devuelta

Tabla 99. Los conjuntos de resultados para AUTH\_LIST\_ROLES\_FOR\_AUTHID

Nombre de columna	Tipo de datos	Descripción
GRANTOR	VARCHAR(128)	El que otorga la función.
GRANTORTYPE	CHAR(1)	Tipo de otorgante: <ul style="list-style-type: none"> <li>• U = El otorgante es un usuario individual</li> </ul>
GRANTEE	VARCHAR(128)	Usuario al que se ha otorgado la función.
GRANTEETYPE	CHAR(1)	Tipo de otorgado: <ul style="list-style-type: none"> <li>• G = Se otorga a un grupo</li> <li>• R = Se otorga a una función</li> <li>• U = El otorgado es un usuario</li> </ul>
ROLENAME	VARCHAR(128)	Nombre de la función otorgada al ID de autorización directa o indirectamente mediante un grupo u otra función.
CREATE_TIME	TIMESTAMP	Hora en que se creó la función.
ADMIN	CHAR(1)	Privilegio para otorgar la función, revocar la función o comentar sobre una función: <ul style="list-style-type: none"> <li>• N = No se mantiene</li> <li>• Y = Se mantiene</li> </ul>

### Ejemplo

Imagine que otorga la función INTERN a una función DOCTOR y la función DOCTOR a la función SPECIALIST y, luego, la función SPECIALIST al usuario ALICE. ALICE pertenece al grupo HOSPITAL y la función EMPLOYEE está otorgada al grupo HOSPITAL. ALICE también pertenece al grupo especial PUBLIC. Además, la función PATIENTS está otorgada a PUBLIC.

*Ejemplo 1:* Reúna todas las funciones otorgadas 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 todas las funciones otorgadas al grupo HOSPITAL.

```

SELECT GRANTOR, GRANTORTYPE, GRRANTEE, 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 todas las funciones otorgadas a la función SPECIALIST.

```

SELECT GRANTOR, GRANTORTYPE, GRRANTEE, 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 todas las funciones otorgadas al grupo PUBLIC.

```

SELECT GRANTOR, GRANTORTYPE, GRRANTEE, 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 sobre uso

La salida de la función de tabla AUTH\_LIST\_ROLES\_FOR\_AUTHID depende de AUTHIDTYPE:

- Para un usuario, devuelve las funciones otorgadas al usuario directa o indirectamente mediante otras funciones o grupos a los que pertenece el usuario (o PUBLIC).
- Para un grupo, devuelve las funciones otorgadas al grupo directa o indirectamente mediante otras funciones.
- Para una función, devuelve las funciones otorgadas a la función directa o indirectamente mediante otras funciones.

## 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 funciones, sólo se devolverán los nombres de los grupos o las funciones.

El esquema es SYSIBMADM.

### Autorización

Privilegio SELECT o CONTROL sobre la vista administrativa AUTHORIZATIONIDS.

### 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 registros(s) seleccionado(s).

### Información devuelta

Tabla 100. 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"> <li>• U: usuario</li> <li>• R: función</li> <li>• G: grupo</li> </ul>

## 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

Privilegio SELECT o CONTROL sobre la vista administrativa OBJECTOWNERS.

## 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 101. 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

Privilegio SELECT o CONTROL sobre la vista administrativa PRIVILEGES.

## 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    SQLE0F00        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_SVR_TYPES    SYSPROC        PROCEDURE
PUBLIC      EXECUTE    LIST_SVR_VERSIONS SYSPROC        PROCEDURE
PUBLIC      EXECUTE    LIST_WRAP_OPTIONS SYSPROC        PROCEDURE
PUBLIC      EXECUTE    LIST_SVR_OPTIONS  SYSPROC        PROCEDURE
...
SYSTEM      UPDATE     POLICY_UNQ        SYSTOOLS       INDEX
PUBLIC      CREATEIN  NULLID            NULLID          SCHEMA
PUBLIC      UPDATE    COLUMNS          SYSSTAT        VIEW
PUBLIC      UPDATE    COLGROUPS         SYSSTAT        VIEW
...

```

## Información devuelta

Tabla 102. 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"> <li>• U: usuario</li> <li>• R: función</li> <li>• G: grupo</li> </ul>
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"> <li>• Y: Se puede otorgar</li> <li>• N: No se puede otorgar</li> </ul>
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.

## 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

- Privilegio SELECT o CONTROL para las vistas administrativas APPL\_PERFORMANCE, SNAPAPPL\_INFO y SNAPAPPL.
- Para acceder a los datos del supervisor de instantáneas también se necesita autorización SYSMON, SYSCTRL, SYSMAINT o 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 registros(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 103. 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 la aplicación

Tabla 103. Información devuelta por la vista administrativa  
APPL\_PERFORMANCE (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)
PERCENT_ROWS_SELECTED	DECIMAL(5,2)	El porcentaje de las filas leídas del disco que se han devuelto a la aplicación.
DBPARTITIONNUM	SMALLINT	La 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

- Privilegio SELECT o CONTROL para las vistas administrativas APPLICATIONS y SNAPAPPL\_INFO.
- Autorización SYSMON, SYSCTRL, SYSMAINT o SYSADM, necesaria para acceder a los datos del supervisor de instantáneas.

### 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 sobre 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 104. 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 la aplicación
DB_NAME	VARCHAR(128)	db_name - Nombre de la base de datos
AGENT_ID	BIGINT	agent_id - Descriptor de contexto de aplicación (ID de agente)
APPL_NAME	VARCHAR(256)	appl_name - Nombre de la aplicación
AUTHID	VARCHAR(128)	auth_id - ID de autorización
APPL_ID	VARCHAR(128)	appl_id - ID de aplicación

Tabla 104. 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"> <li>• BACKUP</li> <li>• COMMIT_ACT</li> <li>• COMP</li> <li>• CONNECTED</li> <li>• CONNECTPEND</li> <li>• CREATE_DB</li> <li>• DECOUPLED</li> <li>• DISCONNECTPEND</li> <li>• INTR</li> <li>• IOERROR_WAIT</li> <li>• LOAD</li> <li>• LOCKWAIT</li> <li>• QUIESCE_TABLESPACE</li> <li>• RECOMP</li> <li>• REMOTE_RQST</li> <li>• RESTART</li> <li>• RESTORE</li> <li>• ROLLBACK_ACT</li> <li>• ROLLBACK_TO_SAVEPOINT</li> <li>• TEND</li> <li>• THABRT</li> <li>• THCOMT</li> <li>• TPREP</li> <li>• UNLOAD</li> <li>• UOWEXEC</li> <li>• UOWWAIT</li> <li>• WAITFOR_REMOTE</li> </ul>
STATUS_CHANGE_TIME	TIMESTAMP	status_change_time - Hora de cambio de estado de la aplicación
SEQUENCE_NO	VARCHAR(4)	sequence_no - Número de secuencia
CLIENT_PRDID	VARCHAR(128)	client_prdid - ID de producto/versión del cliente
CLIENT_PID	BIGINT	client_pid - ID de proceso de cliente

Tabla 104. 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"> <li>• AIX</li> <li>• AIX64</li> <li>• AS400_DRDA</li> <li>• DOS</li> <li>• DYNIX</li> <li>• HP</li> <li>• HP64</li> <li>• HPIA</li> <li>• HPIA64</li> <li>• LINUX</li> <li>• LINUX390</li> <li>• LINUXIA64</li> <li>• LINUXPPC</li> <li>• LINUXPPC64</li> <li>• LINUXX8664</li> <li>• LINUXZ64</li> <li>• MAC</li> <li>• MVS_DRDA</li> <li>• NT</li> <li>• NT64</li> <li>• OS2</li> <li>• OS390</li> <li>• SCO</li> <li>• SGI</li> <li>• SNI</li> <li>• SUN</li> <li>• SUN64</li> <li>• UNKNOWN</li> <li>• UNKNOWN_DRDA</li> <li>• VM_DRDA</li> <li>• VSE_DRDA</li> <li>• WINDOWS</li> <li>• WINDOWS95</li> </ul>



Tabla 104. 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"> <li>• CPIC</li> <li>• LOCAL</li> <li>• NETBIOS</li> <li>• NPIPE</li> <li>• TCPIP (para DB2 UDB)</li> <li>• TCPIP4</li> <li>• TCPIP6</li> </ul>
CLIENT_NNAME	VARCHAR(128)	El elemento del supervisor client_nname está obsoleto. El valor devuelto no es un valor válido.
COORD_NODE_NUM	SMALLINT	coord_node - Coordinación del nodo
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	La 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

- Privilegio SELECT o CONTROL para las vistas administrativas BP\_HITRATIO y SNAPBP.
- Para acceder a los datos del supervisor de instantáneas también se necesita autorización SYSMON, SYSCTRL, SYSMOINT o 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 sobre 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 105. 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 la base de datos
BP_NAME	VARCHAR(128)	bp_name - Nombre de la agrupación de almacenamientos intermedios.

Tabla 105. Información devuelta por la vista administrativa BP\_HITRATIO (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
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

- Privilegio SELECT o CONTROL para las vistas administrativas BP\_READ\_IO y SNAPBP.

- Para acceder a los datos del supervisor de instantáneas también se necesita autorización SYSMON, SYSCTRL, SYSMMAINT o 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 106. 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.

Tabla 106. Información devuelta por la vista administrativa BP\_READ\_IO (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
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.
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	La 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

- Privilegio SELECT o CONTROL para las vistas administrativas BP\_WRITE\_IO y SNAPBP.
- Para acceder a los datos del supervisor de instantáneas también se necesita autorización SYSMON, SYSCTRL, SYSMAINT o 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 107. 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 107. 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 contenedores de espacio de tablas

La vista administrativa CONTAINER\_UTILIZATION devuelve información acerca de los contenedores de espacios de tabla 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

- Privilegio SELECT o CONTROL para las vistas administrativas CONTAINER\_UTILIZATION y SNAPCONTAINER.
- Autorización SYSMON, SYSCTRL, SYSMAINT o SYSADM (necesaria para acceder a los datos del supervisor de instantáneas).

### 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 de las páginas totales y 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 108. 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 del espacio de tablas
CONTAINER_NAME	VARCHAR(256)	container_name - Nombre del 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"> <li>• DISK_EXTENT_TAG</li> <li>• DISK_PAGE_TAG</li> <li>• FILE_EXTENT_TAG</li> <li>• FILE_PAGE_TAG</li> <li>• PATH</li> </ul>
TOTAL_PAGES	BIGINT	container_total_pages - Páginas totales en contenedor
USABLE_PAGES	BIGINT	container_usable_pages - Páginas útiles en contenedor
ACCESSIBLE	SMALLINT	container_accessible - Accesibilidad del contenedor
STRIPE_SET	BIGINT	container_stripe_set - Conjunto de bandas
FS_ID	VARCHAR(22)	fs_id - Número de identificación del sistema de archivos exclusivo
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	La 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

- Privilegio SELECT o CONTROL para las vistas administrativas LOCKS\_HELD, SNAPLOCK y SNAPAPPL\_INFO.
- Para acceder a los datos del supervisor de instantáneas también se necesita autorización SYSMON, SYSCTRL, SYSMAINT o 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 109. 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 la base de datos
AGENT_ID	BIGINT	agent_id - Descriptor de contexto de aplicación (ID de agente)
APPL_NAME	VARCHAR(256)	appl_name - Nombre de la 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
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"> <li>• AUTORESIZE_LOCK</li> <li>• AUTOSTORAGE_LOCK</li> <li>• BLOCK_LOCK</li> <li>• EOT_LOCK</li> <li>• INPLACE_REORG_LOCK</li> <li>• INTERNAL_LOCK</li> <li>• INTERNALB_LOCK</li> <li>• INTERNALC_LOCK</li> <li>• INTERNALJ_LOCK</li> <li>• INTERNALL_LOCK</li> <li>• INTERNALO_LOCK</li> <li>• INTERNALQ_LOCK</li> <li>• INTERNALP_LOCK</li> <li>• INTERNALS_LOCK</li> <li>• INTERNALT_LOCK</li> <li>• INTERNALV_LOCK</li> <li>• KEYVALUE_LOCK</li> <li>• ROW_LOCK</li> <li>• SYSBOOT_LOCK</li> <li>• TABLE_LOCK</li> <li>• TABLE_PART_LOCK</li> <li>• TABLESPACE_LOCK</li> <li>• XML_PATH_LOCK</li> </ul>

Tabla 109. Información devuelta por la vista administrativa LOCKS\_HELD (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
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"> <li>• IN</li> <li>• IS</li> <li>• IX</li> <li>• NON (si no existe bloqueo)</li> <li>• NS</li> <li>• NW</li> <li>• NX</li> <li>• S</li> <li>• SIX</li> <li>• U</li> <li>• W</li> <li>• X</li> <li>• Z</li> </ul>
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"> <li>• CONV</li> <li>• GRNT</li> </ul>
LOCK_ESCALATION	SMALLINT	lock_escalation - Escala de bloqueo
DBPARTITIONNUM	SMALLINT	La 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

- Privilegio SELECT o CONTROL para las vistas administrativas LOCKWAITS, SNAPAPPL\_INFO y SNAPLOCKWAIT.
- Para acceder a los datos del supervisor de instantáneas también se necesita autorización SYSMON, SYSCTRL, SYSMOINT o 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 registros(s) seleccionado(s).

## Información devuelta

Tabla 110. 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 la base de datos
AGENT_ID	BIGINT	agent_id - Descriptor de contexto de aplicación (ID de agente)
APPL_NAME	VARCHAR(256)	appl_name - Nombre de la aplicación
AUTHID	VARCHAR(128)	auth_id - ID de autorización
TBSP_NAME	VARCHAR(128)	tablespace_name - Nombre de espacio de tablas

Tabla 110. Información devuelta por la vista administrativa LOCKWAITS (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
SUBSECTION_NUMBER	BIGINT	ss_number - Número de subsección
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"> <li>• AUTORESIZE_LOCK</li> <li>• AUTOSTORAGE_LOCK</li> <li>• BLOCK_LOCK</li> <li>• EOT_LOCK</li> <li>• INPLACE_REORG_LOCK</li> <li>• INTERNAL_LOCK</li> <li>• INTERNALB_LOCK</li> <li>• INTERNALC_LOCK</li> <li>• INTERNALJ_LOCK</li> <li>• INTERNALL_LOCK</li> <li>• INTERNALO_LOCK</li> <li>• INTERNALQ_LOCK</li> <li>• INTERNALP_LOCK</li> <li>• INTERNALS_LOCK</li> <li>• INTERNALT_LOCK</li> <li>• INTERNALV_LOCK</li> <li>• KEYVALUE_LOCK</li> <li>• ROW_LOCK</li> <li>• SYSBOOT_LOCK</li> <li>• TABLE_LOCK</li> <li>• TABLE_PART_LOCK</li> <li>• TABLESPACE_LOCK</li> <li>• XML_PATH_LOCK</li> </ul>
LOCK_WAIT_START_TIME	TIMESTAMP	lock_wait_start_time - Indicación de la hora de inicio de espera de bloqueo
LOCK_NAME	VARCHAR(32)	lock_name - Nombre de bloqueo

Tabla 110. 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"> <li>• IN</li> <li>• IS</li> <li>• IX</li> <li>• NON (si no existe bloqueo)</li> <li>• NS</li> <li>• NW</li> <li>• NX</li> <li>• S</li> <li>• SIX</li> <li>• U</li> <li>• W</li> <li>• X</li> <li>• Z</li> </ul>
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"> <li>• IN</li> <li>• IS</li> <li>• IX</li> <li>• NON (si no existe bloqueo)</li> <li>• NS</li> <li>• NW</li> <li>• NX</li> <li>• S</li> <li>• SIX</li> <li>• U</li> <li>• W</li> <li>• X</li> <li>• Z</li> </ul>
AGENT_ID_HOLDING_LK	BIGINT	agent_id_holding_lock - Bloqueo de retención de ID de agente
APPL_ID_HOLDING_LK	VARCHAR(128)	appl_id_holding_lk - Bloqueo de retención de ID de aplicación
LOCK_ESCALATION	SMALLINT	lock_escalation - Escala de bloqueo
DBPARTITIONNUM	SMALLINT	La 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

- Privilegio SELECT o CONTROL para las vistas administrativas LOG\_UTILIZATION y SNAPDB.
- Para acceder a los datos del supervisor de instantáneas también se necesita autorización SYSMON, SYSCTRL, SYSMOINT o 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 111. 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 la 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.

Tabla 111. Información devuelta por la vista administrativa LOG\_UTILIZATION (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
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	La partición de base de datos desde la que se han recuperado los datos para esta fila.

## LONG\_RUNNING\_SQL, vista administrativa

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

- Privilegio SELECT o CONTROL para las vistas administrativas LONG\_RUNNING\_SQL, SNAPSTMT, SNAPAPPL\_INFO y SNAPAPPL.
- Para acceder a los datos del supervisor de instantáneas también se necesita autorización SYSMON, SYSCTRL, SYSMAINT o 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 registros(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 112. 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 la aplicación

Tabla 112. 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"> <li>• BACKUP</li> <li>• COMMIT_ACT</li> <li>• COMP</li> <li>• CONNECTED</li> <li>• CONNECTPEND</li> <li>• CREATE_DB</li> <li>• DECOUPLED</li> <li>• DISCONNECTPEND</li> <li>• INTR</li> <li>• IOERROR_WAIT</li> <li>• LOAD</li> <li>• LOCKWAIT</li> <li>• QUIESCE_TABLESPACE</li> <li>• RECOMP</li> <li>• REMOTE_RQST</li> <li>• RESTART</li> <li>• RESTORE</li> <li>• ROLLBACK_ACT</li> <li>• ROLLBACK_TO_SAVEPOINT</li> <li>• TEND</li> <li>• THABRT</li> <li>• THCOMT</li> <li>• TPREP</li> <li>• UNLOAD</li> <li>• UOWEXEC</li> <li>• UOWWAIT</li> <li>• WAITFOR_REMOTE</li> </ul>
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	La 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

- Privilegio SELECT o CONTROL para las vistas administrativas QUERY\_PREP\_COST y SNAPDYN\_SQL.
- Para acceder a los datos del supervisor de instantáneas también se necesita autorización SYSMON, SYSCTRL, SYSMMAINT o SYSADM.

### 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 sobre 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 113. 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.

Tabla 113. Información devuelta por la vista administrativa  
*QUERY\_PREP\_COST* (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
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	La 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.

### SNAPAGENT, vista administrativa

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 114 en la página 348 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL en la vista administrativa SNAPAGENT y el privilegio EXECUTE en la función de tabla SNAP\_GET\_AGENT.

### 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
...	...	...

## Función de tabla SNAP\_GET\_AGENT

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 114 en la página 348 para ver una lista completa de la información que se puede devolver.

## Sintaxis

```
▶▶ SNAP_GET_AGENT ( ( nombre-bd [ , núm-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.

*núm-partición-bd*

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. Si *nombrebd* no se establece en NULL y *núm\_partición\_bd* se establece en NULL, implícitamente se establece -1 para *núm\_partición\_bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE en la función de tabla SNAP\_GET\_AGENT.

## 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 114. 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 la base de datos

Tabla 114. Información devuelta por la vista administrativa SNAPAGENT y la función de tabla SNAP\_GET\_AGENT (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 del motor (EDU)
LOCK_TIMEOUT_VAL	BIGINT	lock_timeout_val - Tiempo 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.

### SNAPAGENT\_MEMORY\_POOL, vista administrativa

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 and SNAPSUBSECTION, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR APPLICATIONS ON alias de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 115 en la página 351 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL en la vista administrativa SNAPAGENT\_MEMORY\_POOL y privilegio EXECUTE en la función de tabla SNAP\_GET\_AGENT\_MEMORY\_POOL.

### 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).

## Función de tabla SNAP\_GET\_AGENT\_MEMORY\_POOL

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 115 en la página 351 para ver una lista completa de la información que se puede devolver.

## Sintaxis

```

▶▶ SNAP_GET_AGENT_MEMORY_POOL ( ( nombre_bd [ , núm_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.

### núm\_partición\_bd

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. Si nombre\_bd no se establece en NULL y núm\_partición\_bd se establece en NULL, implícitamente se establece -1 para núm\_partición\_bd. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona nombre\_bd, se devuelven los datos de todas las particiones de base de datos.



Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE en la función de tabla SNAP\_GET\_AGENT\_MEMORY\_POOL.

## 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 115. 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 la base de datos
AGENT_ID	BIGINT	agent_id - Descriptor de contexto de aplicación (ID de agente)

Tabla 115. 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_PID	BIGINT	agent_pid - Unidad susceptible de envío del 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"> <li>• APP_GROUP</li> <li>• APPL_CONTROL</li> <li>• APPLICATION</li> <li>• BP</li> <li>• CAT_CACHE</li> <li>• DATABASE</li> <li>• DFM</li> <li>• FCMBP</li> <li>• IMPORT_POOL</li> <li>• LOCK_MGR</li> <li>• MONITOR</li> <li>• OTHER</li> <li>• PACKAGE_CACHE</li> <li>• QUERY</li> <li>• SHARED_SORT</li> <li>• SORT</li> <li>• SENTENCIA</li> <li>• STATISTICS</li> <li>• UTILITY</li> </ul>
POOL_CUR_SIZE	BIGINT	pool_cur_size - Tamaño actual de la agrupación de memoria
POOL_WATERMARK	BIGINT	pool_watermark - Nivel de agrupación de memoria
POOL_CONFIG_SIZE	BIGINT	pool_config_size - Tamaño configurado de la agrupación de memoria
DBPARTITIONNUM	SMALLINT	La 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 116 en la página 355 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL para la vista administrativa SNAPAPPL\_INFO y privilegio EXECUTE para la función de tabla SNAP\_GET\_APPL\_INFO\_V95.

### 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).

### Función de tabla SNAP\_GET\_APPL\_INFO\_V95

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 116 en la página 355 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. Si *nombrebd* no se establece en NULL y *núm\_partición\_bd* se establece en NULL, implícitamente se establece -1 para *núm\_partición\_bd*. Si no se utiliza esta opción de entrada, es decir, si sólo se proporciona *nombrebd*, se devuelven los datos de todas las particiones de base de datos.

Si tanto *nombrebd* como *núm\_partición\_bd* se establecen en NULL, los datos se intentarán leer desde el 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_APPL\_INFO\_V95.

## 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 si realiza un SELECT del 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 registros(s) seleccionado(s).

## Información devuelta

Tabla 116. 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 116. 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"> <li>• BACKUP</li> <li>• COMMIT_ACT</li> <li>• COMP</li> <li>• CONNECTED</li> <li>• CONNECTPEND</li> <li>• CREATE_DB</li> <li>• DECOUPLED</li> <li>• DISCONNECTPEND</li> <li>• INTR</li> <li>• IOERROR_WAIT</li> <li>• LOAD</li> <li>• LOCKWAIT</li> <li>• QUIESCE_TABLESPACE</li> <li>• RECOMP</li> <li>• REMOTE_RQST</li> <li>• RESTART</li> <li>• RESTORE</li> <li>• ROLLBACK_ACT</li> <li>• ROLLBACK_TO_SAVEPOINT</li> <li>• TEND</li> <li>• THABRT</li> <li>• THCOMT</li> <li>• TPREP</li> <li>• UNLOAD</li> <li>• UOWEXEC</li> <li>• UOWWAIT</li> <li>• WAITFOR_REMOTE</li> </ul>
CODEPAGE_ID	BIGINT	codepage_id - ID de página de códigos que utiliza la aplicación
NUM_ASSOC_AGENTS	BIGINT	num_assoc_agents - Número de agentes asociados
COORD_NODE_NUM	SMALLINT	coord_node - Nodo de coordinación

Tabla 116. 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 - Elemento del supervisor de niveles de autorización de 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"> <li>• BINDADD</li> <li>• CONNECT</li> <li>• CREATE_EXT_RT</li> <li>• CREATE_NOT_FENC</li> <li>• CREATETAB</li> <li>• DBADM</li> <li>• IMPLICIT_SCHEMA</li> <li>• LOAD</li> <li>• LIBADM</li> <li>• QUIESCE_CONN</li> <li>• SECADM</li> <li>• SYSADM</li> <li>• SYSCTRL</li> <li>• SYSMANT</li> <li>• SYSMON</li> <li>• SYSQUIESCE</li> </ul> <p>Los posibles valores para "fuente" son:</p> <ul style="list-style-type: none"> <li>• USER – autorización que se otorga al usuario o a un rol otorgado al usuario.</li> <li>• 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.</li> </ul>
CLIENT_PID	BIGINT	client_pid - ID de proceso del cliente
COORD_AGENT_PID	BIGINT	coord_agent_pid - Agente de coordinación
STATUS_CHANGE_TIME	TIMESTAMP	status_change_time - Hora de cambio de estado de aplicación

Tabla 116. 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_PLATFORM	VARCHAR(12)	client_platform - Plataforma operativa del cliente. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h, <ul style="list-style-type: none"> <li>• AIX</li> <li>• AIX64</li> <li>• AS400_DRDA</li> <li>• DOS</li> <li>• DYNIX</li> <li>• HP</li> <li>• HP64</li> <li>• HPIA</li> <li>• HPIA64</li> <li>• LINUX</li> <li>• LINUX390</li> <li>• LINUXIA64</li> <li>• LINUXPPC</li> <li>• LINUXPPC64</li> <li>• LINUXX8664</li> <li>• LINUXZ64</li> <li>• MAC</li> <li>• MVS_DRDA</li> <li>• NT</li> <li>• NT64</li> <li>• OS2</li> <li>• OS390</li> <li>• SCO</li> <li>• SGI</li> <li>• SNI</li> <li>• SUN</li> <li>• SUN64</li> <li>• UNKNOWN</li> <li>• UNKNOWN_DRDA</li> <li>• VM_DRDA</li> <li>• VSE_DRDA</li> <li>• WINDOWS</li> </ul>



Tabla 116. 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 comunicación del cliente. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h, <ul style="list-style-type: none"> <li>• CPIC</li> <li>• LOCAL</li> <li>• NETBIOS</li> <li>• NPIPE</li> <li>• TCPIP (para DB2 UDB)</li> <li>• TCPIP4</li> <li>• TCPIP6</li> </ul>
TERRITORY_CODE	SMALLINT	territory_code - Código de territorio de la 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 versión/producto del 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 que utiliza 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 del usuario
CORR_TOKEN	VARCHAR(128)	corr_token - Señal de correlación de DRDA
TPMON_CLIENT_USERID	VARCHAR(256)	tpmon_client_userid - ID de usuario del supervisor de TP
TPMON_CLIENT_WKSTN	VARCHAR(256)	tpmon_client_wkstn - Nombre de estación de trabajo cliente del supervisor de TP

Tabla 116. 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 cliente del supervisor de TP
TPMON_ACC_STR	VARCHAR(200)	tpmon_acc_str - Serie de contabilidad del cliente del supervisor de TP
DBPARTITIONNUM	SMALLINT	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_SYS_APPL	SMALLINT	El valor de IS_SYS_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 "Función de tabla SNAP\_GET\_APPL\_V95" en la página 361 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 117 en la página 362 para ver una lista completa de la información que se puede devolver.

## Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL en la vista administrativa SNAPAPPL y el privilegio EXECUTE en la función de tabla SNAP\_GET\_APPL\_V95.

## 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).

## Función de tabla SNAP\_GET\_APPL\_V95

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 117 en la página 362 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.

*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. Si *nombre\_bd* no se establece en NULL y *núm\_partición\_bd* se establece en NULL, implícitamente se establece -1 para *núm\_partición\_bd*. Si no se utiliza esta opción de entrada, es decir, si sólo se proporciona *nombre\_bd*, se devuelven los datos de todas las particiones de base de datos.

Si tanto *nombre\_bd* como *núm\_partición\_bd* se establecen en NULL, los datos se intentarán leer desde el 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_APPL\_V95.

### 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 117. 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)

Tabla 117. 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_LOG_SPACE_USED	BIGINT	uow_log_space_used - Espacio de registro de unidad 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 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_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
POOL_TEMP_INDEX_L_READS	BIGINT	pool_temp_index_l_reads - Lecturas lógicas de índice temporal de agrupación de almacenamientos intermedios
POOL_TEMP_INDEX_P_READS	BIGINT	pool_temp_index_p_reads - Grabaciones físicas de índice temporal de agrupación de almacenamientos intermedios
POOL_TEMP_XDA_L_READS	BIGINT	pool_temp_xda_l_reads - Lecturas lógicas de datos XSA temporales de agrupación de almacenamientos intermedios

Tabla 117. 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_TEMP_XDA_P_READS	BIGINT	pool_temp_xda_p_reads - Elemento de supervisor de lecturas físicas 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_XDA_P_READS	BIGINT	pool_xda_p_reads - Lecturas físicas de datos XDA 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
DIRECT_READS	BIGINT	direct_reads - Lecturas directas de base de datos
DIRECT_WRITES	BIGINT	direct_writes - Grabaciones directas en base de datos
DIRECT_READ_REQS	BIGINT	direct_read_reqs - Solicitudes de lectura directa
DIRECT_WRITE_REQS	BIGINT	direct_write_reqs - Solicitudes 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 de espera de bloqueos
LOCK_ESCALS	BIGINT	lock_escals - Número de reajustes de bloqueo
X_LOCK_ESCALS	BIGINT	x_lock_escals - Reajustes de bloqueo exclusivos
DEADLOCKS	BIGINT	deadlocks - Puntos muertos detectados
TOTAL_SORTS	BIGINT	total_sorts - Número total de clasificaciones

Tabla 117. 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
TOTAL_SORT_TIME	BIGINT	total_sort_time - Tiempo total de clasificación
SORT_OVERFLOWS	BIGINT	sort_overflows - Desbordamientos de clasificación
COMMIT_SQL_STMTS	BIGINT	commit_sql_stmts - Confirmaciones de sentencias intentadas
ROLLBACK_SQL_STMTS	BIGINT	rollback_sql_stmts - Retrotracciones de sentencias 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 sentencias fallidas
SELECT_SQL_STMTS	BIGINT	select_sql_stmts - Sentencias select de SQL ejecutadas
DDL_SQL_STMTS	BIGINT	ddl_sql_stmts - Sentencias Data Definition Language (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 debidas 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

Tabla 117. 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
REJ_CURS_BLK	BIGINT	rej_curs_blk - Solicitudes rechazadas de cursor de bloqueo
ACC_CURS_BLK	BIGINT	acc_curs_blk - Solicitudes aceptadas de cursor de bloqueo
SQL_REQS_SINCE_COMMIT	BIGINT	sql_reqs_since_commit - Solicitudes SQL desde última confirmación
LOCK_TIMEOUTS	BIGINT	lock_timeouts - Número de tiempos de espera excedidos 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 sustraídos
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 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"> <li>• DYNAMIC_PRIORITY</li> <li>• FIXED_PRIORITY</li> </ul>
PREFETCH_WAIT_TIME	BIGINT	prefetch_wait_time - Tiempo de espera de captación previa
APPL_SECTION_LOOKUPS	BIGINT	appl_section_lookups - Búsquedas de sección
APPL_SECTION_INSERTS	BIGINT	appl_section_inserts - Inserciones de sección



Tabla 117. 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
LOCKS_WAITING	BIGINT	locks_waiting - Agentes actuales en espera de bloqueos
TOTAL_HASH_JOINS	BIGINT	total_hash_joins - Número total de uniones de generación aleatoria
TOTAL_HASH_LOOPS	BIGINT	total_hash_loops - Número total de bucles de generación aleatoria
HASH_JOIN_OVERFLOWS	BIGINT	hash_join_overflows - Desbordamientos de unión de generación aleatoria
HASH_JOIN_SMALL_OVERFLOWS	BIGINT	hash_join_small_overflows - Desbordamientos pequeños de unión de generación aleatoria
APPL_IDLE_TIME	BIGINT	appl_idle_time - Tiempo de desocupación de aplicación
UOW_LOCK_WAIT_TIME	BIGINT	uow_lock_wait_time - Tiempo total de unidad de trabajo en espera de bloqueos
UOW_COMP_STATUS	VARCHAR(14)	uow_comp_status - Estado de finalización de 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"> <li>• APPL_END</li> <li>• UOWABEND</li> <li>• UOWCOMMIT</li> <li>• UOWDEADLOCK</li> <li>• UOWLOCKTIMEOUT</li> <li>• UOWROLLBACK</li> <li>• UOWUNKNOWN</li> </ul>
AGENT_USR_CPU_TIME_S	BIGINT	agent_usr_cpu_time - Tiempo de CPU de usuario empleado por el agente
AGENT_USR_CPU_TIME_MS	BIGINT	agent_usr_cpu_time - Tiempo de CPU de usuario empleado por el agente
AGENT_SYS_CPU_TIME_S	BIGINT	agent_sys_cpu_time - Tiempo de CPU del sistema empleado por el agente
AGENT_SYS_CPU_TIME_MS	BIGINT	agent_sys_cpu_time - Tiempo de CPU del sistema empleado por el agente
APPL_CON_TIME	TIMESTAMP	appl_con_time - Indicación de hora de inicio de solicitud de conexión
CONN_COMPLETE_TIME	TIMESTAMP	conn_complete_time - Indicación de hora de finalización de solicitud de conexión

Tabla 117. 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
LAST_RESET	TIMESTAMP	last_reset - Indicación de hora de último restablecimiento
UOW_START_TIME	TIMESTAMP	uow_start_time - Indicación de hora de inicio de unidad de trabajo
UOW_STOP_TIME	TIMESTAMP	uow_stop_time - Indicación de hora de finalización de unidad de trabajo
PREV_UOW_STOP_TIME	TIMESTAMP	prev_uow_stop_time - Indicación de 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 sentencias
ELAPSED_EXEC_TIME_MS	BIGINT	elapsed_exec_time - Tiempo transcurrido de ejecución de sentencias
INBOUND_COMM_ADDRESS	VARCHAR(32)	inbound_comm_address - Dirección de comunicación 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

Tabla 117. 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
SHR_WORKSPACE_SIZE_TOP	BIGINT	shr_workspace_size_top - Tamaño máximo de espacio de trabajo compartido
DBPARTITIONNUM	SMALLINT	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 118 en la página 371 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL en la vista administrativa SNAPBP y privilegio EXECUTE en la función de tabla SNAP\_GET\_BP\_V95.

### 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)

## Función de tabla SNAP\_GET\_BP\_V95

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 118 en la página 371 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. Si *nombrebd* no se establece en NULL y *núm\_partición\_bd* se establece en NULL, implícitamente se establece -1 para *núm\_partición\_bd*. Si no se utiliza esta opción de entrada, es decir, si sólo se proporciona *nombrebd*, se devuelven los datos de todas las particiones de base de datos.

Si tanto *nombrebd* como *núm\_partición\_bd* se establecen en NULL, los datos se intentarán leer desde el 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

- Autorización SYSMON
- Privilegio EXECUTE en la función de tabla SNAP\_GET\_BP\_V95.

## 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 registros(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```
... TOTAL_PHYSICAL_READS DBPARTITIONNUM
... -----
...                0                0
...                0                0
...                0                0
```

## Información devuelta

Tabla 118. 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.
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

Tabla 118. 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_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_XDA_L_READS	BIGINT	pool_xda_l_reads - Lecturas lógicas de datos XDA 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_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
POOL_ASYNC_DATA_READS	BIGINT	pool_async_data_reads - Lecturas de datos asíncronos de agrupación de almacenamientos intermedios
POOL_ASYNC_DATA_WRITES	BIGINT	pool_async_data_writes - Grabaciones de datos asíncronos de agrupación de almacenamientos intermedios
POOL_ASYNC_INDEX_READS	BIGINT	pool_async_index_reads - de índice asíncrono de agrupación de almacenamientos intermedios
POOL_ASYNC_INDEX_WRITES	BIGINT	pool_async_index_writes - Grabaciones de índice asíncrono de agrupación de almacenamientos intermedios

Tabla 118. 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_XDA_READS	BIGINT	pool_async_xda_reads - Lecturas de datos XDA asíncronos de agrupación de almacenamientos intermedios
POOL_ASYNC_XDA_WRITES	BIGINT	pool_async_xda_writes - Grabaciones de datos XDA asíncronos 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 - Solicitudes de lectura asíncrona de agrupación de almacenamientos intermedios
POOL_ASYNC_INDEX_READ_REQS	BIGINT	pool_async_index_read_reqs - Solicitudes de lectura de índice asíncrona de agrupación de almacenamientos intermedios
POOL_ASYNC_XDA_READ_REQS	BIGINT	pool_async_xda_read_reqs - Solicitudes de lectura XDA asíncrona de agrupación de almacenamientos intermedios
DIRECT_READS	BIGINT	direct_reads - Lecturas directas de base de datos
DIRECT_WRITES	BIGINT	direct_writes - Grabaciones directas en base de datos
DIRECT_READ_REQS	BIGINT	direct_read_reqs - Solicitudes de lectura directa
DIRECT_WRITE_REQS	BIGINT	direct_write_reqs - Solicitudes 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 base de datos cerrados
POOL_TEMP_DATA_L_READS	BIGINT	pool_temp_data_l_reads - Lecturas lógicas de datos temporales de agrupación de almacenamientos intermedios

Tabla 118. 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_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 índice temporal de agrupación de almacenamientos intermedios
POOL_TEMP_INDEX_P_READS	BIGINT	pool_temp_index_p_reads - Grabaciones físicas de índice temporal de agrupación de almacenamientos intermedios
POOL_TEMP_XDA_L_READS	BIGINT	pool_temp_xda_l_reads - Lecturas lógicas de datos XSA temporales de agrupación de almacenamientos intermedios
POOL_TEMP_XDA_P_READS	BIGINT	pool_temp_xda_p_reads - Elemento de supervisor de lecturas físicas de datos XDA temporales de agrupación de almacenamientos intermedios
POOL_NO_VICTIM_BUFFER	BIGINT	pool_no_victim_buffer - Almacenamientos intermedios no 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 por E/S vectorial
VECTORED_IOS	BIGINT	vectored_ios - Número de solicitudes de E/S vectorial
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 119 en la página 377 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL en la vista administrativa SNAPBP\_PART y el privilegio EXECUTE en la función de tabla SNAP\_GET\_BP\_PART.

### 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 119 en la página 377 para ver una lista completa de la información que se puede devolver.

## Sintaxis

```
▶▶ SNAP_GET_BP_PART ( ( nombre-bd [ , núm-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.

### *núm-partición-bd*

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. Si *nombre-bd* no se establece en NULL y *núm-partición-bd* se establece en NULL, implícitamente se establece -1 para *núm-partición-bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE en la función de tabla SNAP\_GET\_BP\_PART.

## 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 119. 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 la 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 tabla correlacionados con la agrupación de almacenamientos intermedios

Tabla 119. Información devuelta por la vista administrativa SNAPBP\_PART y la función de tabla SNAP\_GET\_BP\_PART (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 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 120 en la página 381 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL para la vista administrativa SNAPCONTAINER y privilegio EXECUTE para la función de tabla SNAP\_GET\_CONTAINER\_V91.

### 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

```

## Función de tabla SNAP\_GET\_CONTAINER\_V91

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 120 en la página 381 para ver una lista completa de la información que se puede devolver.

## Sintaxis

```

▶▶ SNAP_GET_CONTAINER_V91 ( ( nombre-bd [ , nm-particin-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 NULL o una serie vacía para tomar la instantánea desde la base de datos conectada actualmente.

*núm-partición-bd*

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. Si *nombrebd* no se establece en NULL y *núm-partición-bd* se establece en NULL, implícitamente se establece -1 para *núm-partición-bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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**

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_CONTAINER\_V91.

**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:\DGTsWhatsNewContainer	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
... 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 120. 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 del espacio de tablas
CONTAINER_NAME	VARCHAR(256)	container_name - Nombre del 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"> <li>• DISK_EXTENT_TAG</li> <li>• DISK_PAGE_TAG</li> <li>• FILE_EXTENT_TAG</li> <li>• FILE_PAGE_TAG</li> <li>• PATH</li> </ul>
TOTAL_PAGES	BIGINT	container_total_pages - Páginas totales en contenedor
USABLE_PAGES	BIGINT	container_usable_pages - Páginas útiles en contenedor
ACCESSIBLE	SMALLINT	container_accessible - Accesibilidad del contenedor
STRIPE_SET	BIGINT	container_stripe_set - Conjunto de bandas
DBPARTITIONNUM	SMALLINT	La 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

Tabla 120. 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_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 “Función de tabla SNAP\_GET\_DB\_V95” en la página 383 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 121 en la página 385 encontrará una lista completa de la información que se devuelve.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL para la vista administrativa SNAPDB y privilegio EXECUTE para la función de tabla SNAP\_GET\_DB\_V95.

### 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 registros(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:

```

db2 "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_FABRICATION_TIME SYNC_RUNSTATS_TIME
-----
              10              100

```

1 registro(s) seleccionado(s).

## **Función de tabla SNAP\_GET\_DB\_V95**

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 121 en la página 385 encontrará una lista completa de la información que se devuelve.

### **Sintaxis**

```

▶▶ SNAP_GET_DB_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.

*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. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, implícitamente se establece -1 para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, si sólo se proporciona *nombrebd*, se devuelven los datos de todas las particiones de base de datos.

Si tanto *nombrebd* como *númparticiónbd* se establecen en NULL, los datos se intentarán leer desde el 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_DB\_V95.

## 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:

```
db2 "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_FABRICATION_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 121. 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

Tabla 121. 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
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"> <li>• ACTIVE</li> <li>• QUIESCE_PEND</li> <li>• QUIESCED</li> <li>• ROLLFWD</li> </ul>
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 121. 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)	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: <ul style="list-style-type: none"> <li>• AIX</li> <li>• AIX64</li> <li>• AS400_DRDA</li> <li>• DOS</li> <li>• DYNIX</li> <li>• HP</li> <li>• HP64</li> <li>• HPIA</li> <li>• HPIA64</li> <li>• LINUX</li> <li>• LINUX390</li> <li>• LINUXIA64</li> <li>• LINUXPPC</li> <li>• LINUXPPC64</li> <li>• LINUXX8664</li> <li>• LINUXZ64</li> <li>• MAC</li> <li>• MVS_DRDA</li> <li>• NT</li> <li>• NT64</li> <li>• OS2</li> <li>• OS390</li> <li>• SCO</li> <li>• SGI</li> <li>• SNI</li> <li>• SUN</li> <li>• SUN64</li> <li>• UNKNOWN</li> <li>• UNKNOWN_DRDA</li> <li>• VM_DRDA</li> <li>• VSE_DRDA</li> <li>• WINDOWS</li> </ul>
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"> <li>• LOCAL</li> <li>• REMOTE</li> </ul>
DB_CONN_TIME	TIMESTAMP	db_conn_time - Indicación de la hora de activación de la base de datos

Tabla 121. 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 hora de último restablecimiento
LAST_BACKUP	TIMESTAMP	last_backup - Indicación de hora de última copia de seguridad
CONNECTIONS_TOP	BIGINT	connections_top - Número máximo de conexiones simultáneas
TOTAL_CONS	BIGINT	total_cons - Conexiones desde 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 en ejecución en la base de datos simultáneamente
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 de espera de bloqueos
LOCK_LIST_IN_USE	BIGINT	lock_list_in_use - Memoria total de lista de bloqueos utilizada
DEADLOCKS	BIGINT	deadlocks - Puntos muertos detectados
LOCK_ESCALS	BIGINT	lock_escals - Número de reajustes de bloqueo
X_LOCK_ESCALS	BIGINT	x_lock_escals - Reajustes de bloqueo exclusivos
LOCKS_WAITING	BIGINT	locks_waiting - Agentes actuales en espera de bloqueos
LOCK_TIMEOUTS	BIGINT	lock_timeouts - Número de tiempos de espera excedidos de bloqueo
NUM_INDOUBT_TRANS	BIGINT	num_indoubt_trans - Número de transacciones dudosas
SORT_HEAP_ALLOCATED	BIGINT	sort_heap_allocated - Pila total de clasificación asignada
SORT_SHRHEAP_ALLOCATED	BIGINT	sort_shrheap_allocated - Pila de compartición de clasificación actualmente asignada
SORT_SHRHEAP_TOP	BIGINT	sort_shrheap_top - Marca de límite superior de pila de compartición de clasificación
POST_SHRTHRESHOLD_SORTS	BIGINT	post_shrthreshold_sorts - Clasificaciones de umbral compartido posteriores
TOTAL_SORTS	BIGINT	total_sorts - Número total de clasificaciones
TOTAL_SORT_TIME	BIGINT	total_sort_time - Tiempo total de clasificación

Tabla 121. 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
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
POOL_ASYNC_DATA_READS	BIGINT	pool_async_data_reads - Lecturas de datos asíncronos 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 asíncronos 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 temporal de agrupación de almacenamientos intermedios
POOL_TEMP_INDEX_P_READS	BIGINT	pool_temp_index_p_reads - Grabaciones físicas de índice temporal de agrupación de almacenamientos intermedios
POOL_ASYNC_INDEX_READS	BIGINT	pool_async_index_reads - de índice asíncrono 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 asíncrono de agrupación de almacenamientos intermedios
POOL_XDA_P_READS	BIGINT	pool_xda_p_reads - Lecturas físicas de datos XDA de agrupación de almacenamientos intermedios

Tabla 121. 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_L_READS	BIGINT	pool_xda_l_reads - Lecturas lógicas de datos XDA de agrupación de almacenamientos intermedios
POOL_XDA_WRITES	BIGINT	pool_xda_writes - Grabaciones de datos XDA de agrupación de almacenamientos intermedios
POOL_ASYNC_XDA_READS	BIGINT	pool_async_xda_reads - Lecturas de datos XDA asíncronos de agrupación de almacenamientos intermedios
POOL_ASYNC_XDA_WRITES	BIGINT	pool_async_xda_writes - Grabaciones de datos XDA asíncronos de agrupación de almacenamientos intermedios
POOL_TEMP_XDA_P_READS	BIGINT	pool_temp_xda_p_reads - Elemento de supervisor de lecturas físicas de datos XDA temporales de agrupación de almacenamientos intermedios
POOL_TEMP_XDA_L_READS	BIGINT	pool_temp_xda_l_reads - Lecturas lógicas de datos XSA temporales 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
POOL_ASYNC_DATA_READ_REQS	BIGINT	pool_async_data_read_reqs - Solicitudes de lectura asíncrona de agrupación de almacenamientos intermedios
POOL_ASYNC_INDEX_READ_REQS	BIGINT	pool_async_index_read_reqs - Solicitudes de lectura de índice asíncrona de agrupación de almacenamientos intermedios
POOL_ASYNC_XDA_READ_REQS	BIGINT	pool_async_xda_read_reqs - Solicitudes de lectura XDA asíncrona de agrupación de almacenamientos intermedios
POOL_NO_VICTIM_BUFFER	BIGINT	pool_no_victim_buffer - Almacenamientos intermedios no víctimas de la agrupación de almacenamientos intermedios
POOL_LSN_GAP_CLNS	BIGINT	pool_lsn_gap_clns - Limpiadores desencadenados del espacio de registro de la agrupación de almacenamientos intermedios



Tabla 121. 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_DRTY_PG_STEAL_CLNS	BIGINT	pool_drty_pg_steal_clns - Limpiadores de páginas víctimas desencadenados de la agrupación de almacenamientos intermedios
POOL_DRTY_PG_THRSH_CLNS	BIGINT	pool_drty_pg_thrsh_clns - Limpiadores desencadenados de umbrales de agrupación de almacenamientos intermedios
PREFETCH_WAIT_TIME	BIGINT	prefetch_wait_time - Tiempo de espera de captación previa
UNREAD_PREFETCH_PAGES	BIGINT	unread_prefetch_pages - Páginas de captación previa no leídas
DIRECT_READS	BIGINT	direct_reads - Lecturas directas de base de datos
DIRECT_WRITES	BIGINT	direct_writes - Grabaciones directas en base de datos
DIRECT_READ_REQS	BIGINT	direct_read_reqs - Solicitudes de lectura directa
DIRECT_WRITE_REQS	BIGINT	direct_write_reqs - Solicitudes 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 sentencias
ELAPSED_EXEC_TIME_MS	BIGINT	elapsed_exec_time - Tiempo transcurrido de ejecución de sentencias
COMMIT_SQL_STMTS	BIGINT	commit_sql_stmts - Confirmaciones de sentencias intentadas
ROLLBACK_SQL_STMTS	BIGINT	rollback_sql_stmts - Retrotracciones de sentencias 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 sentencias fallidas
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 Data Definition Language (DDL) de SQL
INT_AUTO_REBINDS	BIGINT	int_auto_rebinds - Revinculaciones automáticas internas

Tabla 121. 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
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 debidas 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 - Registro total disponible
TOTAL_LOG_USED	BIGINT	total_log_used - Espacio total de registro utilizado
SEC_LOG_USED_TOP	BIGINT	sec_log_used_top - Espacio máximo de registro secundario utilizado
TOT_LOG_USED_TOP	BIGINT	tot_log_used_top - Espacio máximo total de registro utilizado
SEC_LOGS_ALLOCATED	BIGINT	sec_logs_allocated - Registros secundarios asignados actualmente
LOG_READS	BIGINT	log_reads - Número de páginas de registro leídas
LOG_READ_TIME_S	BIGINT	log_read_time - Tiempo de lectura de registro
LOG_READ_TIME_NS	BIGINT	log_read_time - Tiempo de lectura de registro
LOG_WRITES	BIGINT	log_writes - Número de páginas de registro grabadas
LOG_WRITE_TIME_S	BIGINT	log_write_time - Tiempo de grabación de registro
LOG_WRITE_TIME_NS	BIGINT	log_write_time - Tiempo de grabación de registro
NUM_LOG_WRITE_IO	BIGINT	num_log_write_io - Número de grabaciones de registro
NUM_LOG_READ_IO	BIGINT	num_log_read_io - Número de lecturas de registro
NUM_LOG_PART_PAGE_IO	BIGINT	num_log_part_page_io - Número de grabaciones parciales de páginas de registro
NUM_LOG_BUFFER_FULL	BIGINT	num_log_buffer_full - Número de almacenamientos intermedios de registro llenos

Tabla 121. 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_DATA_FOUND_IN_BUFFER	BIGINT	num_log_data_found_in_buffer - Número de datos de registro encontrados en almacenamiento intermedio
APPL_ID_OLDEST_XACT	BIGINT	appl_id_oldest_xact - Aplicación con transacción más antigua
LOG_TO_REDO_FOR_RECOVERY	BIGINT	log_to_redo_for_recovery - Cantidad de registro que hay que rehacer para recuperación
LOG_HELD_BY_DIRTY_PAGES	BIGINT	log_held_by_dirty_pages - Cantidad de espacio de registro 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
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

Tabla 121. 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
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 - Número total de uniones de generación aleatoria
TOTAL_HASH_LOOPS	BIGINT	total_hash_loops - Número total de bucles de generación aleatoria
HASH_JOIN_OVERFLOWS	BIGINT	hash_join_overflows - Desbordamientos de unión de generación aleatoria
HASH_JOIN_SMALL_OVERFLOWS	BIGINT	hash_join_small_overflows - Desbordamientos pequeños de unión de generación aleatoria
POST_SHRTHRESHOLD_HASH_JOINS	BIGINT	post_shrthreshold_hash_joins - Uniones de generación aleatoria de umbral posteriores
ACTIVE_HASH_JOINS	BIGINT	active_hash_joins - Uniones activas de generación aleatoria
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 menor cantidad de espacio de registro 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
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.

Tabla 121. 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_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.

### SNAPDB\_MEMORY\_POOL, vista administrativa

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 122 en la página 398 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL en la vista administrativa SNAPDB\_MEMORY\_POOL y privilegio EXECUTE en la función de tabla SNAP\_GET\_DB\_MEMORY\_POOL.

### 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).

## Función de tabla SNAP\_GET\_DB\_MEMORY\_POOL

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 122 en la página 398 para ver una lista completa de la información que se puede devolver.

## Sintaxis

```

▶▶ SNAP_GET_DB_MEMORY_POOL ( ( nombre-bd [ , nm-particin-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 vlido 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 vaca para tomar la instantnea desde la base de datos conectada actualmente. Especifique un valor NULL para tomar la instantnea desde todas las bases de datos que se encuentran en la misma instancia que la base de datos conectada actualmente.

### nm-particin-bd

Argumento de entrada opcional de tipo INTEGER que especifica un nmero de particin de base de datos vlido. Especifique -1 para la particin de base de datos actual o -2 para un agregado de todas las particiones de base de datos. Si nombrebd no se establece en NULL y nm\_particin\_bd se establece en

NULL, implícitamente se establece -1 para *núm\_partición\_bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre\_bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre\_bd* y *núm-partición\_bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE en la función de tabla SNAP\_GET\_DB\_MEMORY\_POOL.

## 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 122. 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 la base de datos
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"> <li>• APP_GROUP</li> <li>• APPL_CONTROL</li> <li>• APPLICATION</li> <li>• BP</li> <li>• CAT_CACHE</li> <li>• DATABASE</li> <li>• DFM</li> <li>• FCMBP</li> <li>• IMPORT_POOL</li> <li>• LOCK_MGR</li> <li>• MONITOR</li> <li>• OTHER</li> <li>• PACKAGE_CACHE</li> <li>• QUERY</li> <li>• SHARED_SORT</li> <li>• SORT</li> <li>• SENTENCIA</li> <li>• STATISTICS</li> <li>• UTILITY</li> </ul>
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 - Nivel de agrupación de memoria
POOL_CONFIG_SIZE	BIGINT	pool_config_size - Tamaño configurado de la agrupación de memoria
DBPARTITIONNUM	SMALLINT	La 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.

### SNAPDBM, vista administrativa

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 123 en la página 400 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL en la vista administrativa SNAPDBM y el privilegio EXECUTE en la función de tabla SNAP\_GET\_DBM\_V95.

### 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 registros(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
```

### Función de tabla SNAP\_GET\_DBM\_V95

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 123 para ver una lista completa de la información que se puede devolver.

## Sintaxis

```
▶▶ SNAP_GET_DBM_V95 ( [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. Si no se utiliza esta opción de entrada, todas las particiones de la base de datos devolverán datos.

Si *número\_partición\_bd* se establece en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_DBM\_V95.

## 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 123. 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.

Tabla 123. 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
SORT_HEAP_ALLOCATED	BIGINT	sort_heap_allocated - Pila total de clasificación asignada
POST_THRESHOLD_SORTS	BIGINT	post_threshold_sorts - Clasificaciones de umbral posteriores
PIPED_SORTS_REQUESTED	BIGINT	piped_sorts_requested - Clasificaciones conducidas solicitadas
PIPED_SORTS_ACCEPTED	BIGINT	piped_sorts_accepted - Clasificaciones conducidas 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 supervisor del 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 supervisor del 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 a la espera de un símbolo
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"> <li>• ACTIVE</li> <li>• QUIESCE_PEND</li> <li>• QUIESCED</li> </ul>
AGENTS_REGISTERED_TOP	BIGINT	agents_registered_top - Número máximo de agentes registrados
AGENTS_WAITING_TOP	BIGINT	agents_waiting_top - Número máximo de agentes a la 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 desde agrupación
AGENTS_CREATED_EMPTY_POOL	BIGINT	agents_created_empty_pool - Agentes creados debido a una agrupación de agentes vacía
COORD_AGENTS_TOP	BIGINT	coord_agents_top - Número máximo de agentes de coordinación

Tabla 123. 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
MAX_AGENT_OVERFLOWS	BIGINT	max_agent_overflows - Número máximo de desbordamientos de agente
AGENTS_STOLEN	BIGINT	agents_stolen - Agentes sustraídos
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 a la espera de que el sistema principal responda
GW_CONS_WAIT_CLIENT	BIGINT	gw_cons_wait_client - Número de conexiones a la espera de que el cliente envíe una solicitud
POST_THRESHOLD_HASH_JOINS	BIGINT	post_threshold_hash_joins - Umbral de uniones de generación aleatoria
NUM_GW_CONN_SWITCHES	BIGINT	num_gw_conn_switches - Conmutadores de conexión
DB2START_TIME	TIMESTAMP	db2start_time - Indicación de hora de inicio del gestor de bases de datos
LAST_RESET	TIMESTAMP	last_reset - Indicación de hora de último restablecimiento
NUM_NODES_IN_DB2_INSTANCE	INTEGER	num_nodes_in_db2_instance - Número de nodos de 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.

Tabla 123. 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
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>La 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.

### SNAPDBM\_MEMORY\_POOL, vista administrativa

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 124 en la página 405 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON

- Privilegio SELECT o CONTROL en la vista administrativa SNAPDBM\_MEMORY\_POOL y privilegio EXECUTE en la función de tabla SNAP\_GET\_DBM\_MEMORY\_POOL.

## 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
...	

## Función de tabla SNAP\_GET\_DBM\_MEMORY\_POOL

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 124 en la página 405 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 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. Si no se utiliza esta opción de entrada, todas las particiones de la base de datos devolverán datos.

Si *núm-partición-bd* se establece en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE en la función de tabla SNAP\_GET\_DBM\_MEMORY\_POOL.

## 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 124. 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.

Tabla 124. Información que devuelve la vista administrativa `SNAPDBM_MEMORY_POOL` y la función de tabla `SNAP_GET_DBM_MEMORY_POOL` (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
<code>POOL_ID</code>	<code>VARCHAR(14)</code>	<p><code>pool_id</code> - Identificador de agrupación de memoria. Esta interfaz devuelve un identificador de texto basado en las sentencias define de <code>sqlmon.h</code> y es uno de los siguientes:</p> <ul style="list-style-type: none"> <li>• <code>APP_GROUP</code></li> <li>• <code>APPL_CONTROL</code></li> <li>• <code>APPLICATION</code></li> <li>• <code>BP</code></li> <li>• <code>CAT_CACHE</code></li> <li>• <code>DATABASE</code></li> <li>• <code>DFM</code></li> <li>• <code>FCMBP</code></li> <li>• <code>IMPORT_POOL</code></li> <li>• <code>LOCK_MGR</code></li> <li>• <code>MONITOR</code></li> <li>• <code>OTHER</code></li> <li>• <code>PACKAGE_CACHE</code></li> <li>• <code>QUERY</code></li> <li>• <code>SHARED_SORT</code></li> <li>• <code>SORT</code></li> <li>• <code>SENTENCIA</code></li> <li>• <code>STATISTICS</code></li> <li>• <code>UTILITY</code></li> </ul>
<code>POOL_CUR_SIZE</code>	<code>BIGINT</code>	<code>pool_cur_size</code> - Tamaño actual de la agrupación de memoria
<code>POOL_WATERMARK</code>	<code>BIGINT</code>	<code>pool_watermark</code> - Nivel de agrupación de memoria
<code>POOL_CONFIG_SIZE</code>	<code>BIGINT</code>	<code>pool_config_size</code> - Tamaño configurado de la agrupación de memoria
<code>DBPARTITIONNUM</code>	<code>SMALLINT</code>	La 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 desde el 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, SNAPHADR 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 125 en la página 409 encontrará una lista completa de la información que se devuelve.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL para la vista administrativa SNAPDETAILLOG y privilegio EXECUTE para la función de tabla SNAP\_GET\_DETAILLOG\_V91.

### 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 registros(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 125 en la página 409 encontrará una lista completa de la información que se devuelve.

## Sintaxis

```
▶▶ SNAP_GET_DETAILLOG_V91 (—nombre-bd [ , núm-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.

### *núm-partición-bd*

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. Si *nombrebd* no se establece en NULL y *núm-partición-bd* se establece en NULL, implícitamente se establece -1 para *núm-partición-bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_DETAILLOG\_V91.

## 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 125. 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 la base de datos
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	La 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 “vista administrativa SNAPDYN\_SQL” y la “Función de tabla SNAP\_GET\_DYN\_SQL\_V95” en la página 411 devuelven información de instantánea desde el grupo de datos lógicos dynsql.

#### vista administrativa SNAPDYN\_SQL

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 126 en la página 413 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL para la vista administrativa SNAPDYN\_SQL y privilegio EXECUTE para la función de tabla SNAP\_GET\_SQL\_V95.

### 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 126 en la página 413 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 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. Si *nombrebd* no se establece en NULL y *núm\_partición\_bd* se establece en NULL, implícitamente se establece -1 para *núm\_partición\_bd*. Si no se utiliza esta opción de entrada, es decir, si sólo se proporciona *nombrebd*, se devuelven los datos de todas las particiones de base de datos.

Si tanto *nombrebd* como *núm\_partición\_bd* se establecen en NULL, los datos se intentarán leer desde el 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**

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_DYN\_SQL\_V95.

## 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 126. Información devuelta por la vista administrativa SNAPDYN\_SQL y la función de tabla SNAP\_GET\_DYN\_SQL\_V95*

<b>Nombre de columna</b>	<b>Tipo de datos</b>	<b>Descripción o elemento del supervisor correspondiente</b>
SNAPSHOT_TIMESTAMP	TIMESTAMP	La fecha y la hora en que se tomó la instantánea.
NUM_EXECUTIONS	BIGINT	num_executions - Ejecuciones de sentencias
NUM_COMPILATIONS	BIGINT	num_compilations - Compilaciones de sentencias
PREP_TIME_WORST	BIGINT	prep_time_worst - Peor tiempo de preparación de sentencias
PREP_TIME_BEST	BIGINT	prep_time_best - Mejor tiempo de preparación de sentencias
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 total de clasificación
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
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 temporal de agrupación de almacenamientos intermedios

Tabla 126. 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_P_READS	BIGINT	pool_temp_index_p_reads - Grabaciones físicas de índice temporal 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_XDA_P_READS	BIGINT	pool_xda_p_reads - Lecturas físicas de datos XDA de agrupación de almacenamientos intermedios
POOL_TEMP_XDA_L_READS	BIGINT	pool_temp_xda_l_reads - Lecturas lógicas de datos XSA temporales de agrupación de almacenamientos intermedios
POOL_TEMP_XDA_P_READS	BIGINT	pool_temp_xda_p_reads - Elemento de supervisor de lecturas físicas de datos XDA temporales de agrupación de almacenamientos intermedios
TOTAL_EXEC_TIME	BIGINT	total_exec_time - Tiempo transcurrido de ejecución de sentencias
TOTAL_EXEC_TIME_MS	BIGINT	total_exec_time - Tiempo transcurrido de ejecución de sentencias
TOTAL_USR_CPU_TIME	BIGINT	total_usr_cpu_time - CPU total de usuario para una sentencia
TOTAL_USR_CPU_TIME_MS	BIGINT	total_usr_cpu_time - CPU total de usuario para una sentencia
TOTAL_SYS_CPU_TIME	BIGINT	total_sys_cpu_time - CPU total del sistema para una sentencia
TOTAL_SYS_CPU_TIME_MS	BIGINT	total_sys_cpu_time - CPU total del sistema para una sentencia
STMT_TEXT	CLOB(2 M)	stmt_text - Texto de sentencia 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 127 en la página 416 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL para la vista administrativa SNAPFCM y privilegio EXECUTE para la función de tabla SNAP\_GET\_FCM.

### 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

### Función de tabla SNAP\_GET\_FCM

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 127 en la página 416 para ver una lista completa de la información que se puede devolver.

## Sintaxis

```
▶▶ SNAP_GET_FCM ( [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 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. Si no se utiliza esta opción de entrada, todas las particiones de base de datos devolverán los datos.

Si *número-partición-bd* se establece en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_FCM.

## 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 127. 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

Tabla 127. Información que devuelve la vista administrativa SNAPFCM y la función de tabla SNAP\_GET\_FCM (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
CH_FREE	BIGINT	ch_free - Canales libres actualmente
CH_FREE_BOTTOM	BIGINT	ch_free_bottom - Número mínimo de canales libres
DBPARTITIONNUM	SMALLINT	La 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 128 en la página 419 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL en la vista administrativa SNAPFCM\_PART y privilegio EXECUTE en la función de tabla SNAP\_GET\_FCM\_PART.

### 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).

## Función de tabla SNAP\_GET\_FCM\_PART

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 128 en la página 419 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 de partición de base de datos válido. Especifique -1 para la partición actual o -2 para un agregado de todas las particiones. Si no se utiliza esta opción de entrada, todas las particiones devolverán los datos.

Si *núm-partición-bd* se establece en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE en la función de tabla SNAP\_GET\_FCM\_PART.

### 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 128. 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 basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> <li>• INACTIVE</li> <li>• ACTIVE</li> <li>• CONGESTED</li> </ul>
TOTAL_BUFFERS_SENT	BIGINT	total_buffers_sent - Total de almacenamientos intermedios de FCM enviados
TOTAL_BUFFERS_RCVD	BIGINT	total_buffers_rcvd - Total de almacenamientos intermedios de FCM recibidos
DBPARTITIONNUM	SMALLINT	La 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 129 en la página 421 para ver una lista completa de la información que se puede devolver.

## Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL en la vista administrativa SNAPHADR y el privilegio EXECUTE en la función de tabla SNAP\_GET\_HADR.

## 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).

## Función de tabla SNAP\_GET\_HADR

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 129 en la página 421 para ver una lista completa de la información que se puede devolver.

## Sintaxis

```
→ SNAP_GET_HADR (—nombre-bd— [ , núm-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.

*núm-partición-bd*

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. Si *nombrebd* no se establece en NULL y *núm-partición-bd* se establece en NULL, implícitamente se establece -1 para *núm-partición-bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE en la función de tabla SNAP\_GET\_HADR.

### 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 129. 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 la base de datos

Tabla 129. 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_ROLE	VARCHAR(10)	hadr_role - Función 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"> <li>• PRIMARY</li> <li>• STANDARD</li> <li>• STANDBY</li> </ul>
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"> <li>• DISCONNECTED</li> <li>• LOCAL_CATCHUP</li> <li>• PEER</li> <li>• REM_CATCH_PEN</li> <li>• REM_CATCHUP</li> </ul>
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"> <li>• ASYNC</li> <li>• NEARSYNC</li> <li>• SYNC</li> </ul>
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"> <li>• CONGESTED</li> <li>• CONNECTED</li> <li>• DISCONNECTED</li> </ul>
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



Tabla 129. 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_REMOTE_INSTANCE	VARCHAR(128)	had_r_remote_instance - Instancia remota de HADR
HADR_TIMEOUT	BIGINT	had_r_timeout - Tiempo de espera de HADR
HADR_PRIMARY_LOG_FILE	VARCHAR(255)	had_r_primary_log_file - Archivo de anotaciones cronológicas primario de HADR
HADR_PRIMARY_LOG_PAGE	BIGINT	had_r_primary_log_page - Página de anotaciones cronológicas primarias de HADR
HADR_PRIMARY_LOG_LSN	BIGINT	had_r_primary_log_lsn - LSN de anotaciones cronológicas primarias de HADR
HADR_STANDBY_LOG_FILE	VARCHAR(255)	had_r_standby_log_file - Archivo de anotaciones cronológica en espera de HADR
HADR_STANDBY_LOG_PAGE	BIGINT	had_r_standby_log_page - Página de anotaciones cronológicas en espera de HADR
HADR_STANDBY_LOG_LSN	BIGINT	had_r_standby_log_lsn - LSN de anotaciones cronológicas en espera de HADR
HADR_LOG_GAP	BIGINT	had_r_log_gap - Distancia de anotación cronológica de HADR
DBPARTITIONNUM	SMALLINT	La 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 130 en la página 425 para ver una lista completa de la información que se puede devolver.

## Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL en la vista administrativa SNAPLOCK y privilegio EXECUTE en la función de tabla SNAP\_GET\_LOCK.

## 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).

## Función de tabla SNAP\_GET\_LOCK

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 130 en la página 425 para ver una lista completa de la información que se puede devolver.

## Sintaxis

```
▶▶ SNAP_GET_LOCK ( ( nombre-bd [ , núm-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 un valor nulo o una serie vacía para tomar la instantánea desde la base de datos conectada actualmente.

*núm-partición-bd*

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. Si *nombrebd* no se establece en NULL y *núm\_partición\_bd* se establece en NULL, implícitamente se establece -1 para *núm\_partición\_bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_LOCK.

## 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)) 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 130. 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 130. 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 basado en las sentencias define de sqlmon.h y es uno de los siguientes:</p> <ul style="list-style-type: none"> <li>• AUTORESIZE_LOCK</li> <li>• AUTOSTORAGE_LOCK</li> <li>• BLOCK_LOCK</li> <li>• EOT_LOCK</li> <li>• INPLACE_REORG_LOCK</li> <li>• INTERNAL_LOCK</li> <li>• INTERNALB_LOCK</li> <li>• INTERNALC_LOCK</li> <li>• INTERNALJ_LOCK</li> <li>• INTERNALL_LOCK</li> <li>• INTERNALO_LOCK</li> <li>• INTERNALQ_LOCK</li> <li>• INTERNALP_LOCK</li> <li>• INTERNALS_LOCK</li> <li>• INTERNALT_LOCK</li> <li>• INTERNALV_LOCK</li> <li>• KEYVALUE_LOCK</li> <li>• ROW_LOCK</li> <li>• SYSBOOT_LOCK</li> <li>• TABLE_LOCK</li> <li>• TABLE_PART_LOCK</li> <li>• TABLESPACE_LOCK</li> <li>• XML_PATH_LOCK</li> </ul>

Tabla 130. 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_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"> <li>• IN</li> <li>• IS</li> <li>• IX</li> <li>• NON (si no existe bloqueo)</li> <li>• NS</li> <li>• NW</li> <li>• NX</li> <li>• S</li> <li>• SIX</li> <li>• U</li> <li>• W</li> <li>• X</li> <li>• Z</li> </ul>
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"> <li>• CONV</li> <li>• GRNT</li> </ul>
LOCK_ESCALATION	SMALLINT	lock_escalation - Escala de bloqueo
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

Tabla 130. 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_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"> <li>• ALLOW_NEW</li> <li>• DELETE_IN_BLOCK</li> <li>• ESCALATED</li> <li>• INSERT</li> <li>• NEW_REQUEST</li> <li>• RR</li> <li>• RR_IN_BLOCK</li> <li>• UPDATE_DELETE</li> <li>• WAIT_FOR_AVAIL</li> </ul>
LOCK_COUNT	BIGINT	lock_count - Número de bloqueos
LOCK_CURRENT_MODE	VARCHAR(10)	lock_current_mode - Modalidad de bloqueo original antes de la conversión. 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"> <li>• IN</li> <li>• IS</li> <li>• IX</li> <li>• NON (si no existe bloqueo)</li> <li>• NS</li> <li>• NW</li> <li>• NX</li> <li>• S</li> <li>• SIX</li> <li>• U</li> <li>• W</li> <li>• X</li> <li>• Z</li> </ul>
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.

Tabla 130. 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
DBPARTITIONNUM	SMALLINT	La 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 131 en la página 431 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL en la vista administrativa SNAPLOCKWAIT y privilegio EXECUTE en la función de tabla SNAP\_GET\_LOCKWAIT.

### 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
```

## Función de tabla SNAP\_GET\_LOCKWAIT

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 131 en la página 431 para ver una lista completa de la información que se puede devolver.

### Sintaxis

```
▶▶ SNAP_GET_LOCKWAIT ( nombre-bd [ , núm-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 un valor nulo o una serie vacía para tomar la instantánea desde la base de datos conectada actualmente.

#### *núm-partición-bd*

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. Si *nombrebd* no se establece en NULL y *núm-partición-bd* se establece en NULL, implícitamente se establece -1 para *núm-partición-bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_LOCKWAIT.



## 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 131. 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 131. 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 basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> <li>• IN</li> <li>• IS</li> <li>• IX</li> <li>• NON (si no existe bloqueo)</li> <li>• NS</li> <li>• NW</li> <li>• NX</li> <li>• S</li> <li>• SIX</li> <li>• U</li> <li>• W</li> <li>• X</li> <li>• Z</li> </ul>

Tabla 131. 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_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"> <li>• AUTORESIZE_LOCK</li> <li>• AUTOSTORAGE_LOCK</li> <li>• BLOCK_LOCK</li> <li>• EOT_LOCK</li> <li>• INPLACE_REORG_LOCK</li> <li>• INTERNAL_LOCK</li> <li>• INTERNALB_LOCK</li> <li>• INTERNALC_LOCK</li> <li>• INTERNALJ_LOCK</li> <li>• INTERNALL_LOCK</li> <li>• INTERNALO_LOCK</li> <li>• INTERNALQ_LOCK</li> <li>• INTERNALP_LOCK</li> <li>• INTERNALS_LOCK</li> <li>• INTERNALT_LOCK</li> <li>• INTERNALV_LOCK</li> <li>• KEYVALUE_LOCK</li> <li>• ROW_LOCK</li> <li>• SYSBOOT_LOCK</li> <li>• TABLE_LOCK</li> <li>• TABLE_PART_LOCK</li> <li>• TABLESPACE_LOCK</li> <li>• XML_PATH_LOCK</li> </ul>
AGENT_ID_HOLDING_LK	BIGINT	agent_id_holding_lock - Bloqueo de retención de ID de agente
LOCK_WAIT_START_TIME	TIMESTAMP	lock_wait_start_time - Indicación de la hora de inicio de espera de bloqueo

Tabla 131. 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_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"> <li>• IN</li> <li>• IS</li> <li>• IX</li> <li>• NON (si no existe bloqueo)</li> <li>• NS</li> <li>• NW</li> <li>• NX</li> <li>• S</li> <li>• SIX</li> <li>• U</li> <li>• W</li> <li>• X</li> <li>• Z</li> </ul>
LOCK_ESCALATION	SMALLINT	lock_escalation - Escala de bloqueo
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
APPL_ID_HOLDING_LK	VARCHAR(128)	appl_id_holding_lk - Bloqueo de retención de ID de aplicación
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"> <li>• ALLOW_NEW</li> <li>• DELETE_IN_BLOCK</li> <li>• ESCALATED</li> <li>• INSERT</li> <li>• NEW_REQUEST</li> <li>• RR</li> <li>• RR_IN_BLOCK</li> <li>• UPDATE_DELETE</li> <li>• WAIT_FOR_AVAIL</li> </ul>

Tabla 131. 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_CURRENT_MODE	VARCHAR(10)	lock_current_mode - Modalidad de bloqueo original antes de la conversión. 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"> <li>• IN</li> <li>• IS</li> <li>• IX</li> <li>• NON (si no existe bloqueo)</li> <li>• NS</li> <li>• NW</li> <li>• NX</li> <li>• S</li> <li>• SIX</li> <li>• U</li> <li>• W</li> <li>• X</li> <li>• Z</li> </ul>
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	La 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 132 en la página 438 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL en la vista administrativa SNAPSTMT y el privilegio EXECUTE en la función de tabla SNAP\_GET\_STMT.

### 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).

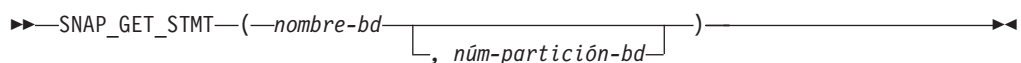
### Función de tabla SNAP\_GET\_STMT

La función de tabla SNAP\_GET\_STMT devuelve la misma información que la vista administrativa SNAPSTMT, 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 132 en la página 438 para ver una lista completa de la información que se puede devolver.

### Sintaxis



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.

### *núm-partición-bd*

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. Si *nombrebd* no se establece en NULL y *núm\_partición\_bd* se establece en NULL, implícitamente se establece -1 para *núm\_partición\_bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE en la función de tabla SNAP\_GET\_STMT.

## 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 132. 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.
DB_NAME	VARCHAR(128)	db_name - Nombre de la 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 que trabaja 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"> <li>• DYNAMIC</li> <li>• NON_STMT</li> <li>• STATIC</li> <li>• STMT_TYPE_UNKNOWN</li> </ul>



Tabla 132. 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"> <li>• CALL</li> <li>• CLOSE</li> <li>• COMPILE</li> <li>• DESCRIBE</li> <li>• EXECUTE</li> <li>• EXECUTE_IMMEDIATE</li> <li>• FETCH</li> <li>• FREE_LOCATOR</li> <li>• GETAA</li> <li>• GETNEXTCHUNK</li> <li>• GETTA</li> <li>• NONE</li> <li>• OPEN</li> <li>• PREP_COMMIT</li> <li>• PREP_EXEC</li> <li>• PREP_OPEN</li> <li>• PREPARE</li> <li>• REBIND</li> <li>• REDIST</li> <li>• REORG</li> <li>• RUNSTATS</li> <li>• SELECT</li> <li>• SET</li> <li>• STATIC_COMMIT</li> <li>• STATIC_ROLLBACK</li> </ul>
SECTION_NUMBER	BIGINT	section_number - Número de sección
QUERY_COST_ESTIMATE	BIGINT	query_cost_estimate - Estimación del coste de la consulta
QUERY_CARD_ESTIMATE	BIGINT	query_card_estimate - Número estimado de consulta de las 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 total de clasificación
SORT_OVERFLOW	BIGINT	sort_overflows - Clasificar desbordamientos

Tabla 132. 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 la hora de inicio de operación de sentencia
STMT_STOP	TIMESTAMP	stmt_stop - Indicación de la hora de detención de operación de sentencia
STMT_USR_CPU_TIME_S	BIGINT	stmt_usr_cpu_time - Tiempo de CPU de usuario que la sentencia utiliza
STMT_USR_CPU_TIME_MS	BIGINT	stmt_usr_cpu_time - Tiempo de CPU de usuario que la sentencia utiliza
STMT_SYS_CPU_TIME_S	BIGINT	stmt_sys_cpu_time - Tiempo de CPU de sistema que la sentencia utiliza
STMT_SYS_CPU_TIME_MS	BIGINT	stmt_sys_cpu_time - Tiempo de CPU de sistema que la sentencia utiliza
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 132. 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	La 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

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.

### Vista administrativa SNAPSTORAGE\_PATHS

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 133 en la página 444 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.

### 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).

### Función de tabla SNAP\_GET\_STORAGE\_PATHS

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 133 en la página 444 para ver una lista completa de la información que se puede devolver.

## Sintaxis

```
▶▶ SNAP_GET_STORAGE_PATHS ( (nombre-bd [ , núm-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.

### *núm-partición-bd*

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. Si *nombrebd* no se establece en NULL y *núm\_partición\_bd* se establece en NULL, implícitamente se establece -1 para *núm\_partición\_bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_STORAGE\_PATHS.

## 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 133. 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 la base de datos
DB_STORAGE_PATH	VARCHAR(256)	db_storage_path - Vía de acceso de almacenamiento automático
DBPARTITIONNUM	SMALLINT	La 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 134 en la página 446 para ver una lista completa de la información que se puede devolver.

## Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL para la vista administrativa SNAPSUBSECTION y privilegio EXECUTE en la función de tabla SNAP\_GET\_SUBSECTION.

## 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

## Función de tabla SNAP\_GET\_SUBSECTION

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 134 en la página 446 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 ( ( nombre-bd [ , nm-particin-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.

### *núm-partición-bd*

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. Si *nombrebd* no se establece en NULL y *núm\_partición\_bd* se establece en NULL, implícitamente se establece -1 para *núm\_partición\_bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_SUBSECTION.

## 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 134. 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.



Tabla 134. 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
DB_NAME	VARCHAR(128)	db_name - Nombre de la 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 tablas desbordados
TQ_CUR_SEND_SPILLS	BIGINT	tq_cur_send_spills - Número actual de almacenamientos intermedios de cola de tablas desbordados
TQ_MAX_SEND_SPILLS	BIGINT	tq_max_send_spills - Número máximo de desbordamientos de almacenamientos intermedios de cola de tablas
TQ_ROWS_READ	BIGINT	tq_rows_read - Número de filas leídas desde las colas de tablas
TQ_ROWS_WRITTEN	BIGINT	tq_rows_written - Número de filas grabadas en las colas de tablas
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 que la subsección utiliza
SS_USR_CPU_TIME_MS	BIGINT	ss_usr_cpu_time - Tiempo de CPU de usuario que la subsección utiliza
SS_SYS_CPU_TIME_S	BIGINT	ss_sys_cpu_time - Tiempo de CPU del sistema que la subsección utiliza
SS_SYS_CPU_TIME_MS	BIGINT	ss_sys_cpu_time - Tiempo de CPU del sistema que la subsección utiliza
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"> <li>• EXEC</li> <li>• TQ_WAIT_TO_RCV</li> <li>• TQ_WAIT_TO_SEND</li> <li>• COMPLETED</li> </ul>
SS_NODE_NUMBER	SMALLINT	ss_node_number - Número de nodo de subsección
TQ_NODE_WAITED_FOR	SMALLINT	tq_node_waited_for - En espera de nodo en una cola de tablas

Tabla 134. 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_WAIT_FOR_ANY	INTEGER	tq_wait_for_any - En espera de que cualquier nodo se envíe en una cola de tablas
TQ_ID_WAITING_ON	INTEGER	tq_id_waiting_on - En espera de nodo en una cola de tablas
DBPARTITIONNUM	SMALLINT	La 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 acerca del estado del conmutador de instantánea 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 135 en la página 450 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL para la vista administrativa SNAPSWITCHES y privilegio EXECUTE para la función de tabla SNAP\_GET\_SWITCHES.

### 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

## Función de tabla SNAP\_GET\_SWITCHES

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 135 en la página 450 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 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. Si no se utiliza esta opción de entrada, todas las particiones de la base de datos devolverán datos.

Si *núm-partición-bd* se establece en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_SWITCHES.

## 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 135. 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.
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.

Tabla 135. Información que devuelve la vista administrativa SNAPSWITCHES y la función de tabla SNAP\_GET\_SWITCHES (continuación)

Nombre de columna	Tipo de datos	Descripción
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	La 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 136 en la página 453 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL para la vista administrativa SNAPTAB y privilegio EXECUTE para la función de tabla SNAP\_GET\_TAB\_V91.

### 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).

## Función de tabla SNAP\_GET\_TAB\_V91

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 136 en la página 453 para ver una lista completa de la información que se puede devolver.

## Sintaxis

```

▶▶ SNAP_GET_TAB_V91 (—nombre-bd— [ , núm-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 NULL o una serie vacía para tomar la instantánea desde la base de datos conectada actualmente.

### núm-partición-bd

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. Si *nombrebd* no se establece en NULL y *núm\_partición\_bd* se establece en NULL, implícitamente se establece -1 para *núm\_partición\_bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_TAB\_V91.

## 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 136. 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"> <li>• USER_TABLE</li> <li>• DROPPED_TABLE</li> <li>• TEMP_TABLE</li> <li>• CATALOG_TABLE</li> <li>• REORG_TABLE</li> </ul>
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

Tabla 136. 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
OVERFLOW_ACCESSES	BIGINT	overflow_accesses - Accesos a los registros desbordados
PAGE_REORGS	BIGINT	page_reorgs - Reorganizaciones de página
DBPARTITIONNUM	SMALLINT	La partición de base de datos desde la que se han recuperado los datos para esta fila.
TBSP_ID	BIGINT	tablespace_id - Identificación del 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 137 en la página 456 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL para la vista administrativa SNAPTAB\_REORG y privilegio EXECUTE para la función de tabla SNAP\_GET\_TAB\_REORG.

### 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 registros(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

## **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 137 en la página 456 para ver una lista completa de la información que se puede devolver.

## **Sintaxis**

```

▶▶ SNAP_GET_TAB_REORG ( ( nombre-bd [ , nm-particin-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 vlido 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 vaca para tomar la instantnea desde la base de datos conectada actualmente.

### *nm-particin-bd*

Argumento de entrada opcional de tipo INTEGER que especifica un nmero de particin de base de datos vlido. Especifique -1 para la particin de base de datos actual o -2 para un agregado de todas las particiones de base de datos. Si *nombrebd* no se establece en NULL y *nm-particin-bd* se establece en NULL, implcitamente se establece -1 para *nm-particin-bd*. Si no se utiliza esta opcin de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_TAB\_REORG.

## 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(TABSCHEMA, 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
...
...
```

## Información devuelta

Tabla 137. 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

Tabla 137. 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_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"> <li>• BUILD</li> <li>• DICT_SAMPLE</li> <li>• INDEX_RECREATE</li> <li>• REPLACE</li> <li>• SORT</li> </ul> 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 137. 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"> <li>• RECLAIM</li> <li>• RECLUSTER</li> </ul> <p>y uno de los siguientes:</p> <ul style="list-style-type: none"> <li>• +OFFLINE</li> <li>• +ONLINE</li> </ul> <p>Si se especifica la modalidad de acceso, es uno de los siguientes:</p> <ul style="list-style-type: none"> <li>• +ALLOW_NONE</li> <li>• +ALLOW_READ</li> <li>• +ALLOW_WRITE</li> </ul> <p>Si está fuera de línea y con la opción RECLUSTER, uno de los siguientes:</p> <ul style="list-style-type: none"> <li>• +INDEXSCAN</li> <li>• +TABLESCAN</li> </ul> <p>Si está fuera de línea, uno de los siguientes:</p> <ul style="list-style-type: none"> <li>• +LONGLOB</li> <li>• +DATAONLY</li> </ul> <p>Si está fuera de línea, y se especifica una opción, uno de los siguientes:</p> <ul style="list-style-type: none"> <li>• +CHOOSE_TEMP</li> <li>• +KEEPDICTIONARY</li> <li>• +RESETDICTIONARY</li> </ul> <p>Si está en línea, y se especifica una opción:</p> <ul style="list-style-type: none"> <li>• +NOTRUNCATE</li> </ul> <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 137. 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"> <li>• COMPLETED</li> <li>• PAUSED</li> <li>• STARTED</li> <li>• STOPPED</li> <li>• TRUNCATE</li> </ul>
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"> <li>• FAIL</li> <li>• SUCCESS</li> </ul>
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 tabla donde se reorganiza la tabla
DBPARTITIONNUM	SMALLINT	La 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 SNAPTBSPP y función de tabla SNAP\_GET\_TBSP\_V91 – Recuperar información de instantáneas del grupo de datos lógicos de espacio de tablas

La vista administrativa SNAPTBSPP 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 138 en la página 462 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL para la vista administrativa SNAPTbsp y privilegio EXECUTE para la función de tabla SNAP\_GET\_TBSP\_V91.

### 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).

### Función de tabla SNAP\_GET\_TBSP\_V91

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 138 en la página 462 para ver una lista completa de la información que se puede devolver.

## Sintaxis

```
▶▶ SNAP_GET_TBSP_V91 ( ( nombre-bd [ , núm-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 NULL o una serie vacía para tomar la instantánea desde la base de datos conectada actualmente.

### *núm-partición-bd*

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. Si *nombrebd* no se establece en NULL y *núm-partición-bd* se establece en NULL, implícitamente se establece -1 para *núm-partición-bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_TBSP\_V91.

## 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 138. Información devuelta por la vista administrativa *SNAPT BSP* 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 del 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"> <li>• DMS</li> <li>• SMS</li> </ul>
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"> <li>• ANY</li> <li>• LARGE</li> <li>• SYSTEMP</li> <li>• USRTEMP</li> </ul>
TBSP_PAGE_SIZE	BIGINT	tablespace_page_size - Tamaño de página de espacio de tabla
TBSP_EXTENT_SIZE	BIGINT	tablespace_extent_size - Tamaño de extensión de espacio de tabla
TBSP_PREFETCH_SIZE	BIGINT	tablespace_prefetch_size - Tamaño de captación previa de espacio de tabla
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



Tabla 138. 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_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 la 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 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
POOL_ASYNC_INDEX_READS	BIGINT	pool_async_index_reads - Lecturas de índices asíncronas de la 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 índices asíncronas de la agrupación de almacenamientos intermedios

Tabla 138. 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_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 la agrupación de almacenamientos intermedios
POOL_ASYNC_XDA_WRITES	BIGINT	pool_async_xda_writes - Grabaciones de datos XDA asíncronas 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_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 la agrupación de almacenamientos intermedios
POOL_ASYNC_WRITE_TIME	BIGINT	pool_async_write_time - Tiempo de grabación asíncrona de la agrupación de almacenamientos intermedios
POOL_ASYNC_DATA_READ_REQS	BIGINT	pool_async_data_read_reqs - Peticiones de lectura asíncrona de la agrupación de almacenamientos intermedios
POOL_ASYNC_INDEX_READ_REQS	BIGINT	pool_async_index_read_reqs - Peticiones de lectura de índices asíncronos de la agrupación de almacenamientos intermedios

Tabla 138. 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_XDA_READ_REQS	BIGINT	pool_async_xda_reqs - Peticiones de lectura XDA asíncronas de la agrupación de almacenamientos intermedios
POOL_NO_VICTIM_BUFFER	BIGINT	pool_no_victim_buffer - Almacenamientos intermedios sin víctimas de la agrupación de almacenamientos intermedios
DIRECT_READS	BIGINT	direct_reads - Lecturas directas de base de datos
DIRECT_WRITES	BIGINT	direct_writes - Grabaciones directas en 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
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"> <li>• NO_REBAL</li> <li>• FWD_REBAL</li> <li>• REV_REBAL</li> </ul>
TBSP_USING_AUTO_STORAGE	SMALLINT	tablespace_using_auto_storage - Uso del almacenamiento automático
TBSP_AUTO_RESIZE_ENABLED	SMALLINT	tablespace_auto_resize_enabled - Dimensionamiento automático habilitado
DBPARTITIONNUM	SMALLINT	La 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

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 139 en la página 468 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\_V91.

### 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).

### Función de tabla SNAP\_GET\_TBSP\_PART\_V91

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 139 en la página 468 para ver una lista completa de la información que se puede devolver.

## Sintaxis

►► SNAP\_GET\_TBSP\_PART\_V91 (—*nombre-bd*—, *núm-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 NULL o una serie vacía para tomar la instantánea desde la base de datos conectada actualmente.

### *núm-partición-bd*

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. Si *nombrebd* no se establece en NULL y *núm\_partición\_bd* se establece en NULL, implícitamente se establece -1 para *núm\_partición\_bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_TBSP\_PART\_V91.

## 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 139. 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 del espacio de tablas

Tabla 139. Información devuelta por la vista administrativa *SNAPT BSP\_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 tabla. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlutil.h, y es una combinación de los siguientes elementos, separados por un signo '+':</p> <ul style="list-style-type: none"> <li>• BACKUP_IN_PROGRESS</li> <li>• BACKUP_PENDING</li> <li>• DELETE_PENDING</li> <li>• DISABLE_PENDING</li> <li>• DROP_PENDING</li> <li>• LOAD_IN_PROGRESS</li> <li>• LOAD_PENDING</li> <li>• NORMAL</li> <li>• OFFLINE</li> <li>• PSTAT_CREATION</li> <li>• PSTAT_DELETION</li> <li>• QUIESCED_EXCLUSIVE</li> <li>• QUIESCED_SHARE</li> <li>• QUIESCED_UPDATE</li> <li>• REBAL_IN_PROGRESS</li> <li>• REORG_IN_PROGRESS</li> <li>• RESTORE_IN_PROGRESS</li> <li>• RESTORE_PENDING</li> <li>• ROLLFORWARD_IN_PROGRESS</li> <li>• ROLLFORWARD_PENDING</li> <li>• STORDEF_ALLOWED</li> <li>• STORDEF_CHANGED</li> <li>• STORDEF_FINAL_VERSION</li> <li>• STORDEF_PENDING</li> <li>• SUSPEND_WRITE</li> </ul>
TBSP_PREFETCH_SIZE	BIGINT	tablespace_prefetch_size - Tamaño de captación previa de espacio de tabla
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 tabla de cambio de estado
TBSP_MIN_RECOVERY_TIME	TIMESTAMP	tablespace_min_recovery_time - Tiempo de recuperación mínimo para avance
TBSP_TOTAL_PAGES	BIGINT	tablespace_total_pages - Páginas totales en espacio de tabla

Tabla 139. 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 útiles en espacio de tabla
TBSP_USED_PAGES	BIGINT	tablespace_used_pages - Páginas utilizadas en espacio de tabla
TBSP_FREE_PAGES	BIGINT	tablespace_free_pages - Páginas libres en espacio de tabla
TBSP_PENDING_FREE_PAGES	BIGINT	tablespace_pending_free_pages - Páginas libres pendientes en espacio de tabla
TBSP_PAGE_TOP	BIGINT	tablespace_page_top - Nivel límite de espacio de tabla
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"> <li>• FWD_REBAL</li> <li>• NO_REBAL</li> <li>• REV_REBAL</li> </ul>
REBALANCER_EXTENTS_REMAINING	BIGINT	tablespace_rebalancer_extents_remaining - Número total de extensiones que ha de procesar el reequilibrador
REBALANCER_EXTENTS_PROCESSED	BIGINT	tablespace_rebalancer_extents_processed - Número de extensiones que el reequilibrador ha procesado
REBALANCER_PRIORITY	BIGINT	tablespace_rebalancer_priority - Prioridad del 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 en la correlación del espacio de tabla
TBSP_NUM_CONTAINERS	BIGINT	tablespace_num_containers - Número de contenedores en espacio de tabla
TBSP_INITIAL_SIZE	BIGINT	tablespace_initial_size - Tamaño inicial de espacio de tabla
TBSP_CURRENT_SIZE	BIGINT	tablespace_current_size - Tamaño actual de espacio de tabla
TBSP_MAX_SIZE	BIGINT	tablespace_max_size - Tamaño máximo de espacio de tabla



Tabla 139. 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	La 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 140 en la página 473 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización `SYSMON`
- Privilegio `SELECT` o `CONTROL` para la vista administrativa `SNAPTbsp_QUIESCER` y privilegio `EXECUTE` para la función de tabla `SNAP_Get_Tbsp_QUIESCER`.

### 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
...		

## 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 140 en la página 473 para ver una lista completa de la información que se puede devolver.

## Sintaxis

```
→ SNAP_GET_TBSP QUIESCER ( nombre-bd [ , nm-particin-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 vlido 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 vaca para tomar la instantnea desde la base de datos conectada actualmente.

### nm-particin-bd

Argumento de entrada opcional de tipo INTEGER que especifica un nmero de particin de base de datos vlido. Especifique -1 para la particin de base de datos actual o -2 para un agregado de todas las particiones de base de

datos. Si *nombrebd* no se establece en NULL y *núm\_partición\_bd* se establece en NULL, implícitamente se establece -1 para *núm\_partición\_bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_TBSP QUIESCER.

### 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 140. Información devuelta por la vista administrativa SNAP\_TBSP 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

Tabla 140. Información devuelta por la vista administrativa *SNAPTbsp\_QUIESCER* y la función de tabla *SNAP\_GET\_TBSP\_QUIESCER* (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
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 sqlutil.h y es uno de los siguientes: <ul style="list-style-type: none"> <li>• EXCLUSIVE</li> <li>• UPDATE</li> <li>• SHARE</li> </ul>
DBPARTITIONNUM	SMALLINT	La 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 141 en la página 477 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL para la vista administrativa *SNAPTbsp\_RANGE* y privilegio EXECUTE para la función de tabla *SNAP\_GET\_TBSP\_RANGE*.

## 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\_TBSP\_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 141 en la página 477 para ver una lista completa de la información que se puede devolver.

## Sintaxis

```
▶▶ SNAP_GET_TBSP_RANGE (—nombre-bd— [ , núm-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 NULL o una serie vacía para tomar la instantánea desde la base de datos conectada actualmente.

### *núm-partición-bd*

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. Si *nombrebd* no se establece en NULL y *núm-partición-bd* se establece en NULL, implícitamente se establece -1 para *núm-partición-bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_TBSP\_RANGE.

## Ejemplos

Seleccionar información acerca del rango de espacio de tablas para el espacio de tablas con *tblsp\_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 141. 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 del 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

Tabla 141. Información devuelta por la vista administrativa SNAPTBSP\_RANGE y la función de tabla SNAP\_GET\_TBSP\_RANGE (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 esta fila.

## Vista administrativa SNAPUTIL y función de tabla SNAP\_GET\_UTIL – Recuperar información acerca de la 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 142 en la página 480 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL para la vista administrativa SNAPUTIL y privilegio EXECUTE para la función de tabla SNAP\_GET\_UTIL.

### 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 registros(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

```

## Función de tabla SNAP\_GET\_UTIL

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 142 en la página 480 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 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. Si no se utiliza esta opción de entrada, todas las particiones de la base de datos devolverán datos.

Si *núm\_partición\_bd* se establece en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_UTIL.

## 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 142. 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 basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> <li>• ASYNC_INDEX_CLEANUP</li> <li>• BACKUP</li> <li>• CRASH_RECOVERY</li> <li>• LOAD</li> <li>• REBALANCE</li> <li>• REDISTRIBUTE</li> <li>• REORG</li> <li>• RESTART_RECREATE_INDEX</li> <li>• RESTORE</li> <li>• ROLLFORWARD_RECOVERY</li> <li>• RUNSTATS</li> </ul>
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.
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

Tabla 142. 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_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"> <li>• ERROR</li> <li>• EXECUTE</li> <li>• WAIT</li> </ul>
UTILITY_INVOKER_TYPE	VARCHAR(10)	utility_invoker_type - Tipo de invocador 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"> <li>• AUTO</li> <li>• USER</li> </ul>
DBPARTITIONNUM	SMALLINT	La 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 acerca de la 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 143 en la página 483 para ver una lista completa de la información que se puede devolver.

## Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL para la vista administrativa SNAPUTIL\_PROGRESS y privilegio EXECUTE para la función de tabla SNAP\_GET\_UTIL\_PROGRESS.

## 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).

## Función de tabla SNAP\_GET\_UTIL\_PROGRESS

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 143 en la página 483 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 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. Si no se utiliza esta opción de entrada, todas las particiones de la base de datos devolverán datos.

Si *núm-partición-bd* se establece en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_UTIL\_PROGRESS.

## 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 143. 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 basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> <li>• ERROR</li> <li>• EXECUTE</li> <li>• WAIT</li> </ul>
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.

Tabla 143. Información devuelta por la vista administrativa SNAPUTIL\_PROGRESS y la función de tabla SNAP\_GET\_UTIL\_PROGRESS (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
PROGRESS_WORK_METRIC	VARCHAR(16)	progress_work_metric - Métrica de trabajo en curso. 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"> <li>• NOT_SUPPORT</li> <li>• BYTES</li> <li>• EXTENTS</li> <li>• INDEXES</li> <li>• PAGES</li> <li>• ROWS</li> <li>• TABLES</li> </ul>
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	La 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—,—núm-partición-bd—)————◄◄
```

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, que pueden ser uno de los siguientes:

- APPL\_ALL
- BUFFERPOOLS\_ALL
- DB2
- DBASE\_ALL
- DBASE\_LOCKS
- DBASE\_TABLES
- DBASE\_TABLESPACES

- DYNAMIC\_SQL

*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 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 desde la base de datos conectada actualmente.

*núm-partición-bd*

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. Si se especifica un valor nulo, se establece -1 de forma implícita.

## Autorización

Para ejecutar el procedimiento, el usuario debe tener autorización SYSADM, SYSCTRL, SYSMAINT o SYSMON. Los usuarios que no tengan autorización SYSADM, SYSCTRL, SYSMAINT 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 sqllib/tmp/SQLMA\_DB2.dat en los sistemas operativos UNIX y sqllib\DB2\tmp\SQLMA\_DB2.dat en un sistema operativo Windows.

## Notas sobre 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.

### SNAPAGENT, vista administrativa

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 114 en la página 348 para ver una lista completa de la información que se puede devolver.

## Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL en la vista administrativa SNAPAGENT y el privilegio EXECUTE en la función de tabla SNAP\_GET\_AGENT.

## 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
...			
...			

## Función de tabla SNAP\_GET\_AGENT

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 114 en la página 348 para ver una lista completa de la información que se puede devolver.



## Sintaxis

```
▶▶ SNAP_GET_AGENT ( (nombre-bd [ , núm-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.

### *núm-partición-bd*

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. Si *nombrebd* no se establece en NULL y *núm\_partición\_bd* se establece en NULL, implícitamente se establece -1 para *núm\_partición\_bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE en la función de tabla SNAP\_GET\_AGENT.

## 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 144. 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 la 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 del motor (EDU)
LOCK_TIMEOUT_VAL	BIGINT	lock_timeout_val - Tiempo 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.

### SNAPAGENT\_MEMORY\_POOL, vista administrativa

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 and SNAPSUBSECTION, ofrece información equivalente al mandato CLP GET SNAPSHOT FOR APPLICATIONS ON alias de base de datos.

El esquema es SYSIBMADM.

Consulte la Tabla 115 en la página 351 para ver una lista completa de la información que se puede devolver.

## Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL en la vista administrativa SNAPAGENT\_MEMORY\_POOL y privilegio EXECUTE en la función de tabla SNAP\_GET\_AGENT\_MEMORY\_POOL.

## 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).

## Función de tabla SNAP\_GET\_AGENT\_MEMORY\_POOL

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 115 en la página 351 para ver una lista completa de la información que se puede devolver.

## Sintaxis

```

▶▶ SNAP_GET_AGENT_MEMORY_POOL ( ( nombre-bd
                                [ , nóm-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.

*núm-partición-bd*

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. Si *nombrebd* no se establece en NULL y *núm\_partición\_bd* se establece en NULL, implícitamente se establece -1 para *núm\_partición\_bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE en la función de tabla SNAP\_GET\_AGENT\_MEMORY\_POOL.

### 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 145. 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 la 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 del 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"> <li>• APP_GROUP</li> <li>• APPL_CONTROL</li> <li>• APPLICATION</li> <li>• BP</li> <li>• CAT_CACHE</li> <li>• DATABASE</li> <li>• DFM</li> <li>• FCMBP</li> <li>• IMPORT_POOL</li> <li>• LOCK_MGR</li> <li>• MONITOR</li> <li>• OTHER</li> <li>• PACKAGE_CACHE</li> <li>• QUERY</li> <li>• SHARED_SORT</li> <li>• SORT</li> <li>• SENTENCIA</li> <li>• STATISTICS</li> <li>• UTILITY</li> </ul>
POOL_CUR_SIZE	BIGINT	pool_cur_size - Tamaño actual de la agrupación de memoria
POOL_WATERMARK	BIGINT	pool_watermark - Nivel de agrupación de memoria
POOL_CONFIG_SIZE	BIGINT	pool_config_size - Tamaño configurado de la agrupación de memoria

Tabla 145. 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
DBPARTITIONNUM	SMALLINT	La 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 116 en la página 355 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL para la vista administrativa SNAPAPPL\_INFO y privilegio EXECUTE para la función de tabla SNAP\_GET\_APPL\_INFO\_V95.

### 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).

## Función de tabla SNAP\_GET\_APPL\_INFO\_V95

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 116 en la página 355 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. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, implícitamente se establece -1 para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, si sólo se proporciona *nombrebd*, se devuelven los datos de todas las particiones de base de datos.

Si tanto *nombrebd* como *númparticiónbd* se establecen en NULL, los datos se intentarán leer desde el 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_APPL\_INFO\_V95.

## 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 si realiza un SELECT del 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 registros(s) seleccionado(s).

## Información devuelta

Tabla 146. 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 146. 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)	<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"> <li>• BACKUP</li> <li>• COMMIT_ACT</li> <li>• COMP</li> <li>• CONNECTED</li> <li>• CONNECTPEND</li> <li>• CREATE_DB</li> <li>• DECOUPLED</li> <li>• DISCONNECTPEND</li> <li>• INTR</li> <li>• IOERROR_WAIT</li> <li>• LOAD</li> <li>• LOCKWAIT</li> <li>• QUIESCE_TABLESPACE</li> <li>• RECOMP</li> <li>• REMOTE_RQST</li> <li>• RESTART</li> <li>• RESTORE</li> <li>• ROLLBACK_ACT</li> <li>• ROLLBACK_TO_SAVEPOINT</li> <li>• TEND</li> <li>• THABRT</li> <li>• THCOMT</li> <li>• TPREP</li> <li>• UNLOAD</li> <li>• UOWEXEC</li> <li>• UOWWAIT</li> <li>• WAITFOR_REMOTE</li> </ul>
CODEPAGE_ID	BIGINT	codepage_id - ID de página de códigos que utiliza la aplicación
NUM_ASSOC_AGENTS	BIGINT	num_assoc_agents - Número de agentes asociados
COORD_NODE_NUM	SMALLINT	coord_node - Nodo de coordinación

Tabla 146. 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 - Elemento del supervisor de niveles de autorización de 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"> <li>• BINDADD</li> <li>• CONNECT</li> <li>• CREATE_EXT_RT</li> <li>• CREATE_NOT_FENC</li> <li>• CREATETAB</li> <li>• DBADM</li> <li>• IMPLICIT_SCHEMA</li> <li>• LOAD</li> <li>• LIBADM</li> <li>• QUIESCE_CONN</li> <li>• SECADM</li> <li>• SYSADM</li> <li>• SYSCTRL</li> <li>• SYSMANT</li> <li>• SYSMON</li> <li>• SYSQUIESCE</li> </ul> <p>Los posibles valores para "fuente" son:</p> <ul style="list-style-type: none"> <li>• USER – autorización que se otorga al usuario o a un rol otorgado al usuario.</li> <li>• 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.</li> </ul>
CLIENT_PID	BIGINT	client_pid - ID de proceso del cliente
COORD_AGENT_PID	BIGINT	coord_agent_pid - Agente de coordinación
STATUS_CHANGE_TIME	TIMESTAMP	status_change_time - Hora de cambio de estado de aplicación

Tabla 146. 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_PLATFORM	VARCHAR(12)	client_platform - Plataforma operativa del cliente. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h, <ul style="list-style-type: none"> <li>• AIX</li> <li>• AIX64</li> <li>• AS400_DRDA</li> <li>• DOS</li> <li>• DYNIX</li> <li>• HP</li> <li>• HP64</li> <li>• HPIA</li> <li>• HPIA64</li> <li>• LINUX</li> <li>• LINUX390</li> <li>• LINUXIA64</li> <li>• LINUXPPC</li> <li>• LINUXPPC64</li> <li>• LINUXX8664</li> <li>• LINUXZ64</li> <li>• MAC</li> <li>• MVS_DRDA</li> <li>• NT</li> <li>• NT64</li> <li>• OS2</li> <li>• OS390</li> <li>• SCO</li> <li>• SGI</li> <li>• SNI</li> <li>• SUN</li> <li>• SUN64</li> <li>• UNKNOWN</li> <li>• UNKNOWN_DRDA</li> <li>• VM_DRDA</li> <li>• VSE_DRDA</li> <li>• WINDOWS</li> </ul>

Tabla 146. 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 comunicación del cliente. Esta interfaz devuelve un identificador de texto basado en las sentencias define de sqlmon.h, <ul style="list-style-type: none"> <li>• CPIC</li> <li>• LOCAL</li> <li>• NETBIOS</li> <li>• NPIPE</li> <li>• TCPIP (para DB2 UDB)</li> <li>• TCPIP4</li> <li>• TCPIP6</li> </ul>
TERRITORY_CODE	SMALLINT	territory_code - Código de territorio de la 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 versión/producto del 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 que utiliza 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 del usuario
CORR_TOKEN	VARCHAR(128)	corr_token - Señal de correlación de DRDA
TPMON_CLIENT_USERID	VARCHAR(256)	tpmon_client_userid - ID de usuario del supervisor de TP
TPMON_CLIENT_WKSTN	VARCHAR(256)	tpmon_client_wkstn - Nombre de estación de trabajo cliente del supervisor de TP

Tabla 146. 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 cliente del supervisor de TP
TPMON_ACC_STR	VARCHAR(200)	tpmon_acc_str - Serie de contabilidad del cliente del supervisor de TP
DBPARTITIONNUM	SMALLINT	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_SYS_APPL	SMALLINT	El valor de IS_SYS_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 360 y la "Función de tabla SNAP\_GET\_APPL\_V95" en la página 361 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 117 en la página 362 para ver una lista completa de la información que se puede devolver.

## Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL en la vista administrativa SNAPAPPL y el privilegio EXECUTE en la función de tabla SNAP\_GET\_APPL\_V95.

## 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).

## Función de tabla SNAP\_GET\_APPL\_V95

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 117 en la página 362 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.

*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. Si *nombre\_bd* no se establece en NULL y *núm\_partición\_bd* se establece en NULL, implícitamente se establece -1 para *núm\_partición\_bd*. Si no se utiliza esta opción de entrada, es decir, si sólo se proporciona *nombre\_bd*, se devuelven los datos de todas las particiones de base de datos.

Si tanto *nombre\_bd* como *núm\_partición\_bd* se establecen en NULL, los datos se intentarán leer desde el 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_APPL\_V95.

### 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 147. 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)

Tabla 147. 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_LOG_SPACE_USED	BIGINT	uow_log_space_used - Espacio de registro de unidad 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 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_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
POOL_TEMP_INDEX_L_READS	BIGINT	pool_temp_index_l_reads - Lecturas lógicas de índice temporal de agrupación de almacenamientos intermedios
POOL_TEMP_INDEX_P_READS	BIGINT	pool_temp_index_p_reads - Grabaciones físicas de índice temporal de agrupación de almacenamientos intermedios
POOL_TEMP_XDA_L_READS	BIGINT	pool_temp_xda_l_reads - Lecturas lógicas de datos XSA temporales de agrupación de almacenamientos intermedios



Tabla 147. 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_TEMP_XDA_P_READS	BIGINT	pool_temp_xda_p_reads - Elemento de supervisor de lecturas físicas 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_XDA_P_READS	BIGINT	pool_xda_p_reads - Lecturas físicas de datos XDA 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
DIRECT_READS	BIGINT	direct_reads - Lecturas directas de base de datos
DIRECT_WRITES	BIGINT	direct_writes - Grabaciones directas en base de datos
DIRECT_READ_REQS	BIGINT	direct_read_reqs - Solicitudes de lectura directa
DIRECT_WRITE_REQS	BIGINT	direct_write_reqs - Solicitudes 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 de espera de bloqueos
LOCK_ESCALS	BIGINT	lock_escals - Número de reajustes de bloqueo
X_LOCK_ESCALS	BIGINT	x_lock_escals - Reajustes de bloqueo exclusivos
DEADLOCKS	BIGINT	deadlocks - Puntos muertos detectados
TOTAL_SORTS	BIGINT	total_sorts - Número total de clasificaciones

Tabla 147. 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
TOTAL_SORT_TIME	BIGINT	total_sort_time - Tiempo total de clasificación
SORT_OVERFLOWS	BIGINT	sort_overflows - Desbordamientos de clasificación
COMMIT_SQL_STMTS	BIGINT	commit_sql_stmts - Confirmaciones de sentencias intentadas
ROLLBACK_SQL_STMTS	BIGINT	rollback_sql_stmts - Retrotracciones de sentencias 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 sentencias fallidas
SELECT_SQL_STMTS	BIGINT	select_sql_stmts - Sentencias select de SQL ejecutadas
DDL_SQL_STMTS	BIGINT	ddl_sql_stmts - Sentencias Data Definition Language (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 debidas 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

Tabla 147. 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
REJ_CURS_BLK	BIGINT	rej_curs_blk - Solicitudes rechazadas de cursor de bloqueo
ACC_CURS_BLK	BIGINT	acc_curs_blk - Solicitudes aceptadas de cursor de bloqueo
SQL_REQS_SINCE_COMMIT	BIGINT	sql_reqs_since_commit - Solicitudes SQL desde última confirmación
LOCK_TIMEOUTS	BIGINT	lock_timeouts - Número de tiempos de espera excedidos 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 sustraídos
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 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"> <li>• DYNAMIC_PRIORITY</li> <li>• FIXED_PRIORITY</li> </ul>
PREFETCH_WAIT_TIME	BIGINT	prefetch_wait_time - Tiempo de espera de captación previa
APPL_SECTION_LOOKUPS	BIGINT	appl_section_lookups - Búsquedas de sección
APPL_SECTION_INSERTS	BIGINT	appl_section_inserts - Inserciones de sección

Tabla 147. 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
LOCKS_WAITING	BIGINT	locks_waiting - Agentes actuales en espera de bloqueos
TOTAL_HASH_JOINS	BIGINT	total_hash_joins - Número total de uniones de generación aleatoria
TOTAL_HASH_LOOPS	BIGINT	total_hash_loops - Número total de bucles de generación aleatoria
HASH_JOIN_OVERFLOWS	BIGINT	hash_join_overflows - Desbordamientos de unión de generación aleatoria
HASH_JOIN_SMALL_OVERFLOWS	BIGINT	hash_join_small_overflows - Desbordamientos pequeños de unión de generación aleatoria
APPL_IDLE_TIME	BIGINT	appl_idle_time - Tiempo de desocupación de aplicación
UOW_LOCK_WAIT_TIME	BIGINT	uow_lock_wait_time - Tiempo total de unidad de trabajo en espera de bloqueos
UOW_COMP_STATUS	VARCHAR(14)	uow_comp_status - Estado de finalización de 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"> <li>• APPL_END</li> <li>• UOWABEND</li> <li>• UOWCOMMIT</li> <li>• UOWDEADLOCK</li> <li>• UOWLOCKTIMEOUT</li> <li>• UOWROLLBACK</li> <li>• UOWUNKNOWN</li> </ul>
AGENT_USR_CPU_TIME_S	BIGINT	agent_usr_cpu_time - Tiempo de CPU de usuario empleado por el agente
AGENT_USR_CPU_TIME_MS	BIGINT	agent_usr_cpu_time - Tiempo de CPU de usuario empleado por el agente
AGENT_SYS_CPU_TIME_S	BIGINT	agent_sys_cpu_time - Tiempo de CPU del sistema empleado por el agente
AGENT_SYS_CPU_TIME_MS	BIGINT	agent_sys_cpu_time - Tiempo de CPU del sistema empleado por el agente
APPL_CON_TIME	TIMESTAMP	appl_con_time - Indicación de hora de inicio de solicitud de conexión
CONN_COMPLETE_TIME	TIMESTAMP	conn_complete_time - Indicación de hora de finalización de solicitud de conexión

Tabla 147. 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
LAST_RESET	TIMESTAMP	last_reset - Indicación de hora de último restablecimiento
UOW_START_TIME	TIMESTAMP	uow_start_time - Indicación de hora de inicio de unidad de trabajo
UOW_STOP_TIME	TIMESTAMP	uow_stop_time - Indicación de hora de finalización de unidad de trabajo
PREV_UOW_STOP_TIME	TIMESTAMP	prev_uow_stop_time - Indicación de 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 sentencias
ELAPSED_EXEC_TIME_MS	BIGINT	elapsed_exec_time - Tiempo transcurrido de ejecución de sentencias
INBOUND_COMM_ADDRESS	VARCHAR(32)	inbound_comm_address - Dirección de comunicación 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

Tabla 147. 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
SHR_WORKSPACE_SIZE_TOP	BIGINT	shr_workspace_size_top - Tamaño máximo de espacio de trabajo compartido
DBPARTITIONNUM	SMALLINT	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 118 en la página 371 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL en la vista administrativa SNAPBP y privilegio EXECUTE en la función de tabla SNAP\_GET\_BP\_V95.

### 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)

## Función de tabla SNAP\_GET\_BP\_V95

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 118 en la página 371 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. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, implícitamente se establece -1 para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, si sólo se proporciona *nombrebd*, se devuelven los datos de todas las particiones de base de datos.

Si tanto *nombredb* como *núm\_partición\_bd* se establecen en NULL, los datos se intentarán leer desde el 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

- Autorización SYSMON
- Privilegio EXECUTE en la función de tabla SNAP\_GET\_BP\_V95.

## 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 registros(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```
... TOTAL_PHYSICAL_READS DBPARTITIONNUM
... -----
...                0                0
...                0                0
...                0                0
```

## Información devuelta

Tabla 148. 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.
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



Tabla 148. 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_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_XDA_L_READS	BIGINT	pool_xda_l_reads - Lecturas lógicas de datos XDA 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_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
POOL_ASYNC_DATA_READS	BIGINT	pool_async_data_reads - Lecturas de datos asíncronos de agrupación de almacenamientos intermedios
POOL_ASYNC_DATA_WRITES	BIGINT	pool_async_data_writes - Grabaciones de datos asíncronos de agrupación de almacenamientos intermedios
POOL_ASYNC_INDEX_READS	BIGINT	pool_async_index_reads - de índice asíncrono de agrupación de almacenamientos intermedios
POOL_ASYNC_INDEX_WRITES	BIGINT	pool_async_index_writes - Grabaciones de índice asíncrono de agrupación de almacenamientos intermedios

Tabla 148. 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_XDA_READS	BIGINT	pool_async_xda_reads - Lecturas de datos XDA asíncronos de agrupación de almacenamientos intermedios
POOL_ASYNC_XDA_WRITES	BIGINT	pool_async_xda_writes - Grabaciones de datos XDA asíncronos 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 - Solicitudes de lectura asíncrona de agrupación de almacenamientos intermedios
POOL_ASYNC_INDEX_READ_REQS	BIGINT	pool_async_index_read_reqs - Solicitudes de lectura de índice asíncrona de agrupación de almacenamientos intermedios
POOL_ASYNC_XDA_READ_REQS	BIGINT	pool_async_xda_read_reqs - Solicitudes de lectura XDA asíncrona de agrupación de almacenamientos intermedios
DIRECT_READS	BIGINT	direct_reads - Lecturas directas de base de datos
DIRECT_WRITES	BIGINT	direct_writes - Grabaciones directas en base de datos
DIRECT_READ_REQS	BIGINT	direct_read_reqs - Solicitudes de lectura directa
DIRECT_WRITE_REQS	BIGINT	direct_write_reqs - Solicitudes 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 base de datos cerrados
POOL_TEMP_DATA_L_READS	BIGINT	pool_temp_data_l_reads - Lecturas lógicas de datos temporales de agrupación de almacenamientos intermedios

Tabla 148. 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_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 índice temporal de agrupación de almacenamientos intermedios
POOL_TEMP_INDEX_P_READS	BIGINT	pool_temp_index_p_reads - Grabaciones físicas de índice temporal de agrupación de almacenamientos intermedios
POOL_TEMP_XDA_L_READS	BIGINT	pool_temp_xda_l_reads - Lecturas lógicas de datos XSA temporales de agrupación de almacenamientos intermedios
POOL_TEMP_XDA_P_READS	BIGINT	pool_temp_xda_p_reads - Elemento de supervisor de lecturas físicas de datos XDA temporales de agrupación de almacenamientos intermedios
POOL_NO_VICTIM_BUFFER	BIGINT	pool_no_victim_buffer - Almacenamientos intermedios no 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 por E/S vectorial
VECTORED_IOS	BIGINT	vectored_ios - Número de solicitudes de E/S vectorial
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 119 en la página 377 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL en la vista administrativa SNAPBP\_PART y el privilegio EXECUTE en la función de tabla SNAP\_GET\_BP\_PART.

### 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 119 en la página 377 para ver una lista completa de la información que se puede devolver.

## Sintaxis

```
►► SNAP_GET_BP_PART ( ( nombre-bd [ , nm-particin-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 vlido 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 vaca para tomar la instantnea desde la base de datos conectada actualmente. Especifique un valor NULL para tomar la instantnea desde todas las bases de datos que se encuentran en la misma instancia que la base de datos conectada actualmente.

### *nm-particin-bd*

Argumento de entrada opcional de tipo INTEGER que especifica un nmero de particin de base de datos vlido. Especifique -1 para la particin de base de datos actual o -2 para un agregado de todas las particiones de base de datos. Si *nombre-bd* no se establece en NULL y *nm-particin-bd* se establece en NULL, implcitamente se establece -1 para *nm-particin-bd*. Si no se utiliza esta opcin de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *nm-particin-bd* se establecen en NULL, los datos se intentarn leer desde el archivo que ha creado el procedimiento SNAP\_WRITE\_FILE. Tenga en cuenta que este archivo podra haberse creado en cualquier momento, lo que significa que es posible que los datos no estn actualizados. Si no existe un archivo con el correspondiente tipo de peticin de API de instantnea, la funcin de tabla SNAP\_GET\_BP\_PART toma una instantnea del nmero de particin de base de datos y la base de datos conectada actualmente.

## Autorizacin

- Autorizacin SYSMON
- Privilegio EXECUTE en la funcin de tabla SNAP\_GET\_BP\_PART.

## Ejemplo

Recuperar datos de todas las agrupaciones de almacenamientos intermedios de todas las bases de datos activas cuando estn 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 149. 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 la 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 tabla correlacionados con la agrupación de almacenamientos intermedios

Tabla 149. Información devuelta por la vista administrativa SNAPBP\_PART y la función de tabla SNAP\_GET\_BP\_PART (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 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 120 en la página 381 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL para la vista administrativa SNAPCONTAINER y privilegio EXECUTE para la función de tabla SNAP\_GET\_CONTAINER\_V91.

### 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

```

## Función de tabla SNAP\_GET\_CONTAINER\_V91

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 120 en la página 381 para ver una lista completa de la información que se puede devolver.

### Sintaxis

```

▶▶ SNAP_GET_CONTAINER_V91 ( ( nombre-bd [ , nm-particin-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 NULL o una serie vacía para tomar la instantánea desde la base de datos conectada actualmente.

*núm-partición-bd*

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. Si *nombrebd* no se establece en NULL y *núm-partición-bd* se establece en NULL, implícitamente se establece -1 para *núm-partición-bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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**

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_CONTAINER\_V91.

**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:\DGTsWhatsNewContainer	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
... 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 150. 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 del espacio de tablas
CONTAINER_NAME	VARCHAR(256)	container_name - Nombre del 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"> <li>• DISK_EXTENT_TAG</li> <li>• DISK_PAGE_TAG</li> <li>• FILE_EXTENT_TAG</li> <li>• FILE_PAGE_TAG</li> <li>• PATH</li> </ul>
TOTAL_PAGES	BIGINT	container_total_pages - Páginas totales en contenedor
USABLE_PAGES	BIGINT	container_usable_pages - Páginas útiles en contenedor
ACCESSIBLE	SMALLINT	container_accessible - Accesibilidad del contenedor
STRIPE_SET	BIGINT	container_stripe_set - Conjunto de bandas
DBPARTITIONNUM	SMALLINT	La 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

Tabla 150. 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_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 382 y la “Función de tabla SNAP\_GET\_DB\_V95” en la página 383 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, SNAPDETAILLOG, 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 121 en la página 385 encontrará una lista completa de la información que se devuelve.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL para la vista administrativa SNAPDB y privilegio EXECUTE para la función de tabla SNAP\_GET\_DB\_V95.

### 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 registros(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:

```

db2 "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_FABRICATION_TIME SYNC_RUNSTATS_TIME
-----
              10              100

```

1 registro(s) seleccionado(s).

## **Función de tabla SNAP\_GET\_DB\_V95**

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 121 en la página 385 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. Si *nombrebd* no se establece en NULL y *númparticiónbd* se establece en NULL, implícitamente se establece -1 para *númparticiónbd*. Si no se utiliza esta opción de entrada, es decir, si sólo se proporciona *nombrebd*, se devuelven los datos de todas las particiones de base de datos.

Si tanto *nombrebd* como *númparticiónbd* se establecen en NULL, los datos se intentarán leer desde el 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_DB\_V95.

## 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:

```
db2 "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_FABRICATION_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 151. 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

Tabla 151. 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
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"> <li>• ACTIVE</li> <li>• QUIESCE_PEND</li> <li>• QUIESCED</li> <li>• ROLLFWD</li> </ul>
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 151. 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"> <li>• AIX</li> <li>• AIX64</li> <li>• AS400_DRDA</li> <li>• DOS</li> <li>• DYNIX</li> <li>• HP</li> <li>• HP64</li> <li>• HPIA</li> <li>• HPIA64</li> <li>• LINUX</li> <li>• LINUX390</li> <li>• LINUXIA64</li> <li>• LINUXPPC</li> <li>• LINUXPPC64</li> <li>• LINUXX8664</li> <li>• LINUXZ64</li> <li>• MAC</li> <li>• MVS_DRDA</li> <li>• NT</li> <li>• NT64</li> <li>• OS2</li> <li>• OS390</li> <li>• SCO</li> <li>• SGI</li> <li>• SNI</li> <li>• SUN</li> <li>• SUN64</li> <li>• UNKNOWN</li> <li>• UNKNOWN_DRDA</li> <li>• VM_DRDA</li> <li>• VSE_DRDA</li> <li>• WINDOWS</li> </ul>
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"> <li>• LOCAL</li> <li>• REMOTE</li> </ul>
DB_CONN_TIME	TIMESTAMP	<p>db_conn_time - Indicación de la hora de activación de la base de datos</p>



Tabla 151. 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 hora de último restablecimiento
LAST_BACKUP	TIMESTAMP	last_backup - Indicación de hora de última copia de seguridad
CONNECTIONS_TOP	BIGINT	connections_top - Número máximo de conexiones simultáneas
TOTAL_CONS	BIGINT	total_cons - Conexiones desde 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 en ejecución en la base de datos simultáneamente
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 de espera de bloqueos
LOCK_LIST_IN_USE	BIGINT	lock_list_in_use - Memoria total de lista de bloqueos utilizada
DEADLOCKS	BIGINT	deadlocks - Puntos muertos detectados
LOCK_ESCALS	BIGINT	lock_escals - Número de reajustes de bloqueo
X_LOCK_ESCALS	BIGINT	x_lock_escals - Reajustes de bloqueo exclusivos
LOCKS_WAITING	BIGINT	locks_waiting - Agentes actuales en espera de bloqueos
LOCK_TIMEOUTS	BIGINT	lock_timeouts - Número de tiempos de espera excedidos de bloqueo
NUM_INDOUBT_TRANS	BIGINT	num_indoubt_trans - Número de transacciones dudosas
SORT_HEAP_ALLOCATED	BIGINT	sort_heap_allocated - Pila total de clasificación asignada
SORT_SHRHEAP_ALLOCATED	BIGINT	sort_shrheap_allocated - Pila de compartición de clasificación actualmente asignada
SORT_SHRHEAP_TOP	BIGINT	sort_shrheap_top - Marca de límite superior de pila de compartición de clasificación
POST_SHRTHRESHOLD_SORTS	BIGINT	post_shrthreshold_sorts - Clasificaciones de umbral compartido posteriores
TOTAL_SORTS	BIGINT	total_sorts - Número total de clasificaciones
TOTAL_SORT_TIME	BIGINT	total_sort_time - Tiempo total de clasificación

Tabla 151. 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
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
POOL_ASYNC_DATA_READS	BIGINT	pool_async_data_reads - Lecturas de datos asíncronos 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 asíncronos 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 temporal de agrupación de almacenamientos intermedios
POOL_TEMP_INDEX_P_READS	BIGINT	pool_temp_index_p_reads - Grabaciones físicas de índice temporal de agrupación de almacenamientos intermedios
POOL_ASYNC_INDEX_READS	BIGINT	pool_async_index_reads - de índice asíncrono 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 asíncrono de agrupación de almacenamientos intermedios
POOL_XDA_P_READS	BIGINT	pool_xda_p_reads - Lecturas físicas de datos XDA de agrupación de almacenamientos intermedios

Tabla 151. 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_L_READS	BIGINT	pool_xda_l_reads - Lecturas lógicas de datos XDA de agrupación de almacenamientos intermedios
POOL_XDA_WRITES	BIGINT	pool_xda_writes - Grabaciones de datos XDA de agrupación de almacenamientos intermedios
POOL_ASYNC_XDA_READS	BIGINT	pool_async_xda_reads - Lecturas de datos XDA asíncronos de agrupación de almacenamientos intermedios
POOL_ASYNC_XDA_WRITES	BIGINT	pool_async_xda_writes - Grabaciones de datos XDA asíncronos de agrupación de almacenamientos intermedios
POOL_TEMP_XDA_P_READS	BIGINT	pool_temp_xda_p_reads - Elemento de supervisor de lecturas físicas de datos XDA temporales de agrupación de almacenamientos intermedios
POOL_TEMP_XDA_L_READS	BIGINT	pool_temp_xda_l_reads - Lecturas lógicas de datos XSA temporales 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
POOL_ASYNC_DATA_READ_REQS	BIGINT	pool_async_data_read_reqs - Solicitudes de lectura asíncrona de agrupación de almacenamientos intermedios
POOL_ASYNC_INDEX_READ_REQS	BIGINT	pool_async_index_read_reqs - Solicitudes de lectura de índice asíncrona de agrupación de almacenamientos intermedios
POOL_ASYNC_XDA_READ_REQS	BIGINT	pool_async_xda_read_reqs - Solicitudes de lectura XDA asíncrona de agrupación de almacenamientos intermedios
POOL_NO_VICTIM_BUFFER	BIGINT	pool_no_victim_buffer - Almacenamientos intermedios no víctimas de la agrupación de almacenamientos intermedios
POOL_LSN_GAP_CLNS	BIGINT	pool_lsn_gap_clns - Limpiadores desencadenados del espacio de registro de la agrupación de almacenamientos intermedios

Tabla 151. 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_DRTY_PG_STEAL_CLNS	BIGINT	pool_drty_pg_steal_clns - Limpiadores de páginas víctimas desencadenados de la agrupación de almacenamientos intermedios
POOL_DRTY_PG_THRSH_CLNS	BIGINT	pool_drty_pg_thrsh_clns - Limpiadores desencadenados de umbrales de agrupación de almacenamientos intermedios
PREFETCH_WAIT_TIME	BIGINT	prefetch_wait_time - Tiempo de espera de captación previa
UNREAD_PREFETCH_PAGES	BIGINT	unread_prefetch_pages - Páginas de captación previa no leídas
DIRECT_READS	BIGINT	direct_reads - Lecturas directas de base de datos
DIRECT_WRITES	BIGINT	direct_writes - Grabaciones directas en base de datos
DIRECT_READ_REQS	BIGINT	direct_read_reqs - Solicitudes de lectura directa
DIRECT_WRITE_REQS	BIGINT	direct_write_reqs - Solicitudes 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 sentencias
ELAPSED_EXEC_TIME_MS	BIGINT	elapsed_exec_time - Tiempo transcurrido de ejecución de sentencias
COMMIT_SQL_STMTS	BIGINT	commit_sql_stmts - Confirmaciones de sentencias intentadas
ROLLBACK_SQL_STMTS	BIGINT	rollback_sql_stmts - Retrotracciones de sentencias 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 sentencias fallidas
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 Data Definition Language (DDL) de SQL
INT_AUTO_REBINDS	BIGINT	int_auto_rebinds - Revinculaciones automáticas internas

Tabla 151. 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
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 debidas 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 - Registro total disponible
TOTAL_LOG_USED	BIGINT	total_log_used - Espacio total de registro utilizado
SEC_LOG_USED_TOP	BIGINT	sec_log_used_top - Espacio máximo de registro secundario utilizado
TOT_LOG_USED_TOP	BIGINT	tot_log_used_top - Espacio máximo total de registro utilizado
SEC_LOGS_ALLOCATED	BIGINT	sec_logs_allocated - Registros secundarios asignados actualmente
LOG_READS	BIGINT	log_reads - Número de páginas de registro leídas
LOG_READ_TIME_S	BIGINT	log_read_time - Tiempo de lectura de registro
LOG_READ_TIME_NS	BIGINT	log_read_time - Tiempo de lectura de registro
LOG_WRITES	BIGINT	log_writes - Número de páginas de registro grabadas
LOG_WRITE_TIME_S	BIGINT	log_write_time - Tiempo de grabación de registro
LOG_WRITE_TIME_NS	BIGINT	log_write_time - Tiempo de grabación de registro
NUM_LOG_WRITE_IO	BIGINT	num_log_write_io - Número de grabaciones de registro
NUM_LOG_READ_IO	BIGINT	num_log_read_io - Número de lecturas de registro
NUM_LOG_PART_PAGE_IO	BIGINT	num_log_part_page_io - Número de grabaciones parciales de páginas de registro
NUM_LOG_BUFFER_FULL	BIGINT	num_log_buffer_full - Número de almacenamientos intermedios de registro llenos

Tabla 151. 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_DATA_FOUND_IN_BUFFER	BIGINT	num_log_data_found_in_buffer - Número de datos de registro encontrados en almacenamiento intermedio
APPL_ID_OLDEST_XACT	BIGINT	appl_id_oldest_xact - Aplicación con transacción más antigua
LOG_TO_REDO_FOR_RECOVERY	BIGINT	log_to_redo_for_recovery - Cantidad de registro que hay que rehacer para recuperación
LOG_HELD_BY_DIRTY_PAGES	BIGINT	log_held_by_dirty_pages - Cantidad de espacio de registro 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
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

Tabla 151. 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
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 - Número total de uniones de generación aleatoria
TOTAL_HASH_LOOPS	BIGINT	total_hash_loops - Número total de bucles de generación aleatoria
HASH_JOIN_OVERFLOWS	BIGINT	hash_join_overflows - Desbordamientos de unión de generación aleatoria
HASH_JOIN_SMALL_OVERFLOWS	BIGINT	hash_join_small_overflows - Desbordamientos pequeños de unión de generación aleatoria
POST_SHRTHRESHOLD_HASH_JOINS	BIGINT	post_shrthreshold_hash_joins - Uniones de generación aleatoria de umbral posteriores
ACTIVE_HASH_JOINS	BIGINT	active_hash_joins - Uniones activas de generación aleatoria
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 menor cantidad de espacio de registro 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
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.

Tabla 151. 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_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.

### SNAPDB\_MEMORY\_POOL, vista administrativa

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, SNAPDETAILOG, 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 122 en la página 398 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL en la vista administrativa SNAPDB\_MEMORY\_POOL y privilegio EXECUTE en la función de tabla SNAP\_GET\_DB\_MEMORY\_POOL.

### 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).

## Función de tabla SNAP\_GET\_DB\_MEMORY\_POOL

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 122 en la página 398 para ver una lista completa de la información que se puede devolver.

## Sintaxis

```

▶▶ SNAP_GET_DB_MEMORY_POOL ( ( nombre-bd [ , n m-partici n-bd ] ) )

```

El esquema es SYSPROC.

## Par metros de la funci n de tabla

### nombre-bd

Argumento de entrada de tipo VARCHAR(128) que especifica un nombre de base de datos v lido 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 vac a para tomar la instant nea desde la base de datos conectada actualmente. Especifique un valor NULL para tomar la instant nea desde todas las bases de datos que se encuentran en la misma instancia que la base de datos conectada actualmente.

### n m-partici n-bd

Argumento de entrada opcional de tipo INTEGER que especifica un n mero de partici n de base de datos v lido. Especifique -1 para la partici n de base de datos actual o -2 para un agregado de todas las particiones de base de datos. Si nombrebd no se establece en NULL y n m\_partici n\_bd se establece en

NULL, implícitamente se establece -1 para *núm\_partición\_bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre\_bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre\_bd* y *núm-partición\_bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE en la función de tabla SNAP\_GET\_DB\_MEMORY\_POOL.

## 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 152. 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 la base de datos
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"> <li>• APP_GROUP</li> <li>• APPL_CONTROL</li> <li>• APPLICATION</li> <li>• BP</li> <li>• CAT_CACHE</li> <li>• DATABASE</li> <li>• DFM</li> <li>• FCMBP</li> <li>• IMPORT_POOL</li> <li>• LOCK_MGR</li> <li>• MONITOR</li> <li>• OTHER</li> <li>• PACKAGE_CACHE</li> <li>• QUERY</li> <li>• SHARED_SORT</li> <li>• SORT</li> <li>• SENTENCIA</li> <li>• STATISTICS</li> <li>• UTILITY</li> </ul>
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 - Nivel de agrupación de memoria
POOL_CONFIG_SIZE	BIGINT	pool_config_size - Tamaño configurado de la agrupación de memoria
DBPARTITIONNUM	SMALLINT	La 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.

### SNAPDBM, vista administrativa

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 123 en la página 400 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL en la vista administrativa SNAPDBM y el privilegio EXECUTE en la función de tabla SNAP\_GET\_DBM\_V95.

### 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 registros(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
```

### Función de tabla SNAP\_GET\_DBM\_V95

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 123 en la página 400 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. Si no se utiliza esta opción de entrada, todas las particiones de la base de datos devolverán datos.

Si *núm\_partición\_bd* se establece en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_DBM\_V95.

## 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 153. 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.

Tabla 153. 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
SORT_HEAP_ALLOCATED	BIGINT	sort_heap_allocated - Pila total de clasificación asignada
POST_THRESHOLD_SORTS	BIGINT	post_threshold_sorts - Clasificaciones de umbral posteriores
PIPED_SORTS_REQUESTED	BIGINT	piped_sorts_requested - Clasificaciones conducidas solicitadas
PIPED_SORTS_ACCEPTED	BIGINT	piped_sorts_accepted - Clasificaciones conducidas 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 supervisor del 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 supervisor del 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 a la espera de un símbolo
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"> <li>• ACTIVE</li> <li>• QUIESCE_PEND</li> <li>• QUIESCED</li> </ul>
AGENTS_REGISTERED_TOP	BIGINT	agents_registered_top - Número máximo de agentes registrados
AGENTS_WAITING_TOP	BIGINT	agents_waiting_top - Número máximo de agentes a la 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 desde agrupación
AGENTS_CREATED_EMPTY_POOL	BIGINT	agents_created_empty_pool - Agentes creados debido a una agrupación de agentes vacía
COORD_AGENTS_TOP	BIGINT	coord_agents_top - Número máximo de agentes de coordinación

Tabla 153. 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
MAX_AGENT_OVERFLOW	BIGINT	max_agent_overflows - Número máximo de desbordamientos de agente
AGENTS_STOLEN	BIGINT	agents_stolen - Agentes sustraídos
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 a la espera de que el sistema principal responda
GW_CONS_WAIT_CLIENT	BIGINT	gw_cons_wait_client - Número de conexiones a la espera de que el cliente envíe una solicitud
POST_THRESHOLD_HASH_JOINS	BIGINT	post_threshold_hash_joins - Umbral de uniones de generación aleatoria
NUM_GW_CONN_SWITCHES	BIGINT	num_gw_conn_switches - Conmutadores de conexión
DB2START_TIME	TIMESTAMP	db2start_time - Indicación de hora de inicio del gestor de bases de datos
LAST_RESET	TIMESTAMP	last_reset - Indicación de hora de último restablecimiento
NUM_NODES_IN_DB2_INSTANCE	INTEGER	num_nodes_in_db2_instance - Número de nodos de 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.

Tabla 153. 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
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>La 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.

### SNAPDBM\_MEMORY\_POOL, vista administrativa

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 124 en la página 405 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON



- Privilegio SELECT o CONTROL en la vista administrativa SNAPDBM\_MEMORY\_POOL y privilegio EXECUTE en la función de tabla SNAP\_GET\_DBM\_MEMORY\_POOL.

## 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
...	

## Función de tabla SNAP\_GET\_DBM\_MEMORY\_POOL

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 124 en la página 405 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 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. Si no se utiliza esta opción de entrada, todas las particiones de la base de datos devolverán datos.

Si *núm-partición-bd* se establece en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE en la función de tabla SNAP\_GET\_DBM\_MEMORY\_POOL.

## 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 154. 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.

Tabla 154. Información que devuelve la vista administrativa SNAPDBM\_MEMORY\_POOL y la función de tabla SNAP\_GET\_DBM\_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"> <li>• APP_GROUP</li> <li>• APPL_CONTROL</li> <li>• APPLICATION</li> <li>• BP</li> <li>• CAT_CACHE</li> <li>• DATABASE</li> <li>• DFM</li> <li>• FCMBP</li> <li>• IMPORT_POOL</li> <li>• LOCK_MGR</li> <li>• MONITOR</li> <li>• OTHER</li> <li>• PACKAGE_CACHE</li> <li>• QUERY</li> <li>• SHARED_SORT</li> <li>• SORT</li> <li>• SENTENCIA</li> <li>• STATISTICS</li> <li>• UTILITY</li> </ul>
POOL_CUR_SIZE	BIGINT	pool_cur_size - Tamaño actual de la agrupación de memoria
POOL_WATERMARK	BIGINT	pool_watermark - Nivel de agrupación de memoria
POOL_CONFIG_SIZE	BIGINT	pool_config_size - Tamaño configurado de la agrupación de memoria
DBPARTITIONNUM	SMALLINT	La 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 desde el 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, SNAPHADR 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 125 en la página 409 encontrará una lista completa de la información que se devuelve.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL para la vista administrativa SNAPDETAILLOG y privilegio EXECUTE para la función de tabla SNAP\_GET\_DETAILLOG\_V91.

### 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 registros(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 125 en la página 409 encontrará una lista completa de la información que se devuelve.

## Sintaxis

```
▶▶ SNAP_GET_DETAILLOG_V91 (—nombre-bd [—núm-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.

### *núm-partición-bd*

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. Si *nombrebd* no se establece en NULL y *núm-partición-bd* se establece en NULL, implícitamente se establece -1 para *núm-partición-bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_DETAILLOG\_V91.

## 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 155. 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 la base de datos
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	La 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 “vista administrativa SNAPDYN\_SQL” en la página 409 y la “Función de tabla SNAP\_GET\_DYN\_SQL\_V95” en la página 411 devuelven información de instantánea desde el grupo de datos lógicos dynsql.

#### vista administrativa SNAPDYN\_SQL

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 126 en la página 413 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL para la vista administrativa SNAPDYN\_SQL y privilegio EXECUTE para la función de tabla SNAP\_GET\_SQL\_V95.

### 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 126 en la página 413 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 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. Si *nombrebd* no se establece en NULL y *núm\_partición\_bd* se establece en NULL, implícitamente se establece -1 para *núm\_partición\_bd*. Si no se utiliza esta opción de entrada, es decir, si sólo se proporciona *nombrebd*, se devuelven los datos de todas las particiones de base de datos.

Si tanto *nombrebd* como *núm\_partición\_bd* se establecen en NULL, los datos se intentarán leer desde el 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**

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_DYN\_SQL\_V95.



## 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 156. 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 sentencias
NUM_COMPILATIONS	BIGINT	num_compilations - Compilaciones de sentencias
PREP_TIME_WORST	BIGINT	prep_time_worst - Peor tiempo de preparación de sentencias
PREP_TIME_BEST	BIGINT	prep_time_best - Mejor tiempo de preparación de sentencias
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 total de clasificación
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
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 temporal de agrupación de almacenamientos intermedios

Tabla 156. 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_P_READS	BIGINT	pool_temp_index_p_reads - Grabaciones físicas de índice temporal 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_XDA_P_READS	BIGINT	pool_xda_p_reads - Lecturas físicas de datos XDA de agrupación de almacenamientos intermedios
POOL_TEMP_XDA_L_READS	BIGINT	pool_temp_xda_l_reads - Lecturas lógicas de datos XSA temporales de agrupación de almacenamientos intermedios
POOL_TEMP_XDA_P_READS	BIGINT	pool_temp_xda_p_reads - Elemento de supervisor de lecturas físicas de datos XDA temporales de agrupación de almacenamientos intermedios
TOTAL_EXEC_TIME	BIGINT	total_exec_time - Tiempo transcurrido de ejecución de sentencias
TOTAL_EXEC_TIME_MS	BIGINT	total_exec_time - Tiempo transcurrido de ejecución de sentencias
TOTAL_USR_CPU_TIME	BIGINT	total_usr_cpu_time - CPU total de usuario para una sentencia
TOTAL_USR_CPU_TIME_MS	BIGINT	total_usr_cpu_time - CPU total de usuario para una sentencia
TOTAL_SYS_CPU_TIME	BIGINT	total_sys_cpu_time - CPU total del sistema para una sentencia
TOTAL_SYS_CPU_TIME_MS	BIGINT	total_sys_cpu_time - CPU total del sistema para una sentencia
STMT_TEXT	CLOB(2 M)	stmt_text - Texto de sentencia 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 127 en la página 416 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL para la vista administrativa SNAPFCM y privilegio EXECUTE para la función de tabla SNAP\_GET\_FCM.

### 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

### Función de tabla SNAP\_GET\_FCM

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 127 en la página 416 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 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. Si no se utiliza esta opción de entrada, todas las particiones de base de datos devolverán los datos.

Si *núm-partición-bd* se establece en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_FCM.

## 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 157. 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

Tabla 157. Información que devuelve la vista administrativa SNAPFCM y la función de tabla SNAP\_GET\_FCM (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
CH_FREE	BIGINT	ch_free - Canales libres actualmente
CH_FREE_BOTTOM	BIGINT	ch_free_bottom - Número mínimo de canales libres
DBPARTITIONNUM	SMALLINT	La 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 128 en la página 419 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL en la vista administrativa SNAPFCM\_PART y privilegio EXECUTE en la función de tabla SNAP\_GET\_FCM\_PART.

### 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).

## Función de tabla SNAP\_GET\_FCM\_PART

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 128 en la página 419 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 de partición de base de datos válido. Especifique -1 para la partición actual o -2 para un agregado de todas las particiones. Si no se utiliza esta opción de entrada, todas las particiones devolverán los datos.

Si *núm\_partición\_bd* se establece en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE en la función de tabla SNAP\_GET\_FCM\_PART.

### 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 158. 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 basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> <li>• INACTIVE</li> <li>• ACTIVE</li> <li>• CONGESTED</li> </ul>
TOTAL_BUFFERS_SENT	BIGINT	total_buffers_sent - Total de almacenamientos intermedios de FCM enviados
TOTAL_BUFFERS_RCVD	BIGINT	total_buffers_rcvd - Total de almacenamientos intermedios de FCM recibidos
DBPARTITIONNUM	SMALLINT	La 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 129 en la página 421 para ver una lista completa de la información que se puede devolver.

## Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL en la vista administrativa SNAPHADR y el privilegio EXECUTE en la función de tabla SNAP\_GET\_HADR.

## 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).

## Función de tabla SNAP\_GET\_HADR

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 129 en la página 421 para ver una lista completa de la información que se puede devolver.

## Sintaxis

```
▶▶ SNAP_GET_HADR ( ( nombre-bd [ , n m-partici n-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.

*núm-partición-bd*

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. Si *nombrebd* no se establece en NULL y *núm\_partición\_bd* se establece en NULL, implícitamente se establece -1 para *núm\_partición\_bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE en la función de tabla SNAP\_GET\_HADR.

### 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 159. 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 la base de datos

Tabla 159. 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_ROLE	VARCHAR(10)	<p>hadr_role - Función HADR. 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"> <li>• PRIMARY</li> <li>• STANDARD</li> <li>• STANDBY</li> </ul>
HADR_STATE	VARCHAR(14)	<p>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:</p> <ul style="list-style-type: none"> <li>• DISCONNECTED</li> <li>• LOCAL_CATCHUP</li> <li>• PEER</li> <li>• REM_CATCH_PEN</li> <li>• REM_CATCHUP</li> </ul>
HADR_SYNCMODE	VARCHAR(10)	<p>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:</p> <ul style="list-style-type: none"> <li>• ASYNC</li> <li>• NEARSYNC</li> <li>• SYNC</li> </ul>
HADR_CONNECT_STATUS	VARCHAR(12)	<p>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:</p> <ul style="list-style-type: none"> <li>• CONGESTED</li> <li>• CONNECTED</li> <li>• DISCONNECTED</li> </ul>
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

Tabla 159. 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_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	La 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 130 en la página 425 para ver una lista completa de la información que se puede devolver.

## Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL en la vista administrativa SNAPLOCK y privilegio EXECUTE en la función de tabla SNAP\_GET\_LOCK.

## 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).

## Función de tabla SNAP\_GET\_LOCK

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 130 en la página 425 para ver una lista completa de la información que se puede devolver.

## Sintaxis

```
▶▶ SNAP_GET_LOCK ( ( nombre-bd [ , nm-particin-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 vlido 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 vaca para tomar la instantnea desde la base de datos conectada actualmente.

*nm-particin-bd*

Argumento de entrada opcional de tipo INTEGER que especifica un nmero de particin de base de datos vlido. Especifique -1 para la particin de base

de datos actual o -2 para un agregado de todas las particiones de base de datos. Si *nombrebd* no se establece en NULL y *núm\_partición\_bd* se establece en NULL, implícitamente se establece -1 para *núm\_partición\_bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_LOCK.

## 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)) 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 160. 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 160. 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 basado en las sentencias define de sqlmon.h y es uno de los siguientes:</p> <ul style="list-style-type: none"> <li>• AUTORESIZE_LOCK</li> <li>• AUTOSTORAGE_LOCK</li> <li>• BLOCK_LOCK</li> <li>• EOT_LOCK</li> <li>• INPLACE_REORG_LOCK</li> <li>• INTERNAL_LOCK</li> <li>• INTERNALB_LOCK</li> <li>• INTERNALC_LOCK</li> <li>• INTERNALJ_LOCK</li> <li>• INTERNALL_LOCK</li> <li>• INTERNALO_LOCK</li> <li>• INTERNALQ_LOCK</li> <li>• INTERNALP_LOCK</li> <li>• INTERNALS_LOCK</li> <li>• INTERNALT_LOCK</li> <li>• INTERNALV_LOCK</li> <li>• KEYVALUE_LOCK</li> <li>• ROW_LOCK</li> <li>• SYSBOOT_LOCK</li> <li>• TABLE_LOCK</li> <li>• TABLE_PART_LOCK</li> <li>• TABLESPACE_LOCK</li> <li>• XML_PATH_LOCK</li> </ul>

Tabla 160. 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_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"> <li>• IN</li> <li>• IS</li> <li>• IX</li> <li>• NON (si no existe bloqueo)</li> <li>• NS</li> <li>• NW</li> <li>• NX</li> <li>• S</li> <li>• SIX</li> <li>• U</li> <li>• W</li> <li>• X</li> <li>• Z</li> </ul>
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"> <li>• CONV</li> <li>• GRNT</li> </ul>
LOCK_ESCALATION	SMALLINT	lock_escalation - Escala de bloqueo
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



Tabla 160. 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_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"> <li>• ALLOW_NEW</li> <li>• DELETE_IN_BLOCK</li> <li>• ESCALATED</li> <li>• INSERT</li> <li>• NEW_REQUEST</li> <li>• RR</li> <li>• RR_IN_BLOCK</li> <li>• UPDATE_DELETE</li> <li>• WAIT_FOR_AVAIL</li> </ul>
LOCK_COUNT	BIGINT	lock_count - Número de bloqueos
LOCK_CURRENT_MODE	VARCHAR(10)	lock_current_mode - Modalidad de bloqueo original antes de la conversión. 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"> <li>• IN</li> <li>• IS</li> <li>• IX</li> <li>• NON (si no existe bloqueo)</li> <li>• NS</li> <li>• NW</li> <li>• NX</li> <li>• S</li> <li>• SIX</li> <li>• U</li> <li>• W</li> <li>• X</li> <li>• Z</li> </ul>
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.

Tabla 160. 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
DBPARTITIONNUM	SMALLINT	La 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 131 en la página 431 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL en la vista administrativa SNAPLOCKWAIT y privilegio EXECUTE en la función de tabla SNAP\_GET\_LOCKWAIT.

### 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
```

## Función de tabla SNAP\_GET\_LOCKWAIT

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 131 en la página 431 para ver una lista completa de la información que se puede devolver.

### Sintaxis

```
▶▶ SNAP_GET_LOCKWAIT ( nombre-bd [ , núm-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 un valor nulo o una serie vacía para tomar la instantánea desde la base de datos conectada actualmente.

#### *núm-partición-bd*

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. Si *nombrebd* no se establece en NULL y *núm-partición-bd* se establece en NULL, implícitamente se establece -1 para *núm-partición-bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_LOCKWAIT.

## 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 161. 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 161. 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 basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> <li>• IN</li> <li>• IS</li> <li>• IX</li> <li>• NON (si no existe bloqueo)</li> <li>• NS</li> <li>• NW</li> <li>• NX</li> <li>• S</li> <li>• SIX</li> <li>• U</li> <li>• W</li> <li>• X</li> <li>• Z</li> </ul>

Tabla 161. 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_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"> <li>• AUTORESIZE_LOCK</li> <li>• AUTOSTORAGE_LOCK</li> <li>• BLOCK_LOCK</li> <li>• EOT_LOCK</li> <li>• INPLACE_REORG_LOCK</li> <li>• INTERNAL_LOCK</li> <li>• INTERNALB_LOCK</li> <li>• INTERNALC_LOCK</li> <li>• INTERNALJ_LOCK</li> <li>• INTERNALL_LOCK</li> <li>• INTERNALO_LOCK</li> <li>• INTERNALQ_LOCK</li> <li>• INTERNALP_LOCK</li> <li>• INTERNALS_LOCK</li> <li>• INTERNALT_LOCK</li> <li>• INTERNALV_LOCK</li> <li>• KEYVALUE_LOCK</li> <li>• ROW_LOCK</li> <li>• SYSBOOT_LOCK</li> <li>• TABLE_LOCK</li> <li>• TABLE_PART_LOCK</li> <li>• TABLESPACE_LOCK</li> <li>• XML_PATH_LOCK</li> </ul>
AGENT_ID_HOLDING_LK	BIGINT	agent_id_holding_lock - Bloqueo de retención de ID de agente
LOCK_WAIT_START_TIME	TIMESTAMP	lock_wait_start_time - Indicación de la hora de inicio de espera de bloqueo

Tabla 161. 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_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"> <li>• IN</li> <li>• IS</li> <li>• IX</li> <li>• NON (si no existe bloqueo)</li> <li>• NS</li> <li>• NW</li> <li>• NX</li> <li>• S</li> <li>• SIX</li> <li>• U</li> <li>• W</li> <li>• X</li> <li>• Z</li> </ul>
LOCK_ESCALATION	SMALLINT	lock_escalation - Escala de bloqueo
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
APPL_ID_HOLDING_LK	VARCHAR(128)	appl_id_holding_lk - Bloqueo de retención de ID de aplicación
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"> <li>• ALLOW_NEW</li> <li>• DELETE_IN_BLOCK</li> <li>• ESCALATED</li> <li>• INSERT</li> <li>• NEW_REQUEST</li> <li>• RR</li> <li>• RR_IN_BLOCK</li> <li>• UPDATE_DELETE</li> <li>• WAIT_FOR_AVAIL</li> </ul>

Tabla 161. 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_CURRENT_MODE	VARCHAR(10)	lock_current_mode - Modalidad de bloqueo original antes de la conversión. 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"> <li>• IN</li> <li>• IS</li> <li>• IX</li> <li>• NON (si no existe bloqueo)</li> <li>• NS</li> <li>• NW</li> <li>• NX</li> <li>• S</li> <li>• SIX</li> <li>• U</li> <li>• W</li> <li>• X</li> <li>• Z</li> </ul>
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	La 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 132 en la página 438 para ver una lista completa de la información que se puede devolver.

## Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL en la vista administrativa SNAPSTMT y el privilegio EXECUTE en la función de tabla SNAP\_GET\_STMT.

## 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).

## Función de tabla SNAP\_GET\_STMT

La función de tabla SNAP\_GET\_STMT devuelve la misma información que la vista administrativa SNAPSTMT, 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 132 en la página 438 para ver una lista completa de la información que se puede devolver.

## Sintaxis

```
▶▶ SNAP_GET_STMT ( ( nombre-bd [ , nm-particin-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.

### *núm-partición-bd*

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. Si *nombrebd* no se establece en NULL y *núm\_partición\_bd* se establece en NULL, implícitamente se establece -1 para *núm\_partición\_bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE en la función de tabla SNAP\_GET\_STMT.

## 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 162. 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.
DB_NAME	VARCHAR(128)	db_name - Nombre de la 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 que trabaja 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"> <li>• DYNAMIC</li> <li>• NON_STMT</li> <li>• STATIC</li> <li>• STMT_TYPE_UNKNOWN</li> </ul>

Tabla 162. 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"> <li>• CALL</li> <li>• CLOSE</li> <li>• COMPILE</li> <li>• DESCRIBE</li> <li>• EXECUTE</li> <li>• EXECUTE_IMMEDIATE</li> <li>• FETCH</li> <li>• FREE_LOCATOR</li> <li>• GETAA</li> <li>• GETNEXTCHUNK</li> <li>• GETTA</li> <li>• NONE</li> <li>• OPEN</li> <li>• PREP_COMMIT</li> <li>• PREP_EXEC</li> <li>• PREP_OPEN</li> <li>• PREPARE</li> <li>• REBIND</li> <li>• REDIST</li> <li>• REORG</li> <li>• RUNSTATS</li> <li>• SELECT</li> <li>• SET</li> <li>• STATIC_COMMIT</li> <li>• STATIC_ROLLBACK</li> </ul>
SECTION_NUMBER	BIGINT	section_number - Número de sección
QUERY_COST_ESTIMATE	BIGINT	query_cost_estimate - Estimación del coste de la consulta
QUERY_CARD_ESTIMATE	BIGINT	query_card_estimate - Número estimado de consulta de las 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 total de clasificación
SORT_OVERFLOWS	BIGINT	sort_overflows - Clasificar desbordamientos

Tabla 162. 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 la hora de inicio de operación de sentencia
STMT_STOP	TIMESTAMP	stmt_stop - Indicación de la hora de detención de operación de sentencia
STMT_USR_CPU_TIME_S	BIGINT	stmt_usr_cpu_time - Tiempo de CPU de usuario que la sentencia utiliza
STMT_USR_CPU_TIME_MS	BIGINT	stmt_usr_cpu_time - Tiempo de CPU de usuario que la sentencia utiliza
STMT_SYS_CPU_TIME_S	BIGINT	stmt_sys_cpu_time - Tiempo de CPU de sistema que la sentencia utiliza
STMT_SYS_CPU_TIME_MS	BIGINT	stmt_sys_cpu_time - Tiempo de CPU de sistema que la sentencia utiliza
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 162. 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	La 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

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.

### Vista administrativa SNAPSTORAGE\_PATHS

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 133 en la página 444 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.

### 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).

### Función de tabla SNAP\_GET\_STORAGE\_PATHS

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 133 en la página 444 para ver una lista completa de la información que se puede devolver.

## Sintaxis

```
▶▶ SNAP_GET_STORAGE_PATHS ( (nombre-bd [ , núm-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.

### *núm-partición-bd*

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. Si *nombrebd* no se establece en NULL y *núm\_partición\_bd* se establece en NULL, implícitamente se establece -1 para *núm\_partición\_bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_STORAGE\_PATHS.

## 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 163. 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 la base de datos
DB_STORAGE_PATH	VARCHAR(256)	db_storage_path - Vía de acceso de almacenamiento automático
DBPARTITIONNUM	SMALLINT	La 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 134 en la página 446 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL para la vista administrativa SNAPSUBSECTION y privilegio EXECUTE en la función de tabla SNAP\_GET\_SUBSECTION.

### 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

### Función de tabla SNAP\_GET\_SUBSECTION

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 134 en la página 446 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 ( ( nombre-bd [ , nm-particin-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.

### *núm-partición-bd*

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. Si *nombrebd* no se establece en NULL y *núm-partición-bd* se establece en NULL, implícitamente se establece -1 para *núm-partición-bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_SUBSECTION.

## 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 164. 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.

Tabla 164. 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
DB_NAME	VARCHAR(128)	db_name - Nombre de la 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 tablas desbordados
TQ_CUR_SEND_SPILLS	BIGINT	tq_cur_send_spills - Número actual de almacenamientos intermedios de cola de tablas desbordados
TQ_MAX_SEND_SPILLS	BIGINT	tq_max_send_spills - Número máximo de desbordamientos de almacenamientos intermedios de cola de tablas
TQ_ROWS_READ	BIGINT	tq_rows_read - Número de filas leídas desde las colas de tablas
TQ_ROWS_WRITTEN	BIGINT	tq_rows_written - Número de filas grabadas en las colas de tablas
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 que la subsección utiliza
SS_USR_CPU_TIME_MS	BIGINT	ss_usr_cpu_time - Tiempo de CPU de usuario que la subsección utiliza
SS_SYS_CPU_TIME_S	BIGINT	ss_sys_cpu_time - Tiempo de CPU del sistema que la subsección utiliza
SS_SYS_CPU_TIME_MS	BIGINT	ss_sys_cpu_time - Tiempo de CPU del sistema que la subsección utiliza
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"> <li>• EXEC</li> <li>• TQ_WAIT_TO_RCV</li> <li>• TQ_WAIT_TO_SEND</li> <li>• COMPLETED</li> </ul>
SS_NODE_NUMBER	SMALLINT	ss_node_number - Número de nodo de subsección
TQ_NODE_WAITED_FOR	SMALLINT	tq_node_waited_for - En espera de nodo en una cola de tablas

Tabla 164. 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_WAIT_FOR_ANY	INTEGER	tq_wait_for_any - En espera de que cualquier nodo se envíe en una cola de tablas
TQ_ID_WAITING_ON	INTEGER	tq_id_waiting_on - En espera de nodo en una cola de tablas
DBPARTITIONNUM	SMALLINT	La 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 acerca del estado del conmutador de instantánea 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 135 en la página 450 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL para la vista administrativa SNAPSWITCHES y privilegio EXECUTE para la función de tabla SNAP\_GET\_SWITCHES.

### 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

## Función de tabla SNAP\_GET\_SWITCHES

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 135 en la página 450 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 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. Si no se utiliza esta opción de entrada, todas las particiones de la base de datos devolverán datos.

Si *núm-partición-bd* se establece en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_SWITCHES.

### 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 165. 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.
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.

Tabla 165. Información que devuelve la vista administrativa SNAPSWITCHES y la función de tabla SNAP\_GET\_SWITCHES (continuación)

Nombre de columna	Tipo de datos	Descripción
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	La 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 136 en la página 453 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL para la vista administrativa SNAPTAB y privilegio EXECUTE para la función de tabla SNAP\_GET\_TAB\_V91.

### Ejemplo

Recuperar el esquema y el nombre de todas las tablas activas.

```
SELECT SUBSTR(TABSCHEMA,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).

## Función de tabla SNAP\_GET\_TAB\_V91

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 136 en la página 453 para ver una lista completa de la información que se puede devolver.

## Sintaxis

```

▶▶ SNAP_GET_TAB_V91 (—nombre-bd [ , núm-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 NULL o una serie vacía para tomar la instantánea desde la base de datos conectada actualmente.

### *núm-partición-bd*

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. Si *nombrebd* no se establece en NULL y *núm-partición\_bd* se establece en NULL, implícitamente se establece -1 para *núm-partición\_bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_TAB\_V91.

## 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 166. 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"><li>• USER_TABLE</li><li>• DROPPED_TABLE</li><li>• TEMP_TABLE</li><li>• CATALOG_TABLE</li><li>• REORG_TABLE</li></ul>
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

Tabla 166. 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
OVERFLOW_ACCESSES	BIGINT	overflow_accesses - Accesos a los registros desbordados
PAGE_REORGS	BIGINT	page_reorgs - Reorganizaciones de página
DBPARTITIONNUM	SMALLINT	La partición de base de datos desde la que se han recuperado los datos para esta fila.
TBSP_ID	BIGINT	tablespace_id - Identificación del 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 137 en la página 456 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL para la vista administrativa SNAPTAB\_REORG y privilegio EXECUTE para la función de tabla SNAP\_GET\_TAB\_REORG.

### 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 registros(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

## 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 137 en la página 456 para ver una lista completa de la información que se puede devolver.

## Sintaxis

```

▶▶ SNAP_GET_TAB_REORG ( ( nombre-bd [ , núm-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 NULL o una serie vacía para tomar la instantánea desde la base de datos conectada actualmente.

### núm-partición-bd

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. Si *nombrebd* no se establece en NULL y *núm-partición-bd* se establece en NULL, implícitamente se establece -1 para *núm-partición-bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_TAB\_REORG.

## 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(TABSCHEMA, 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
...
...
```

## Información devuelta

Tabla 167. 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

Tabla 167. 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_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"> <li>• BUILD</li> <li>• DICT_SAMPLE</li> <li>• INDEX_RECREATE</li> <li>• REPLACE</li> <li>• SORT</li> </ul> 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 167. 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"> <li>• RECLAIM</li> <li>• RECLUSTER</li> </ul> <p>y uno de los siguientes:</p> <ul style="list-style-type: none"> <li>• +OFFLINE</li> <li>• +ONLINE</li> </ul> <p>Si se especifica la modalidad de acceso, es uno de los siguientes:</p> <ul style="list-style-type: none"> <li>• +ALLOW_NONE</li> <li>• +ALLOW_READ</li> <li>• +ALLOW_WRITE</li> </ul> <p>Si está fuera de línea y con la opción RECLUSTER, uno de los siguientes:</p> <ul style="list-style-type: none"> <li>• +INDEXSCAN</li> <li>• +TABLESCAN</li> </ul> <p>Si está fuera de línea, uno de los siguientes:</p> <ul style="list-style-type: none"> <li>• +LONGLOB</li> <li>• +DATAONLY</li> </ul> <p>Si está fuera de línea, y se especifica una opción, uno de los siguientes:</p> <ul style="list-style-type: none"> <li>• +CHOOSE_TEMP</li> <li>• +KEEPDICTIONARY</li> <li>• +RESETDICTIONARY</li> </ul> <p>Si está en línea, y se especifica una opción:</p> <ul style="list-style-type: none"> <li>• +NOTRUNCATE</li> </ul> <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 167. 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"> <li>• COMPLETED</li> <li>• PAUSED</li> <li>• STARTED</li> <li>• STOPPED</li> <li>• TRUNCATE</li> </ul>
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"> <li>• FAIL</li> <li>• SUCCESS</li> </ul>
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 tabla donde se reorganiza la tabla
DBPARTITIONNUM	SMALLINT	La 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 138 en la página 462 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL para la vista administrativa SNAPTbsp y privilegio EXECUTE para la función de tabla SNAP\_GET\_TBSP\_V91.

### 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).

### Función de tabla SNAP\_GET\_TBSP\_V91

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 138 en la página 462 para ver una lista completa de la información que se puede devolver.

## Sintaxis

```
▶▶ SNAP_GET_TBSP_V91 ( ( nombre-bd [ , núm-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 NULL o una serie vacía para tomar la instantánea desde la base de datos conectada actualmente.

### *núm-partición-bd*

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. Si *nombrebd* no se establece en NULL y *núm-partición-bd* se establece en NULL, implícitamente se establece -1 para *núm-partición-bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_TBSP\_V91.

## 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 168. Información devuelta por la vista administrativa *SNAPT BSP* 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 del 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"> <li>• DMS</li> <li>• SMS</li> </ul>
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"> <li>• ANY</li> <li>• LARGE</li> <li>• SYSTEMP</li> <li>• USRTEMP</li> </ul>
TBSP_PAGE_SIZE	BIGINT	tablespace_page_size - Tamaño de página de espacio de tabla
TBSP_EXTENT_SIZE	BIGINT	tablespace_extent_size - Tamaño de extensión de espacio de tabla
TBSP_PREFETCH_SIZE	BIGINT	tablespace_prefetch_size - Tamaño de captación previa de espacio de tabla
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

Tabla 168. 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_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 la 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 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
POOL_ASYNC_INDEX_READS	BIGINT	pool_async_index_reads - Lecturas de índices asíncronas de la 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 índices asíncronas de la agrupación de almacenamientos intermedios

Tabla 168. 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_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 la agrupación de almacenamientos intermedios
POOL_ASYNC_XDA_WRITES	BIGINT	pool_async_xda_writes - Grabaciones de datos XDA asíncronas 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_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 la agrupación de almacenamientos intermedios
POOL_ASYNC_WRITE_TIME	BIGINT	pool_async_write_time - Tiempo de grabación asíncrona de la agrupación de almacenamientos intermedios
POOL_ASYNC_DATA_READ_REQS	BIGINT	pool_async_data_read_reqs - Peticiones de lectura asíncrona de la agrupación de almacenamientos intermedios
POOL_ASYNC_INDEX_READ_REQS	BIGINT	pool_async_index_read_reqs - Peticiones de lectura de índices asíncronos de la agrupación de almacenamientos intermedios

Tabla 168. 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_ASYNC_XDA_READ_REQS	BIGINT	pool_async_xda_reqs - Peticiones de lectura XDA asíncronas de la agrupación de almacenamientos intermedios
POOL_NO_VICTIM_BUFFER	BIGINT	pool_no_victim_buffer - Almacenamientos intermedios sin víctimas de la agrupación de almacenamientos intermedios
DIRECT_READS	BIGINT	direct_reads - Lecturas directas de base de datos
DIRECT_WRITES	BIGINT	direct_writes - Grabaciones directas en 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
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"> <li>• NO_REBAL</li> <li>• FWD_REBAL</li> <li>• REV_REBAL</li> </ul>
TBSP_USING_AUTO_STORAGE	SMALLINT	tablespace_using_auto_storage - Uso del almacenamiento automático
TBSP_AUTO_RESIZE_ENABLED	SMALLINT	tablespace_auto_resize_enabled - Dimensionamiento automático habilitado
DBPARTITIONNUM	SMALLINT	La 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

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 139 en la página 468 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\_V91.

### 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).

### Función de tabla SNAP\_GET\_TBSP\_PART\_V91

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 139 en la página 468 para ver una lista completa de la información que se puede devolver.

## Sintaxis

►► SNAP\_GET\_TBSP\_PART\_V91 (—*nombre-bd*—  
, *núm-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 NULL o una serie vacía para tomar la instantánea desde la base de datos conectada actualmente.

### *núm-partición-bd*

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. Si *nombrebd* no se establece en NULL y *núm\_partición\_bd* se establece en NULL, implícitamente se establece -1 para *núm\_partición\_bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_TBSP\_PART\_V91.

## 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 169. 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 del espacio de tablas

Tabla 169. 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 tabla. Esta interfaz devuelve un identificador de texto basado en las sentencias define de <i>sqlutil.h</i>, y es una combinación de los siguientes elementos, separados por un signo '+':</p> <ul style="list-style-type: none"> <li>• BACKUP_IN_PROGRESS</li> <li>• BACKUP_PENDING</li> <li>• DELETE_PENDING</li> <li>• DISABLE_PENDING</li> <li>• DROP_PENDING</li> <li>• LOAD_IN_PROGRESS</li> <li>• LOAD_PENDING</li> <li>• NORMAL</li> <li>• OFFLINE</li> <li>• PSTAT_CREATION</li> <li>• PSTAT_DELETION</li> <li>• QUIESCED_EXCLUSIVE</li> <li>• QUIESCED_SHARE</li> <li>• QUIESCED_UPDATE</li> <li>• REBAL_IN_PROGRESS</li> <li>• REORG_IN_PROGRESS</li> <li>• RESTORE_IN_PROGRESS</li> <li>• RESTORE_PENDING</li> <li>• ROLLFORWARD_IN_PROGRESS</li> <li>• ROLLFORWARD_PENDING</li> <li>• STORDEF_ALLOWED</li> <li>• STORDEF_CHANGED</li> <li>• STORDEF_FINAL_VERSION</li> <li>• STORDEF_PENDING</li> <li>• SUSPEND_WRITE</li> </ul>
TBSP_PREFETCH_SIZE	BIGINT	tablespace_prefetch_size - Tamaño de captación previa de espacio de tabla
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 tabla de cambio de estado
TBSP_MIN_RECOVERY_TIME	TIMESTAMP	tablespace_min_recovery_time - Tiempo de recuperación mínimo para avance
TBSP_TOTAL_PAGES	BIGINT	tablespace_total_pages - Páginas totales en espacio de tabla

Tabla 169. 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 útiles en espacio de tabla
TBSP_USED_PAGES	BIGINT	tablespace_used_pages - Páginas utilizadas en espacio de tabla
TBSP_FREE_PAGES	BIGINT	tablespace_free_pages - Páginas libres en espacio de tabla
TBSP_PENDING_FREE_PAGES	BIGINT	tablespace_pending_free_pages - Páginas libres pendientes en espacio de tabla
TBSP_PAGE_TOP	BIGINT	tablespace_page_top - Nivel límite de espacio de tabla
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"> <li>• FWD_REBAL</li> <li>• NO_REBAL</li> <li>• REV_REBAL</li> </ul>
REBALANCER_EXTENTS_REMAINING	BIGINT	tablespace_rebalancer_extents_remaining - Número total de extensiones que ha de procesar el reequilibrador
REBALANCER_EXTENTS_PROCESSED	BIGINT	tablespace_rebalancer_extents_processed - Número de extensiones que el reequilibrador ha procesado
REBALANCER_PRIORITY	BIGINT	tablespace_rebalancer_priority - Prioridad del 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 en la correlación del espacio de tabla
TBSP_NUM_CONTAINERS	BIGINT	tablespace_num_containers - Número de contenedores en espacio de tabla
TBSP_INITIAL_SIZE	BIGINT	tablespace_initial_size - Tamaño inicial de espacio de tabla
TBSP_CURRENT_SIZE	BIGINT	tablespace_current_size - Tamaño actual de espacio de tabla
TBSP_MAX_SIZE	BIGINT	tablespace_max_size - Tamaño máximo de espacio de tabla

Tabla 169. 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	La 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 140 en la página 473 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización `SYSMON`
- Privilegio `SELECT` o `CONTROL` para la vista administrativa `SNAPTbsp_QUIESCER` y privilegio `EXECUTE` para la función de tabla `SNAP_Get_Tbsp_QUIESCER`.

### 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
...		

## 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 140 en la página 473 para ver una lista completa de la información que se puede devolver.

## Sintaxis

```
→ SNAP_GET_TBSP QUIESCER ( nombre-bd [ , nm-particin-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 vlido 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 vaca para tomar la instantnea desde la base de datos conectada actualmente.

### nm-particin-bd

Argumento de entrada opcional de tipo INTEGER que especifica un nmero de particin de base de datos vlido. Especifique -1 para la particin de base de datos actual o -2 para un agregado de todas las particiones de base de

datos. Si *nombrebd* no se establece en NULL y *núm\_partición\_bd* se establece en NULL, implícitamente se establece -1 para *núm\_partición\_bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_TBSP QUIESCER.

## 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 170. Información devuelta por la vista administrativa SNAP\_TBSP 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

Tabla 170. Información devuelta por la vista administrativa *SNAPTbsp\_QUIESCER* y la función de tabla *SNAP\_GET\_TBSP\_QUIESCER* (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
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 sqlutil.h y es uno de los siguientes: <ul style="list-style-type: none"> <li>• EXCLUSIVE</li> <li>• UPDATE</li> <li>• SHARE</li> </ul>
DBPARTITIONNUM	SMALLINT	La 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 141 en la página 477 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL para la vista administrativa *SNAPTbsp\_RANGE* y privilegio EXECUTE para la función de tabla *SNAP\_GET\_TBSP\_RANGE*.

## 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\_TBSP\_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 141 en la página 477 para ver una lista completa de la información que se puede devolver.

## Sintaxis

```
▶▶ SNAP_GET_TBSP_RANGE ( (nombre-bd [ , núm-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 NULL o una serie vacía para tomar la instantánea desde la base de datos conectada actualmente.

### *núm-partición-bd*

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. Si *nombrebd* no se establece en NULL y *núm-partición-bd* se establece en NULL, implícitamente se establece -1 para *núm-partición-bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_TBSP\_RANGE.

## 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 171. 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 del 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

Tabla 171. Información devuelta por la vista administrativa SNAPTBSP\_RANGE y la función de tabla SNAP\_GET\_TBSP\_RANGE (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 esta fila.

## Vista administrativa SNAPUTIL y función de tabla SNAP\_GET\_UTIL – Recuperar información acerca de la 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 142 en la página 480 para ver una lista completa de la información que se puede devolver.

### Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL para la vista administrativa SNAPUTIL y privilegio EXECUTE para la función de tabla SNAP\_GET\_UTIL.

### 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 registros(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

```

## Función de tabla SNAP\_GET\_UTIL

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 142 en la página 480 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 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. Si no se utiliza esta opción de entrada, todas las particiones de la base de datos devolverán datos.

Si *núm-partición-bd* se establece en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_UTIL.

## 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 172. 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 basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> <li>• ASYNC_INDEX_CLEANUP</li> <li>• BACKUP</li> <li>• CRASH_RECOVERY</li> <li>• LOAD</li> <li>• REBALANCE</li> <li>• REDISTRIBUTE</li> <li>• REORG</li> <li>• RESTART_RECREATE_INDEX</li> <li>• RESTORE</li> <li>• ROLLFORWARD_RECOVERY</li> <li>• RUNSTATS</li> </ul>
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.
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

Tabla 172. 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_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"> <li>• ERROR</li> <li>• EXECUTE</li> <li>• WAIT</li> </ul>
UTILITY_INVOKER_TYPE	VARCHAR(10)	utility_invoker_type - Tipo de invocador 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"> <li>• AUTO</li> <li>• USER</li> </ul>
DBPARTITIONNUM	SMALLINT	La 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 acerca de la 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 143 en la página 483 para ver una lista completa de la información que se puede devolver.

## Autorización

- Autorización SYSMON
- Privilegio SELECT o CONTROL para la vista administrativa SNAPUTIL\_PROGRESS y privilegio EXECUTE para la función de tabla SNAP\_GET\_UTIL\_PROGRESS.

## 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).

## Función de tabla SNAP\_GET\_UTIL\_PROGRESS

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 143 en la página 483 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 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. Si no se utiliza esta opción de entrada, todas las particiones de la base de datos devolverán datos.

Si *núm-partición-bd* se establece en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_UTIL\_PROGRESS.

### 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 173. 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 basado en las sentencias define de sqlmon.h y es uno de los siguientes: <ul style="list-style-type: none"> <li>• ERROR</li> <li>• EXECUTE</li> <li>• WAIT</li> </ul>
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.



Tabla 173. Información devuelta por la vista administrativa SNAPUTIL\_PROGRESS y la función de tabla SNAP\_GET\_UTIL\_PROGRESS (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
PROGRESS_WORK_METRIC	VARCHAR(16)	progress_work_metric - Métrica de trabajo en curso. 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"> <li>• NOT_SUPPORT</li> <li>• BYTES</li> <li>• EXTENTS</li> <li>• INDEXES</li> <li>• PAGES</li> <li>• ROWS</li> <li>• TABLES</li> </ul>
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	La 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—,—núm-partición-bd—)————◄◄
```

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, que pueden ser uno de los siguientes:

- APPL\_ALL
- BUFFERPOOLS\_ALL
- DB2
- DBASE\_ALL
- DBASE\_LOCKS
- DBASE\_TABLES
- DBASE\_TABLESPACES

- DYNAMIC\_SQL

*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 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 desde la base de datos conectada actualmente.

*núm-partición-bd*

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. Si se especifica un valor nulo, se establece -1 de forma implícita.

## Autorización

Para ejecutar el procedimiento, el usuario debe tener autorización SYSADM, SYSCTRL, SYSMAINT o SYSMON. Los usuarios que no tengan autorización SYSADM, SYSCTRL, SYSMAINT 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 sobre 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 utilización y configuración 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

- Privilegio SELECT o CONTROL para las vistas administrativas TBSP\_UTILIZATION, SNAPTbsp, SNAPTbsp\_PART y la vista de catálogo SYSCAT.TABLESPACE.
- Para acceder a los datos del supervisor de instantáneas también se necesita autorización SYSMON, SYSCTRL, SYSMaint o 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 174. 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 del 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"> <li>• DMS</li> <li>• SMS</li> </ul>

Tabla 174. Información devuelta por la vista administrativa  
TBSP\_UTILIZATION (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
TBSP_CONTENT_TYPE	VARCHAR(10)	<p>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:</p> <ul style="list-style-type: none"> <li>• ANY</li> <li>• LONG</li> <li>• SYSTEMP</li> <li>• USRTEMP</li> </ul>
TBSP_CREATE_TIME	TIMESTAMP	Hora de creación del espacio de tablas.
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"> <li>• BACKUP_IN_PROGRESS</li> <li>• BACKUP_PENDING</li> <li>• DELETE_PENDING</li> <li>• DISABLE_PENDING</li> <li>• DROP_PENDING</li> <li>• LOAD_IN_PROGRESS</li> <li>• LOAD_PENDING</li> <li>• NORMAL</li> <li>• OFFLINE</li> <li>• PSTAT_CREATION</li> <li>• PSTAT_DELETION</li> <li>• QUIESCED_EXCLUSIVE</li> <li>• QUIESCED_SHARE</li> <li>• QUIESCED_UPDATE</li> <li>• REBAL_IN_PROGRESS</li> <li>• REORG_IN_PROGRESS</li> <li>• RESTORE_IN_PROGRESS</li> <li>• RESTORE_PENDING</li> <li>• ROLLFORWARD_IN_PROGRESS</li> <li>• ROLLFORWARD_PENDING</li> <li>• STORDEF_ALLOWED</li> <li>• STORDEF_CHANGED</li> <li>• STORDEF_FINAL_VERSION</li> <li>• STORDEF_PENDING</li> <li>• SUSPEND_WRITE</li> </ul>
TBSP_TOTAL_SIZE_KB	BIGINT	El tamaño total del espacio de tablas en KB, calculado como páginas_total*tamaño_página/1024.

Tabla 174. Información devuelta por la vista administrativa  
TBSP\_UTILIZATION (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
TBSP_USABLE_SIZE_KB	BIGINT	El tamaño total utilizable del espacio de tablas en KB, calculado como $\text{págs\_utilizables} * \text{tamaño\_pág} / 1024$ .
TBSP_USED_SIZE_KB	BIGINT	El tamaño total utilizado del espacio de tablas en KB, calculado como $\text{páginas\_utilizadas} * \text{tamaño\_página} / 1024$ .
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 - Páginas totales en espacio de tabla
TBSP_USABLE_PAGES	BIGINT	tablespace_usable_pages - Páginas útiles en espacio de tabla
TBSP_USED_PAGES	BIGINT	tablespace_used_pages - Páginas utilizadas en espacio de tabla
TBSP_FREE_PAGES	BIGINT	tablespace_free_pages - Páginas libres en espacio de tabla
TBSP_PAGE_TOP	BIGINT	tablespace_page_top - Nivel límite de espacio de tabla
TBSP_PAGE_SIZE	INTEGER	tablespace_page_size - Tamaño de página de espacio de tabla
TBSP_EXTENT_SIZE	INTEGER	tablespace_extent_size - Tamaño de extensión de espacio de tabla
TBSP_PREFETCH_SIZE	BIGINT	tablespace_prefetch_size - Tamaño de captación previa de espacio de tabla
TBSP_MAX_SIZE	BIGINT	tablespace_max_size - Tamaño máximo de espacio de tabla
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

Tabla 174. Información devuelta por la vista administrativa  
TBSP\_UTILIZATION (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
TBSP_LAST_RESIZE_FAILED	SMALLINT	tablespace_last_resize_failed - Último intento de redimensionamiento fallido
TBSP_USING_AUTO_STORAGE	SMALLINT	tablespace_using_auto_storage - Uso del almacenamiento automático
TBSP_AUTO_RESIZE_ENABLED	SMALLINT	tablespace_auto_resize_enabled - Dimensionamiento automático habilitado
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 en espacio de tabla
REMARKS	VARCHAR(254)	Comentario suministrado por el usuario.
DBPARTITIONNUM	SMALLINT	La 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

- Privilegio SELECT o CONTROL para las vistas administrativas TOP\_DYNAMIC\_SQL y SNAPDYN\_SQL.
- Para acceder a los datos del supervisor de instantáneas también se necesita autorización SYSMON, SYSCTRL, SYSMOINT o 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 175. 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 sentencias
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	La partición de base de datos desde la que se han recuperado los datos para esta fila.

## Rutinas de procedimientos de SQL

### 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  
► (—*blobsar*—, —*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. La persona que invoca el procedimiento GET\_ROUTINE\_SAR debe tener autorización DBADM.

*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 extrae texto de la rutina de los catálogos. Los valores válidos son:

- 0 Dejar el texto de la rutina intacto. Este 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

```

▶▶—PUT_ROUTINE_SAR—————▶
▶—(—blobsar—————)————▶
   |, —propietario-nuevo—, —distintivo-registro-uso—|

```

El esquema es SYSFUN.

El procedimiento PUT\_ROUTINE\_SAR pasa el archivo necesario para crear una rutina SQL en el servidor y, a continuación, define la rutina. La persona que invoca el procedimiento PUT\_ROUTINE\_SAR debe tener autorización DBADM.

### *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.

## REBIND\_ROUTINE\_PACKAGE

►►—REBIND\_ROUTINE\_PACKAGE—(—*tipo*—,—*serie-nombre-rutina*—,—*resolver*—)—►►

El esquema es SYSPROC.

El procedimiento REBIND\_ROUTINE\_PACKAGE vuelve a vincular el paquete asociado a un procedimiento SQL. 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.

### *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 el registro especial CURRENT SCHEMA cuando se procesa la rutina. La *serie-nombre-rutina* no puede incluir comillas dobles ("").

### *resolver*

Argumento de entrada de tipo VARCHAR(12) que especifica la semántica de

vinculación que debe utilizarse. Un valor de 'ANY' indica que todas las coincidencias posibles en la vía de acceso a SQL están pensadas para resolver referencias a cualquier objeto que utilice la vía de acceso a SQL para la resolución de objetos. Un valor de 'CONSERVATIVE' indica que sólo aquellos objetos definidos antes de la última indicación de fecha y hora se consideran para la resolución.

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.

## 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".

Ejemplo:

```
CALL SYSPROC.SET_ROUTINE_OPTS(CAST (NULL AS VARCHAR(1)))
```

---

## 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 eliminando 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, sin espacios incluidos.

## Autorización

- SYSADM, SYSMON, SYSCTRL o SYSMANT
- 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 sobre 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 176. Información devuelta por el procedimiento ANALYZE\_LOG\_SPACE

Nombre de columna	Tipo de columna	Descripción
PARTITION_NUM	SMALLINT	Número de la partición de base de datos del análisis del espacio de anotación cronológica.
TOTAL_LOG_SIZE	BIGINT	Espacio total de anotación cronológica asignado (en bytes); -1 indica un espacio ilimitado.
AVAIL_LOG_SPACE	BIGINT	Cantidad de espacio de anotación cronológica en bytes que es libre y que el proceso de redistribución puede utilizar.
DATA_SKEW	BIGINT	Valor absoluto en bytes del tamaño de los datos que se desvía del nivel de destino.
REQ_LOG_SPACE	BIGINT	Cantidad necesaria de espacio, indicada en bytes, para llegar a la distribución de datos deseada.
NUM_OF_STEPS	SMALLINT	Número de pasos necesarios para reducir el desvío de datos a cero.
MAX_STEP_SIZE	BIGINT	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 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 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 bitwise desde Tabla 177 en la página 638, 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 entero de (REDIST\_STAGE\_SIZE | REDIST\_NEXT\_STEP), el llamador de esta función sólo debe proporcionar *stageSize* y *nextStep* para recibir los valores, y el resto de los parámetros de salida pueden ser nulos.

Tabla 177. 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 178 que indican los campos de destino que se grabarán o sobre los que se grabará en el registro de valores de redistribución.

Tabla 178. 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 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 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.

### 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)
```

Si desea ver un ejemplo de uso completo de los procedimientos de redistribución por pasos, consulta Procedimiento STEPWISE\_REDISTRIBUTE\_DBPG

### Notas sobre 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 SQL COMMIT, no se soporta la ejecución del proceso de redistribución en una conexión de tipo 2.

## 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 179. Tabla STMG\_OBJECT\_TYPE

Nombre de columna	Tipo de datos	Posibilidad de nulos	Descripción
OBJ_TYPE	INTEGER	N	Valor entero que corresponde a un tipo de objeto de almacenamiento <ul style="list-style-type: none"> <li>• 0 - Base de datos</li> <li>• 1 - Grupo de particiones de base de datos</li> <li>• 2 - Espacio de tabla</li> <li>• 3 - Contenedor de espacio de tablas</li> <li>• 4 - Tabla</li> <li>• 5 - Índice</li> </ul>
TYPE_NAME	VARCHAR	N	Nombre descriptivo del tipo de objeto de almacenamiento <ul style="list-style-type: none"> <li>• STMG_DATABASE</li> <li>• STMG_DBPGROUP</li> <li>• STMG_TABLESPACE</li> <li>• STMG_CONTAINER</li> <li>• STMG_TABLE</li> <li>• STMG_INDEX</li> </ul>

## 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 tabla
- 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 – Creación de 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 o bien:

- 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.
- La autorización SYSADM o DBADM

## Notas sobre 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 – Eliminar 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 SYSADM o DBADM
- Privilegio DROPIN para el esquema de dichas tablas
- Privilegio CONTROL para dichas tablas

---

## 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 de la función de tabla

*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 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 sobre 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

Este procedimiento hace que se recopile información sobre una determinada actividad y se grabe en el supervisor de sucesos de actividades activas. Cuando se aplica a una actividad con actividades hijo, este procedimiento genera de forma recurrente un registro correspondiente a cada actividad hijo hasta el nivel inferior. Esta información se recopila y se envía cuando se llama a este procedimiento. No se espera hasta que finalice la ejecución de la actividad. 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 de la función de tabla

*descriptorcontexto\_aplicación*

Argumento de entrada de tipo BIGINT que especifica el descriptor de contexto de aplicación cuya actividad se va a capturar. 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 capturar. 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 capturar. Si el argumento es nulo, no se encontrará ninguna actividad y se devolverá SQL4702N con SQLSTATE 5U035.

## Autorización

Privilegio EXECUTE para el procedimiento  
WLM\_CAPTURE\_ACTIVITY\_IN\_PROGRESS

## Ejemplo

Supongamos que un procedimiento particular MYSCHEMA.MYSLOWSTP se ejecuta de forma más lenta de lo habitual. Un usuario se queja y 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 tiene la posibilidad de capturar la actividad del procedimiento almacenado y cualquiera de las actividades anidadas en la misma.

Suponiendo que un supervisor de sucesos correspondiente a actividades de DB2 llamado DB2ACTIVITIES existe y se ha activado, el administrador puede utilizar la función WLM\_GET\_WORKLOAD\_OCCURRENCE\_ACTIVITIES para obtener el descriptor de contexto de aplicación, el ID de unidad de trabajo y el ID de actividad correspondientes a 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, para un supervisor de sucesos de actividad llamado DB2ACTIVITIES el administrador puede utilizar la siguiente función de tabla para ver dónde ha empleado su tiempo la actividad:

```
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, CHILDSTMT.STMT_TEXT, CHILD.ACT_EXEC_TIME  
  FROM RAH PARENT, ACTIVITY_DB2ACTIVITIES CHILD,
```

```

        ACTIVITYSTMT_DB2ACTIVITIES CHILDSTMT
WHERE PARENT.APPL_ID = CHILD.APPL_ID AND
      CHILD.APPL_ID = CHILDSTMT.APPL_ID AND
      PARENT.UOW_ID = CHILD.PARENT_UOW_ID AND
      CHILD.UOW_ID = CHILDSTMT.UOW_ID AND
      PARENT.ACTIVITY_ID = CHILD.PARENT_ACTIVITY_ID AND
      CHILD.ACTIVITY_ID = CHILDSTMT.ACTIVITY_ID AND
      PARENT.LEVEL < 64
    )
SELECT UOW_ID, ACTIVITY_ID, SUBSTR(STMT_TEXT,1,40),
       ACT_EXEC_TIME AS
       LIFE_TIME
FROM RAH
ORDER BY UOW_ID, ACTIVITY_ID;

```

U continuación se muestra un ejemplo de una consulta que utiliza la función de tabla:

```

SELECT * FROM TABLE(SHOWCAPTUREDACTIVITY(1, 2, 3))
AS ACTS ORDER BY UOW_ID, ACTIVITY_ID

```

### Nota sobre uso

Si no hay ningún supervisor de sucesos de actividades activo, se devuelve un SQL1633W con SQLSTATE 01H53.

Si utiliza este procedimiento para recopilar información de actividad para un procedimiento que tiene parámetros INOUT, los valores de INOUT pueden quedar sobregabados cuando se produce la captura. Esta situación no se produce si ha creado la carga de trabajo, la clase de servicio, la acción de trabajo o el umbral de predicción para los que está capturando datos de actividad como COLLECT ACTIVITY WITH DETAILS AND VALUES o si modifica el objeto y especifica la palabra clave COLLECT ACTIVITY DATA con la cláusula ON COORDINATOR o ON ALL y con las cláusulas WITH DETAILS AND VALUES.

## WLM\_COLLECT\_STATS - Coleccionar y restaurar estadísticas de gestión de carga de trabajo

Este procedimiento crea estadísticas para clases de servicio, cargas de trabajo, clases de trabajo y colas de umbral que debe reunir y grabar el supervisor de eventos de estadísticas. Las estadísticas para clases de servicio, cargas de trabajo, clases de trabajo y colas de umbral también se restauran. Si no hay ningún supervisor de eventos de estadísticas activos, las estadísticas sólo se restaurarán.

### 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 retorno = 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 sobre uso

El procedimiento WLM\_COLLECT\_STATS se utiliza para reunir estadísticas de forma manual. 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 este procedimiento se llama al mismo tiempo que otra solicitud de reunión y restauración en proceso (por ejemplo, el procedimiento se llama mientras otra invocación está en proceso o mientras se realiza una recolección automática) se devuelve un mensaje de aviso, SQL1632W con SQLSTATE 01H53 y la solicitud se ignora.

El procedimiento WLM\_COLLECT\_STATS sólo inicia el proceso de recolección y restauración. Puede volver antes de que el proceso haya finalizado, es decir, 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 el intervalo de recolección previo o en el nuevo intervalo de colección que se acaba de iniciar.

## WLM\_GET\_ACTIVITY\_DETAILS - Devolver información detallada sobre una actividad específica

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.

### 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 Tabla 181 en la página 652 y Tabla 182 en la página 655.

```
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_PARTITIÖN_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_PARTITIÖN_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         0
1   PARENT_ACTIVITY_ID    0
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 180. 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 Tabla 181 y Tabla 182 en la página 655 para ver los posibles valores.
VALUE	VARCHAR(1024)	Valores de elemento. Consulte Tabla 181 y Tabla 182 en la página 655 para ver los valores posibles.

Tabla 181. Elementos devueltos

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.
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.
ACTIVITY_ID	Identificador exclusivo de actividad dentro de una aplicación.

Tabla 181. 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.
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.
ACTIVITY_STATE	Los valores posibles incluyen: <ul style="list-style-type: none"> <li>• CANCEL_PENDING</li> <li>• EXECUTING</li> <li>• IDLE</li> <li>• INITIALIZING</li> <li>• QP_CANCEL_PENDING</li> <li>• QP_QUEUED</li> <li>• QUEUED</li> <li>• TERMINATING</li> <li>• UNKNOWN</li> </ul>
ACTIVITY_TYPE	Los valores posibles incluyen: <ul style="list-style-type: none"> <li>• CALL</li> <li>• DDL</li> <li>• LOAD</li> <li>• OTHER</li> <li>• READ_DML</li> <li>• WRITE_DML</li> </ul>
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.
INVOCATION_ID	Distingue una determinada invocación de esta actividad de otras del mismo nivel de anidamiento.
ROUTINE_ID	Identificador exclusivo de la rutina.
UTILITY_ID	Si la actividad es un programa de utilidad, es el ID del programa de utilidad. De lo contrario, este campo es 0.
SERVICE_CLASS_ID	Identificador exclusivo de la clase de servicio a la que pertenece esta 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.

Tabla 181. Elementos devueltos (continuación)

Nombre de elemento	Descripción
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.
ENTRY_TIME	La hora a la que esta actividad ha llegado al sistema.
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.
LAST_REFERENCE_TIME	Cada vez que se produce una solicitud en esta actividad, este campo se actualiza.
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.
SECTION_NUMBER	Si la actividad es una sentencia SQL, representa su número de sección.
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.
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.
QUERY_COST_ESTIMATE	Coste estimado, en timerons, para una consulta, determinado por el compilador de SQL.
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.



Tabla 181. Elementos devueltos (continuación)

Nombre de elemento	Descripción
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 DPE, 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.
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 hora no están activados, este elemento no se recopila y se graba el valor -1.
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 hora no están activados, este elemento no se recopila y se graba el valor -1.
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.

Se devuelve la siguiente información únicamente si los umbrales correspondientes se aplican a la actividad.

Tabla 182. Elementos devueltos si se aplica

Nombre de elemento	Descripción
CONCURRENTWORKLOADACTIVITIES_THRESHOLD_ID	ID del umbral.
CONCURRENTWORKLOADACTIVITIES_THRESHOLD_VALUE	Valor que, cuando se supera, desencadenará el umbral.
CONCURRENTWORKLOADACTIVITIES_THRESHOLD_VIOLATED	Yes indica que esta actividad ha violado el umbral. No indica que esta actividad no ha violado el umbral.
CONCURRENTDBCOORDACTIVITIES_DB_THRESHOLD_ID	ID del umbral.
CONCURRENTDBCOORDACTIVITIES_DB_THRESHOLD_VALUE	Valor que, cuando se supera, desencadenará el umbral.
CONCURRENTDBCOORDACTIVITIES_DB_THRESHOLD_QUEUED	Si este umbral ha colocado la actividad en cola.
CONCURRENTDBCOORDACTIVITIES_DB_THRESHOLD_VIOLATED	'Yes' indica que se ha violado el umbral. 'No' indica que el umbral aún no se ha violado.
CONCURRENTDBCOORDACTIVITIES_WORK_ACTION_SET_THRESHOLD_ID	ID del umbral.
CONCURRENTDBCOORDACTIVITIES_WORK_ACTION_SET_THRESHOLD_VALUE	Valor que, cuando se supera, desencadenará el umbral.

Tabla 182. Elementos devueltos si se aplica (continuación)

Nombre de elemento	Descripción
CONCURRENTDBCOORDACTIVITIES_WORK_ACTION_SET_THRESHOLD_QUEUED	'Yes' indica que el umbral ha colocado en cola la actividad. 'No' indica que la actividad no se ha colocado en cola.
CONCURRENTDBCOORDACTIVITIES_WORK_ACTION_SET_THRESHOLD_VIOLATED	'Yes' indica que se ha violado el umbral. 'No' indica que el umbral aún no se ha violado.
CONCURRENTDBCOORDACTIVITIES_SUPERCLASS_THRESHOLD_ID	ID del umbral.
CONCURRENTDBCOORDACTIVITIES_SUPERCLASS_THRESHOLD_VALUE	Valor que, cuando se supera, desencadenará el umbral.
CONCURRENTDBCOORDACTIVITIES_SUPERCLASS_THRESHOLD_QUEUED	'Yes' indica que el umbral ha colocado en cola la actividad. 'No' indica que la actividad no se ha colocado en cola.
CONCURRENTDBCOORDACTIVITIES_SUPERCLASS_THRESHOLD_VIOLATED	'Yes' indica que se ha violado el umbral. 'No' indica que el umbral aún no se ha violado.
CONCURRENTDBCOORDACTIVITIES_SUBCLASS_THRESHOLD_ID	ID del umbral.
CONCURRENTDBCOORDACTIVITIES_SUBCLASS_THRESHOLD_VALUE	Valor que, cuando se supera, desencadenará el umbral.
CONCURRENTDBCOORDACTIVITIES_SUBCLASS_THRESHOLD_QUEUED	Si este umbral ha colocado la actividad en cola.
CONCURRENTDBCOORDACTIVITIES_SUBCLASS_THRESHOLD_VIOLATED	'Yes' indica que se ha violado el umbral. 'No' indica que el umbral aún no se ha violado.
ESTIMATEDSQLCOST_THRESHOLD_ID	ID del umbral.
ESTIMATEDSQLCOST_THRESHOLD_VALUE	Valor que, cuando se supera, desencadenará el umbral.
ESTIMATEDSQLCOST_THRESHOLD_VIOLATED	'Yes' indica que se ha violado el umbral. 'No' indica que el umbral aún no se ha violado.
SQLTEMPSPACE_THRESHOLD_ID	ID del umbral.
SQLTEMPSPACE_THRESHOLD_VALUE	Valor que, cuando se supera, desencadenará el umbral.
SQLTEMPSPACE_THRESHOLD_VIOLATED	'Yes' indica que se ha violado el umbral. 'No' indica que el umbral aún no se ha violado.
SQLROWSRETURNED_THRESHOLD_ID	ID del umbral.
SQLROWSRETURNED_THRESHOLD_VALUE	Valor que, cuando se supera, desencadenará el umbral.
SQLROWSRETURNED_THRESHOLD_VIOLATED	'Yes' indica que se ha violado el umbral. 'No' indica que el umbral aún no se ha violado.
ACTIVITYTOTALTIME_THRESHOLD_ID	ID del umbral.
ACTIVITYTOTALTIME_THRESHOLD_VALUE	Valor que, cuando se supera, desencadenará el umbral.

Tabla 182. Elementos devueltos si se aplica (continuación)

Nombre de elemento	Descripción
ACTIVITYTOTALTIME_THRESHOLD_VIOLATED	'Yes' indica que se ha violado el umbral. 'No' indica que el umbral aún no se ha violado.

## Función de tabla WLM\_GET\_QUEUE\_STATS - Devolver estadísticas de cola de umbral

Esta función devuelve estadísticas básicas de una o más colas de umbral.

Esta función devuelve una fila de estadísticas para cada cola de umbral. Las estadísticas son para colas de todas las particiones activas.

### 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 válido. Los valores posibles son:

- CONCDDBC: 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
- NULL o serie vacía: se devuelven datos para todos los predicados de umbral posibles. 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 válido. Los valores posibles son:

- DB: base de datos
- SB: subclase de servicio
- SP: superclase de servicio
- WA: conjunto de acciones de trabajo
- NULL o serie vacía: se devuelven datos para todos los dominios de umbral posibles. 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 válido. 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 válido. 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

Para ver todas las estadísticas básicas de las colas en el 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
```

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

THRESHNAME	THRESHOLD_PREDICATE	THRESHOLD_DOMAIN	...
-----	-----	-----	-----
LIMIT1	CONCDBC	DB	...
LIMIT2	SCCONN	SP	...
LIMIT3	DBCNN	DB	...

Datos de salida de esta consulta (continuación).

...	PART	QUEUE_SIZE_TOP	QUEUE_TIME_TOTAL	QUEUE_ASSIGN
...	-----	-----	-----	-----
...	0	12	1238540	734
...	0	4	741249	24
...	0	7	412785	128

## Nota sobre uso

No se realiza ninguna agregación en las colas (en una partición) o en las particiones (para una cola o más), aunque este tipo de agregación puede archivarse utilizando consultas de SQL tal como se muestra en el anterior ejemplo.

## Información devuelta

Tabla 183. 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:</p> <ul style="list-style-type: none"> <li>• CONCDBC: umbral de actividades de coordinador de base de datos simultáneo</li> <li>• DBCONN: umbral total de conexiones de partición de base de datos</li> <li>• SCCONN: umbral total de conexiones de partición de clase de servicio</li> </ul> <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:</p> <ul style="list-style-type: none"> <li>• DB: base de datos</li> <li>• SB: subclase de servicio</li> <li>• SP: superclase de servicio</li> <li>• WA: conjunto de acciones de trabajo</li> </ul> <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>El nombre exclusivo del umbral responsable de esta cola. Los valores del dominio de umbral coinciden con los de la columna THRESHOLDNAME en la vista SYSCAT.THRESHOLDS.</p>
THRESHOLD_ID	INTEGER	<p>El id exclusivo del umbral responsable de esta cola. Los valores del dominio de umbral coinciden con los de la columna THRESHOLDID en la vista SYSCAT.THRESHOLDS.</p>
DBPARTITIONNUM	SMALLINT	<p>Número de partición desde donde 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. Nulo, si el dominio del umbral no es ninguna superclase o subclase de servicio.</p>
SERVICE_SUBCLASS_NAME	VARCHAR(128)	<p>Nombre de la subclase de servicio que es el dominio del umbral responsable de esta cola. Nulo, si el dominio del umbral no es ninguna subclase de servicio.</p>

Tabla 183. Información devuelta para WLM\_GET\_QUEUE\_STATS (continuación)

Nombre de columna	Tipo de datos	Descripción
WORK_ACTION_SET_NAME	VARCHAR(128)	Nombre del conjunto de acciones de trabajo que es el dominio del umbral responsable de esta cola. 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. 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. 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. Hay cuatro posibles eventos que pueden lanzar una restauración de estadísticas que, a su vez, actualizarán esta indicación horaria:</p> <ul style="list-style-type: none"> <li>• Se llama el procedimiento WLM_COLLECT_STATS.</li> <li>• El proceso periódico de colección y restauración controlado por el parámetro de configuración WLM_COLLECT_INT provoca una colección y una restauración.</li> <li>• La base de datos se reactiva.</li> <li>• El umbral para el que se notifican las estadísticas de cola se ha modificado y el cambio se ha confirmado.</li> </ul> <p>La indicación horaria LAST_RESET es la hora local.</p>
QUEUE_SIZE_TOP	INTEGER	Mayor número de conexiones o actividades en la cola alcanzado 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.
QUEUE_TIME_LATEST	BIGINT	Tiempo transcurrido en la cola por la última conexión o actividad para dejar la cola. Se expresa en milisegundos.

Tabla 183. Información devuelta para WLM\_GET\_QUEUE\_STATS (continuación)

Nombre de columna	Tipo de datos	Descripción
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.

## WLM\_GET\_SERVICE\_CLASS\_AGENTS - Listar agentes en ejecución en una clase de servicio

Esta función devuelve la lista de agentes, procesos en modalidad delimitada (db2fmps) y entidades del sistema de la partición especificada que se están ejecutando en el servicio dado o en nombre de la 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—(—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 un nombre de superclase de servicio válido que está en la 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 para las que coinciden 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 para las que coinciden 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 para las que coinciden los otros parámetros. El descriptor de contexto de aplicación 0 sólo devolverá 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 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\_SERVICE\_CLASS\_AGENTS

## Ejemplo

Devolver 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"
```

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

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

Aquí vemos 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 nos indica que la solicitud es una solicitud de captación.

## Nota sobre uso

Los parámetros tienen el efecto de sumarse (operación AND). Es decir, si se especifican registros conflictivos como un nombre de superclase SUP\_A y un nombre de subclase SUB\_B de modo que SUB\_B no es una subclase de SUP\_A, no se devuelve ninguna fila.

## Información devuelta

Tabla 184. 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.



Tabla 184. Información devuelta por WLM\_GET\_SERVICE\_CLASS\_AGENTS (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.
DBPARTITIONNUM	SMALLINT	Número de partición de la que se ha recopilado este registro.
ENTITY	VARCHAR(32)	Si el tipo de entidad de esta fila es un agente, este campo muestra "db2agent". Si el tipo de entidad de esta fila es un proceso en modalidad delimitada, este campo muestra "db2fmp (pid)", donde pid es el ID del proceso en modalidad delimitada. De lo contrario, se muestra 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 ocurrencia de carga de trabajo. No identifica la ocurrencia 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. Como alternativa, se puede utilizar el descriptor de contexto de la aplicación en lugar del número de partición de base de datos de coordinación.
UOW_ID	INTEGER	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.
ACTIVITY_ID	INTEGER	Identificador de actividad exclusivo dentro de una unidad de trabajo.
PARENT_UOW_ID	INTEGER	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 nulo si esta actividad no tiene actividad padre.
PARENT_ACTIVITY_ID	INTEGER	Identificador exclusiva de actividad dentro de una unidad de trabajo correspondiente a la actividad padre de la actividad cuyo ID es activity_id. Devuelve 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, este campo es nulo.
AGENT_TYPE	VARCHAR(32)	De coordinación o subagente. Si se trata de uno de coordinación, el ID de agente puede variar en entorno de concentrador. Los tipos de agente se representan mediante: <ul style="list-style-type: none"> <li>• COORDINATOR</li> <li>• OTHER</li> <li>• PDBSUBAGENT</li> <li>• SMPSUBAGENT</li> </ul>
SMP_COORDINATOR	INTEGER	Si el agente es o no de coordinación smp; 1 indica que lo es, 0 que no lo es.

Tabla 184. Información devuelta por WLM\_GET\_SERVICE\_CLASS\_AGENTS (continuación)

Nombre de columna	Tipo de datos	Descripción
AGENT_SUBTYPE	VARCHAR(32)	Los subtipos posibles incluyen: <ul style="list-style-type: none"> <li>• DSS</li> <li>• OTHER</li> <li>• RPC</li> <li>• SMP</li> </ul>
AGENT_STATE	VARCHAR(32)	Si el agente está asociado o activo. Los valores posibles son: <ul style="list-style-type: none"> <li>• ACTIVE</li> <li>• ASSOCIATED</li> </ul>
EVENT_TYPE	VARCHAR(32)	El tipo del último suceso que ha procesado este agente. Los valores posibles son: <ul style="list-style-type: none"> <li>• ACQUIRE</li> <li>• PROCESS</li> <li>• WAIT</li> </ul>
EVENT_OBJECT	VARCHAR(32)	El objeto del último suceso que ha procesado este agente. Los valores posibles son: <ul style="list-style-type: none"> <li>• COMPRESSION_DICTIONARY_BUILD</li> <li>• IMPLICIT_REBIND</li> <li>• INDEX_RECREATE</li> <li>• LOCK</li> <li>• LOCK_ESCALATION</li> <li>• QP_QUEUE</li> <li>• REMOTE_REQUEST</li> <li>• REQUEST</li> <li>• ROUTINE</li> <li>• WLM_QUEUE</li> </ul>
EVENT_STATE	VARCHAR(32)	El estado del último suceso que ha procesado este agente. Los valores posibles son: <ul style="list-style-type: none"> <li>• EXECUTING</li> <li>• IDLE</li> </ul>
REQUEST_ID	VARCHAR(64)	Exclusivo únicamente en combinación con application_handle. Se puede utilizar para distinguir entre tener una solicitud que tarde mucho y tener varias solicitudes. Por ejemplo, para distinguir varias captaciones de una captación larga.

Tabla 184. Información devuelta por WLM\_GET\_SERVICE\_CLASS\_AGENTS (continuación)

Nombre de columna	Tipo de datos	Descripción
REQUEST_TYPE	VARCHAR(32)	<p>El tipo de solicitud. Los valores posibles son:</p> <ul style="list-style-type: none"> <li>• Para agentes de coordinación: <ul style="list-style-type: none"> <li>- CLOSE</li> <li>- COMMIT</li> <li>- COMPILE</li> <li>- DESCRIBE</li> <li>- EXCSQLSET</li> <li>- EXECIMMD</li> <li>- EXECUTE</li> <li>- FETCH</li> <li>- INTERNAL &lt;número&gt;</li> <li>- OPEN</li> <li>- PREPARE</li> <li>- REBIND</li> <li>- REDISTRIBUTE</li> <li>- REORG</li> <li>- ROLLBACK</li> <li>- RUNSTATS</li> </ul> </li> <li>• Para subagentes (DSS y SMP): <ul style="list-style-type: none"> <li>- muestra el número de subsección en el formato "SUBSECCIÓN:&lt;número subsección&gt;" si el número de subsección es distinto de cero. De lo contrario, devuelve NULL.</li> </ul> </li> </ul>

Tabla 184. 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"> <li>• Para subagentes (RPC):                             <ul style="list-style-type: none"> <li>- ABP</li> <li>- CATALOG</li> <li>- INTERNAL</li> <li>- REORG</li> <li>- RUNSTATS</li> <li>- WLM</li> </ul> </li> <li>• Para subagentes (OTHER):                             <ul style="list-style-type: none"> <li>- ABP</li> <li>- APP_RBSVPT</li> <li>- APP_RELSVPT</li> <li>- BACKUP</li> <li>- CLOSE</li> <li>- EXTERNAL_RBSVPT</li> <li>- EVMON</li> <li>- FORCE</li> <li>- FORCE_ALL</li> <li>- INTERNAL &lt;número&gt;</li> <li>- INTERRUPT</li> <li>- NOOP: si no hay solicitud</li> <li>- QP</li> <li>- REDISTRIBUTE</li> <li>- STMT_RBSVPT</li> <li>- STOP_USING</li> <li>- UPDATE_DBM_CFG</li> <li>- WLM</li> </ul> </li> </ul> <p>Si el tipo de solicitud es uno de los tipos internos, el valor se muestra como 'INTERNAL' seguido del valor real de la constante interna.</p>
NESTING_LEVEL	INTEGER	Representa el 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	Distingue una determinada invocación de una actividad de otras del mismo nivel de anidamiento.
ROUTINE_ID	INTEGER	Identificador exclusivo de la rutina. Nulo si no forma parte de una rutina.

## WLM\_GET\_SERVICE\_CLASS\_WORKLOAD\_OCCURRENCES - Lista de ocurrencias de carga de trabajo

Esta función devuelve la lista de todas las ocurrencias de carga de trabajo que se ejecutan en una determinada clase de servicio de una determinada partición. 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 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 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 ocurrencias 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),1,9) AS COORDPART,  
       SUBSTR(CHAR(APPLICATION_HANDLE),1,7) AS APPHNDL,  
       SUBSTR(WORKLOAD_NAME,1,18) 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,  
       WORKLOADDEF_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      SYSDEFAULTWORKLOAD 1
... 2      SYSDEFAULTWORKLOAD 2
... 1      SYSDEFAULTWORKLOAD 1
... 2      SYSDEFAULTWORKLOAD 2
... 1      SYSDEFAULTWORKLOAD 1
... 2      SYSDEFAULTWORKLOAD 2
... 1      SYSDEFAULTWORKLOAD 1
... 2      SYSDEFAULTWORKLOAD 2

```

### Nota sobre uso

Los parámetros tienen el efecto de sumarse (operación AND). Es decir, si se especifican registros conflictivos como un nombre de superclase SUP\_A y un nombre de subclase 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 ocurrencia 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 185. 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 partición de la que se ha recopilado este registro.
COORD_PARTITION_NUM	SMALLINT	Número de partición de la partición de coordinación de la ocurrencia de carga de trabajo dada.

Tabla 185. 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 ocurrencia de carga de trabajo. No identifica la ocurrencia 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. Como alternativa, se puede utilizar el descriptor de contexto de la aplicación en lugar del número de partición de base de datos de coordinación.
WORKLOAD_OCCURRENCE_STATE	VARCHAR(32)	Los valores posibles incluyen: <ul style="list-style-type: none"> <li>• DECOUPLED - La ocurrencia de la carga de trabajo no tiene ningún agente de coordinación asignado (caso de concentrador)</li> <li>• DISCONNECTPEND - La ocurrencia de la carga de trabajo se está desconectando de la base de datos</li> <li>• FORCED - Se ha forzado la ocurrencia de la carga de trabajo</li> <li>• QUEUED - Un umbral de colas de gestión de cargas de trabajo ha colocado en cola el agente de coordinación de la ocurrencia de la carga de trabajo</li> <li>• TRANSIENT - La ocurrencia de la carga de trabajo aún no se ha correlacionado con una superclase de servicio</li> <li>• UOWEXEC - La ocurrencia de la carga de trabajo está procesando una solicitud</li> <li>• UOWWAIT - La ocurrencia de la carga de trabajo está a la espera de una solicitud procedente del cliente</li> </ul>
UOW_ID	INTEGER	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 ocurrencia de carga de trabajo.
SYSTEM_AUTH_ID	VARCHAR(128)	ID de autorización del sistema bajo el que la ocurrencia de la carga de trabajo se ha inyectado en el sistema.
SESSION_AUTH_ID	VARCHAR(128)	ID de autorización de la sesión bajo el que la ocurrencia de la carga de trabajo se ha inyectado en el sistema.

Tabla 185. Información devuelta para WLM\_GET\_SERVICE\_CLASS\_WORKLOAD\_OCCURRENCES (continuación)

Nombre de columna	Tipo de datos	Descripción
APPLICATION_NAME	VARCHAR(128)	El nombre de la aplicación que ha creado esta ocurrencia de la carga de trabajo.
CLIENT_WRKSTNNAME	VARCHAR(255)	Valor actual del registro especial CLIENT_WRKSTNNAME correspondiente a esta ocurrencia de la carga de trabajo.
CLIENT_ACCTNG	VARCHAR(255)	Valor actual del registro especial CLIENT_ACCTNG correspondiente a esta ocurrencia de la carga de trabajo.
CLIENT_USER	VARCHAR(255)	Valor actual del registro especial CLIENT_USER correspondiente a esta ocurrencia de la carga de trabajo.
CLIENT_APPLNAME	VARCHAR(255)	Valor actual del registro especial CLIENT_APPLNAME correspondiente a esta ocurrencia 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 ocurrencia de la carga de trabajo. Esta estadística se actualiza cada vez que una actividad de esta ocurrencia 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 ocurrencia de la carga de trabajo. Esta estadística se actualiza cada vez que una actividad de esta ocurrencia 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 ocurrencia 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 ocurrencia 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 ocurrencia 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

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 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 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, el estado global del sistema se puede supervisar de forma regular 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 el hecho de pasar un valor nulo para un argumento no restringe el resultado en dicho argumento, excepto en el caso del último argumento, númparticiónbd, para el que el valor -2 significa que se devolverán datos procedentes de todas las particiones de base de datos). La siguiente sentencia devuelve estadísticas de clase de servicio como duración media de la actividad y 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 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(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 ofrecerá al administrador una buena vista a nivel alto de la "carga" de cada partición para una determinada base de datos, comprobando los tiempos promedios de ejecución y los números de actividades. Cualquier variación significativa de los indicadores de nivel alto que devuelven estas funciones de tabla puede indicar un cambio en la carga del sistema.

## Notas sobre 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 (operación AND). Es decir, si se especifican registros conflictivos como un nombre de superclase SUPA y un nombre de subclase SUBB de modo que SUBB no es una subclase de SUPA, no se devuelve ninguna fila.

### Información devuelta

Tabla 186. 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 restablecieron por última vez las estadísticas. Se pueden producir cuatro sucesos que causarán un restablecimiento de estadísticas, que actualizará esta indicación horaria:</p> <ul style="list-style-type: none"> <li>• Se llama al procedimiento WLM_COLLECT_STATS.</li> <li>• El proceso periódico de recopilación y restablecimiento controlado por el parámetro de configuración WLM_COLLECT_INT genera una recopilación y un restablecimiento.</li> <li>• Se reactiva la base de datos.</li> <li>• La subclase de servicio para la que se notifican estadísticas se ha modificado y el cambio se ha confirmado.</li> </ul> <p>La indicación horaria de LAST_RESET está en la hora local.</p>
COORD_ACT_COMPLETED_TOTAL	BIGINT	Número total de actividades de coordinación que han enviado los usuarios desde el último restablecimiento que se han completado satisfactoriamente.
COORD_ACT_ABORTED_TOTAL	BIGINT	Número total de actividades de coordinación que han enviado los usuarios desde el último restablecimiento que se han completado con errores.
COORD_ACT_REJECTED_TOTAL	BIGINT	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.
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 subclase de servicio.

Tabla 186. Información devuelta para WLM\_GET\_SERVICE\_SUBCLASS\_STATS (continuación)

Nombre de columna	Tipo de datos	Descripción
COORD_ACT_LIFETIME_TOP	BIGINT	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.
COORD_ACT_LIFETIME_AVG	DOUBLE	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.
COORD_ACT_LIFETIME_STDDEV	DOUBLE	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.
COORD_ACT_EXEC_TIME_AVG	DOUBLE	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.
COORD_ACT_EXEC_TIME_STDDEV	DOUBLE	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.

Tabla 186. Información devuelta para WLM\_GET\_SERVICE\_SUBCLASS\_STATS (continuación)

Nombre de columna	Tipo de datos	Descripción
COORD_ACT_QUEUE_TIME_AVG	DOUBLE	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.
COORD_ACT_QUEUE_TIME_STDDEV	DOUBLE	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.
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.
NUM_REQUESTS_TOTAL	BIGINT	Número de solicitudes que se han empezado a ejecutar en esta subclase de servicio desde el último restablecimiento. 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.
REQUEST_EXEC_TIME_AVG	DOUBLE	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.

Tabla 186. Información devuelta para WLM\_GET\_SERVICE\_SUBCLASS\_STATS (continuación)

Nombre de columna	Tipo de datos	Descripción
REQUEST_EXEC_TIME_STDDEV	DOUBLE	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.
REQUEST_EXEC_TIME_TOTAL	BIGINT	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.

## WLM\_GET\_SERVICE\_SUPERCLASS\_STATS - Devolver estadísticas de superclases de servicio

Esta función devuelve estadísticas básicas de 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 un nombre de superclase de servicio válido que está en la 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.

*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\_SUPERCLASS\_STATS.

## Ejemplo

Para ver todas 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
```

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

```
SERVICE_SUPERCLASS_NAME  DBPARTITIONNUM ...  
-----  
SYSDEFAULTSYSTEMCLASS    0 ...  
SYSDEFAULTMAINTENANCECLASS 0 ...  
SYSDEFAULTUSERCLASS       0 ...
```

Datos de salida de esta consulta (continuación).

```
... 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. No se realiza ninguna agregación entre superclases de servicio (de una partición) ni entre particiones (correspondientes a una o varias superclases de servicio). Sin embargo, se puede lograr la agregación mediante consultas SQL, tal como se muestra en el ejemplo anterior.

## Información devuelta

Tabla 187. 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 187. Información devuelta para WLM\_GET\_SERVICE\_SUPERCLASS\_STATS (continuación)

Nombre de columna	Tipo de datos	Descripción
LAST_RESET	TIMESTAMP	<p>Hora en que se restablecieron por última vez las estadísticas. Se pueden producir cuatro sucesos que causarán un restablecimiento de estadísticas, que actualizará esta indicación horaria:</p> <ul style="list-style-type: none"> <li>• Se llama al procedimiento WLM_COLLECT_STATS.</li> <li>• El proceso periódico de recopilación y restablecimiento controlado por el parámetro de configuración WLM_COLLECT_INT genera una recopilación y un restablecimiento.</li> <li>• Se reactiva la base de datos.</li> <li>• La superclase de servicio para la que se notifican estadísticas se ha modificado y el cambio se ha confirmado.</li> </ul> <p>La indicación horaria de LAST_RESET está en la hora local.</p>
CONCURRENT_CONNECTION_TOP	INTEGER	Número más alto de conexiones simultáneas de coordinación que se ha alcanzado en esta clase desde el último restablecimiento.

## WLM\_GET\_WORK\_ACTION\_SET\_STATS - Devolver estadísticas de conjunto de acciones de trabajo

Esta función 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 específico para el que hay que devolver estadísticas. Si el argumento es nulo o una serie vacía, se devuelven estadísticas correspondientes a todas 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 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\_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, hay 8 actividades actualmente en ejecución (o en cola) en ReadClass, 4 actividades actualmente en ejecución (o en cola) en WriteClass, 2 actividades actualmente en ejecución (o en cola) en LoadClass y 3 actividades actualmente en ejecución (o en cola) que no se han asignado a ninguna clase de trabajo. Puesto que no hay ninguna acción de trabajo asociada a la clase de trabajo WriteClass, las 4 actividades a las que se aplican se cuentan en la clase artificial "\*" junto con las 3 actividades que no se han asignado a ninguna clase de trabajo.

```
SELECT 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 TOTAL_WLO_ACTS  
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
```

A continuación se muestra un ejemplo de salida de esta consulta.

WORK_ACTION_SET_NAME	PART	WORK_CLASS_NAME	LAST_RESET	TOTAL_WLO_ACTS
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 188. Información devuelta para WLM\_GET\_WORK\_ACTION\_SET\_STATS

Nombre de columna	Tipo de datos	Descripción
WORK_ACTION_SET_NAME	VARCHAR(128)	El nombre del conjunto de acciones de trabajo.
DBPARTITIONNUM	SMALLINT	Número de partición de la que se ha recopilado este registro.

Tabla 188. Información devuelta para WLM\_GET\_WORK\_ACTION\_SET\_STATS (continuación)

Nombre de columna	Tipo de datos	Descripción
LAST_RESET	TIMESTAMP	<p>Hora en que se restablecieron por última vez las estadísticas. Se pueden producir cuatro sucesos que causarán un restablecimiento de estadísticas, que actualizará esta indicación horaria:</p> <ul style="list-style-type: none"> <li>• Se llama al procedimiento WLM_COLLECT_STATS.</li> <li>• El proceso periódico de recopilación y restablecimiento controlado por el parámetro de configuración WLM_COLLECT_INT genera una recopilación y un restablecimiento.</li> <li>• Se reactiva la base de datos.</li> <li>• El conjunto de acciones de trabajo para el que se notifican estadísticas se ha modificado y el cambio se ha confirmado.</li> </ul> <p>La indicación horaria de LAST_RESET está en la hora local.</p>
WORK_CLASS_NAME	VARCHAR(128)	<p>El nombre de la clase de trabajo relacionada con el conjunto de acciones de trabajo especificado. Debe haber una acción de trabajo asociada a esta clase de trabajo para que aparezca en esta tabla. "*" 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.</p>
ACT_TOTAL	BIGINT	<p>El número de actividades de cualquier nivel de anidamiento que se han asignado a la clase de trabajo especificada por WORK_CLASS_NAME.</p>

## WLM\_GET\_WORKLOAD\_OCCURRENCE\_ACTIVITIES - Devolver una lista de actividades

Esta función 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.

### 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 devuelve una lista de actividades. Si el argumento es nulo, se recuperan datos correspondientes a todas las aplicaciones 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 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\_WORKLOAD\_OCCURRENCE\_ACTIVITIES.

## Ejemplo

Una vez identificado un 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 es 1, determinado mediante el mandato de lista de aplicaciones:

```
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
```

A continuación se muestra un ejemplo de salida de esta consulta.

COORD	PART	UOWID	ACTID	PARUOWID	PARACTID	ACTTYPE	NESTING
0	0	2	3	-	-	CALL	0
0	0	2	5	2	3	READ_DML	1

## Nota sobre uso

El valor QUEDED 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 y buscar si el event\_object del agente de la partición de catálogo tiene el valor WLM\_QUEUE.

## Información devuelta

Tabla 189. 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 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.

Tabla 189. Información devuelta por WLM\_GET\_WORKLOAD\_OCCURRENCE\_ACTIVITIES (continuación)

Nombre de columna	Tipo de datos	Descripción
DBPARTITIONNUM	SMALLINT	Número de la 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. Es una hora local. Es campo puede ser nulo si la actividad ha entrado en el sistema pero está en cola y aún no se ha empezado a ejecutar.
UOW_ID	INTEGER	Identificador exclusivo de unidad de trabajo dentro de una aplicación. Se refiere a 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	Identificador exclusivo de unidad de trabajo dentro de una aplicación. Se refiere a la unidad de trabajo original en la que se ha iniciado la actividad padre. Devuelve nulo si la actividad no tiene actividad padre o si está en una partición remota.
PARENT_ACTIVITY_ID	INTEGER	Identificador exclusiva de actividad dentro de una unidad de trabajo correspondiente a la actividad padre de la actividad cuyo ID es ACTIVITY_ID. Devuelve nulo si la actividad no tiene actividad padre o si está en una partición remota.

Tabla 189. Información devuelta por WLM\_GET\_WORKLOAD\_OCCURRENCE\_ACTIVITIES (continuación)

Nombre de columna	Tipo de datos	Descripción
ACTIVITY_STATE	VARCHAR(32)	<p>Los valores posibles son:</p> <ul style="list-style-type: none"> <li>• CANCEL_PENDING - La actividad se ha cancelado cuando no había ningún agente trabajando de forma activa en una solicitud correspondiente a la actividad. La siguiente vez que se envíe una solicitud como parte de la actividad, la actividad se cancelará y el usuario que haya enviado la actividad recibirá un error SQL4725N.</li> <li>• EXECUTING - Los agentes están trabajando de forma activa en una solicitud correspondiente a la actividad.</li> <li>• IDLE - No hay ningún agente procesando de forma activa una solicitud correspondiente a la actividad.</li> <li>• INITIALIZING - 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 o no que se ejecute la actividad.</li> <li>• QP_CANCEL_PENDING - Igual que el estado CANCEL_PENDING, pero la actividad ha sido cancelada por Query Patroller en lugar de mediante el procedimiento WLM_CANCEL_ACTIVITY.</li> <li>• QP_QUEUED - Query Patroller ha colocado la actividad en cola</li> <li>• QUEUED - 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</li> <li>• TERMINATING - Ha finalizado la ejecución de la actividad y se está eliminando del sistema.</li> </ul>
ACTIVITY_TYPE	VARCHAR(32)	<p>Los valores posibles son:</p> <ul style="list-style-type: none"> <li>• CALL</li> <li>• DDL</li> <li>• LOAD</li> <li>• OTHER</li> <li>• READ_DML</li> <li>• WRITE_DML</li> </ul> <p>Consulte “Tipos de trabajo de clase de trabajo y sentencias SQL” in <i>Workload Manager Guide and Reference</i> para ver una descripción de los distintos tipos de sentencias SQL asociados a cada tipo de actividad.</p>

Tabla 189. Información devuelta por `WLM_GET_WORKLOAD_OCCURRENCE_ACTIVITIES` (continuación)

Nombre de columna	Tipo de datos	Descripción
NESTING_LEVEL	INTEGER	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.
INVOCATION_ID	INTEGER	Distingue una determinada invocación de esta actividad de otras del mismo nivel de anidamiento.
ROUTINE_ID	INTEGER	Identificador exclusivo de la rutina.
UTILITY_ID	INTEGER	Si la actividad es un programa de utilidad, es el ID del programa de utilidad. De lo contrario, este campo tiene el valor nulo.
SERVICE_CLASS_ID	INTEGER	Identificador exclusivo de la clase de servicio a la que pertenece esta actividad.
DATABASE_WORK_ACTION_SET_ID	INTEGER	Si esta actividad se ha clasificado en una clase de trabajo de ámbito de base de datos, esta columna contiene el ID del conjunto de la clase de trabajo del que esta clase de trabajo es miembro. Esta columna contiene un valor nulo si la actividad no se ha clasificado en una clase de trabajo de ámbito de base de datos.
DATABASE_WORK_CLASS_ID	INTEGER	Si esta actividad se ha clasificado en una clase de trabajo de ámbito de base de datos, esta columna contiene el ID de la clase de trabajo. Esta columna contiene un valor nulo si la actividad no se ha clasificado en una clase de trabajo de ámbito de base de datos.
SERVICE_CLASS_WORK_ACTION_SET_ID	INTEGER	Si esta actividad se ha clasificado en una clase de trabajo de ámbito de clase de servicio, esta columna contiene el ID del conjunto de la acción de trabajo asociado al conjunto de la clase de trabajo al que pertenece esta clase de trabajo. Esta columna contiene un valor nulo si la actividad no se ha clasificado en una clase de trabajo de ámbito de clase de servicio.
SERVICE_CLASS_WORK_CLASS_ID	INTEGER	Si esta actividad se ha clasificado en una clase de trabajo de ámbito de clase de servicio, esta columna contiene el ID de la clase de trabajo asociado a esta actividad. Esta columna contiene un valor nulo si la actividad no se ha clasificado en una clase de trabajo de ámbito de clase de servicio.

## WLM\_GET\_WORKLOAD\_STATS - Devolver estadísticas de carga de trabajo

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*

Un 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 ocurrencias simultáneas de la carga de trabajo MYUSERWORKLOAD ha sido 2 y que el número más alto de actividades simultáneas en cualquiera de estas ocurrencias 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 SQL.

## Información devuelta

Tabla 190. Información devuelta por WLM\_GET\_WORKLOAD\_STATS

Nombre de columna	Tipo de datos	Descripción
WORKLOAD_NAME	BIGINT	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
LAST_RESET	TIMESTAMP	<p>Hora en que se restablecieron por última vez las estadísticas. Se pueden producir cuatro sucesos que causarán un restablecimiento de estadísticas, que actualizará esta indicación horaria:</p> <ul style="list-style-type: none"> <li>• Se llama al procedimiento WLM_COLLECT_STATS.</li> <li>• El proceso periódico de recopilación y restablecimiento controlado por el parámetro de configuración WLM_COLLECT_INT genera una recopilación y un restablecimiento.</li> <li>• Se reactiva la base de datos.</li> <li>• La carga de trabajo para la que se notifican estadísticas se ha modificado y el cambio se ha confirmado.</li> </ul> <p>La indicación horaria de LAST_RESET está en la hora local.</p>
CONCURRENT_WLO_TOP	INTEGER	Número más alto de ocurrencias 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 (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 ocurrencia de esta carga de trabajo desde el último restablecimiento. Lo actualiza cada ocurrencia 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 ocurrencia de esta carga de trabajo que se han completado desde el último restablecimiento. Lo actualiza cada ocurrencia 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 ocurrencia de esta carga de trabajo que han terminado anormalmente antes de finalizar desde el último restablecimiento. Lo actualiza cada ocurrencia de carga de trabajo al final de su unidad de trabajo.



Tabla 190. Información devuelta por WLM\_GET\_WORKLOAD\_STATS (continuación)

Nombre de columna	Tipo de datos	Descripción
COORD_ACT_REJECTED_TOTAL	BIGINT	Número total de actividades de coordinación a cualquier nivel de anidamiento asignadas a cualquier ocurrencia de esta carga de trabajo que se han rechazado antes de su ejecución desde el último restablecimiento. Lo actualiza cada ocurrencia 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.
WLO_COMPLETED_TOTAL	BIGINT	El número de ocurrencias de carga de trabajo que hay que completar desde el último restablecimiento.

## 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 *establadestino*. Si se especifica nulo para este parámetro, los objetos nuevos se crearán utilizando los mismos espacios de tabla que utilizan los objetos de origen.

#### *establadestino*

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 *establaorigen* de espacios de tabla. El espacio de tabla número n de la lista *establaorigen* se correlacionará con el espacio de tabla número n de la lista *establadestino* 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 191. 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.
SENTENCIA	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 sobre 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
  - tablas con tipo
  - 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 - Eliminar 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 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 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 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 a ADMIN\_DROP\_SCHEMA; es decir, la tabla se debe eliminar para poder reclamar el espacio que consume en SYSTOOLSPACE.

Tabla 192. 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.

Tabla 192. Formato de la tabla de errores de ADMIN\_DROP\_SCHEMA (continuación)

Nombre de columna	Tipo de datos	Descripción
SQLSTATE	CHAR(5)	SQLSTATE del error.
ERROR_TIMESTAMP	TIMESTAMP	Hora a la que ha fallado el mandato de eliminación.
SENTENCIA	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 de parám. de salida

-----

Nombre del parámetro: ERRORTABSCHEMA

Valor del parámetro: ERRORSCHEMA <-- error!

Nombre del parámetro: ERRORTAB

Valor del parámetro: ERRORTABLE <-- error!

Estado de retorno = 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 sobre uso

- Si hay objetos en otro esquema que dependan 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
  - tablas con tipo
  - 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 tabla 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.

## 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. 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 desharrán los cambios hechos mediante la operación de modificación de tabla, en el caso de que una operación de



retrotracción 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 sobre uso**

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.

## RAPPLICATION\_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** – Recuperación de 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 193 en la página 698): **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 194 en la página 698.

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.COMPILE_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.COMPILE_ENV(:hv1)) AS t
WHERE NAME = 'SCHEMA'
```

## Información devuelta

Tabla 193. 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 194 para obtener más información.
VALUE	VARCHAR(1024)	Valor del elemento.

Tabla 194. 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 CURRENT ISOLATION o de la opción de vinculación ISOLATION 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 CURRENT QUERY OPTIMIZATION o de la opción de vinculación QUERYOPT 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 <i>min_dec_div_3</i> .
DEGREE	Es el nivel solicitado de intra-paralelismo que se pasa al compilador de SQL. El valor se obtiene a partir del registro especial CURRENT DEGREE o de la opción de vinculación DEGREE 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 LANGLVL del paquete actual. Los valores posibles son 'DB2' o 'SQL92'.
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 CURRENT REFRESH AGE o de la opción de vinculación REFRESHAGE 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 CURRENT SCHEMA o de la opción de vinculación QUALIFIER del paquete actual.

Tabla 194. Elementos de un entorno de compilación devueltos por la función de tabla *COMPILATION\_ENV* (continuación)

Nombre de elemento	Descripción
PATH	Es la vía de acceso de función que se pasa al compilador de SQL. El valor se obtiene a partir del registro especial CURRENT PATH o de la opción de vinculación FUNC_PATH 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 CURRENT DEFAULT TRANSFORM GROUP o de la opción de vinculación de paquetes TRANSFORMGROUP.
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 horaria 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 horaria es la indicación horaria actual o la indicación horaria 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 contactos

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

Privilegio SELECT o CONTROL sobre la vista administrativa CONTACTGROUPS y privilegio EXECUTE sobre la función de tabla ADMIN\_GET\_CONTACTGROUPS.

### 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 195. 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

Privilegio SELECT o CONTROL sobre la vista administrativa CONTACTS y privilegio EXECUTE sobre la función de tabla ADMIN\_GET\_CONTACTS.

### 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  
user5 EMAIL user2@ca.ibm.com

128 DBA on Page  
- DBA Extraordinaire

4 registro(s) seleccionado(s).

## Nota sobre uso

El servidor DAS debe haberse creado y estar en ejecución.

## Información devuelta

Tabla 196. 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"><li>• 'EMAIL'</li><li>• 'PAGE'</li></ul>
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 archivos históricos

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

Privilegio SELECT o CONTROL sobre la vista administrativa DB\_HISTORY y privilegio EXECUTE sobre la función de tabla ADMIN\_LIST\_HIST.

### 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 197. 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 hora que marca el inicio de un suceso anotado cronológicamente.
SEQNUM	SMALLINT	Número de secuencia.
END_TIME	VARCHAR(14)	Indicación de 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.
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 198 en la página 705 para ver los valores posibles.



Tabla 197. Información devuelta por la vista administrativa DB\_HISTORY (continuación)

Nombre de columna	Tipo de datos	Descripción
OPERATIONTYPE	CHAR(1)	Identificador de acción de una operación. Consulte la Tabla 198 en la página 705 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 tabla 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.
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).

Tabla 197. Información devuelta por la vista administrativa DB\_HISTORY (continuación)

Nombre de columna	Tipo de datos	Descripción
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 dígitos 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 de SQL).
SQLERRD2	INTEGER	Consulte SQLCA (área de comunicaciones de SQL).
SQLERRD3	INTEGER	Consulte SQLCA (área de comunicaciones de SQL).
SQLERRD4	INTEGER	Consulte SQLCA (área de comunicaciones de SQL).
SQLERRD5	INTEGER	Consulte SQLCA (área de comunicaciones de SQL).
SQLERRD6	INTEGER	Consulte SQLCA (área de comunicaciones de SQL).
SQLWARN	VARCHAR(11)	Conjunto de indicadores de aviso, cada uno de los cuales contiene un blanco o 'W'. Consulte SQLCA (área de comunicaciones de 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 198. Valores de OPERATION y OPERATIONTYPE

Valor de la operación	Descripción del valor de la operación	Tipo de operación
A	Añadir espacio de tabla	Ninguno
B	Hacer copia de seguridad	Los tipos de operación son: <ul style="list-style-type: none"> <li>• D = fuera de línea delta</li> <li>• E = en línea delta</li> <li>• F = fuera de línea</li> <li>• I = fuera de línea incremental</li> <li>• N = en línea</li> <li>• O = en línea incremental</li> </ul>
C	Cargar copia	Ninguno
D	Tabla eliminada	Ninguno
F	Realizar recuperación en avance	Los tipos de operación son: <ul style="list-style-type: none"> <li>• E = fin de anotaciones cronológicas</li> <li>• P = punto en el tiempo</li> </ul>
G	Reorganizar tabla	Los tipos de operación son: <ul style="list-style-type: none"> <li>• F = fuera de línea</li> <li>• N = en línea</li> </ul>
L	Cargar	Los tipos de operación son: <ul style="list-style-type: none"> <li>• I = insertar</li> <li>• R = sustituir</li> </ul>
N	Cambiar nombre de espacio de tabla	Ninguno
O	Eliminar espacio de tabla	Ninguno
Q	Inmovilizar	Los tipos de operación son: <ul style="list-style-type: none"> <li>• S = inmovilizar compartición</li> <li>• U = inmovilizar actualización</li> <li>• X = inmovilizar exclusividad</li> <li>• Z = inmovilizar restablecimiento</li> </ul>
R	Restaurar	Los tipos de operación son: <ul style="list-style-type: none"> <li>• F = fuera de línea</li> <li>• I = fuera de línea incremental</li> <li>• N = en línea</li> <li>• O = en línea incremental</li> <li>• R = volver a crear</li> </ul>
T	Modificar espacio de tabla	Los tipos de operación son: <ul style="list-style-type: none"> <li>• C = añadir contenedores</li> <li>• R = volver a equilibrar</li> </ul>

Tabla 198. Valores de OPERATION y OPERATIONTYPE (continuación)

Valor de la operación	Descripción del valor de la operación	Tipo de operación
U	Descargar	Ninguno
X	Archivar anotaciones cronológicas	Los tipos de operación son: <ul style="list-style-type: none"> <li>• F = vía de acceso de archivado fallido</li> <li>• M = vía de acceso de anotaciones cronológicas de duplicación</li> <li>• N = truncamiento forzado mediante mandato ARCHIVE LOG</li> <li>• P = vía de acceso de anotaciones cronológicas primarias</li> <li>• 1 = primer método de archivado de anotaciones cronológicas</li> <li>• 2 = segundo método de archivado de anotaciones cronológicas</li> </ul>

## 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

Privilegio SELECT o CONTROL sobre la vista administrativa DBPATHS y privilegio EXECUTE sobre la función de tabla ADMIN\_LIST\_DB\_PATHS.

### 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\SQLGDIR\
... 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 199. Información devuelta por la vista administrativa DBPATHS

Nombre de columna	Tipo de datos	Descripción
DBPARTITIONNUM	SMALLINT	Número de partición de base de datos.
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 200 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 200. Valores de la columna TYPE

Valor de tipo	Descripción
TBSP_DEVICE	Dispositivo en bruto correspondiente a un espacio de tabla del espacio gestionado por la base de datos (DMS)
TBSP_CONTAINER	Contenedor de archivos correspondiente a un espacio de tabla DMS.

Tabla 200. Valores de la columna TYPE (continuación)

Valor de tipo	Descripción
TBSP_DIRECTORY	Directorio correspondiente a un espacio de tabla 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.

Tabla 200. Valores de la columna TYPE (continuación)

Valor de tipo	Descripción
	<ul style="list-style-type: none"> <li>• Para espacios de tabla 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.</li> <li>• Los contenedores de espacios de tabla gestionados por el almacenamiento automático no se devuelven de forma individual; quedan reflejados en la columna de vía de acceso del almacenamiento automático.</li> <li>• Las vías de acceso del almacenamiento automático se devuelven una vez por cada partición de base de datos.</li> <li>• 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.</li> <li>• Si la salida desde <code>SELECT * FROM SYSIBMADM.DBPATHS</code> se está utilizando para crear un archivo de configuración <code>db2relocatedb</code> (un archivo que contiene la información necesaria sobre la configuración para reubicar una base de datos), la salida <code>DBPATH</code> debe modificarse como corresponda antes de que pueda utilizarse en el archivo de configuración.</li> </ul> <p>Por ejemplo, la siguiente salida <code>DBPATH</code>:</p> <pre>/storage/svtdbm3/svtdbm3/NODE0000/SQL00001/</pre> <p>se puede utilizar para especificar el parámetro <code>DB_PATH</code> en un archivo de configuración <code>db2relocatedb</code> del modo siguiente:</p> <pre>DB_PATH=/storage/svtdbm3,/storage_copy2/svtdbm3</pre> <ul style="list-style-type: none"> <li>• La vía de acceso <code>LOCAL_DB_DIRECTORY</code> puede contener información correspondiente a varias bases de datos. Debido a que <code>sqlbdir</code> 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.</li> <li>• 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.</li> </ul>

## 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 consulta determinada. El tipo de datos del resultado es CLOB(50M).

### Sintaxis

►►—EXPLAIN\_FORMAT\_STATS—(—*instantánea*—)—————►►

El esquema es SYSPROC.

### Parámetros de la función escalar

#### *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 = ''
      EXPLAIN_LEVEL = '0' AND
      STMTNO = 1 AND
      SECTNO = 201
```

A continuación presentamos una salida de ejemplo de esta función:

Contexto espacio de tabla:

```
-----
Nombre:                                USERSPACE1
Actividad general:                      7.500000
Velocidad de transferencia:             0.060000
Tamaño de captación previa:            32
Tamaño de extensión:                   32
Tipo:                                    Gestionado por la base de datos
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 intervalo:	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      Sincrónicas
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
Esquema:DB2USER2
Norma exclusiva:                               Índice duplicado
Utilizado en el operador:                       No
Pares de captación de páginas:                 No disponible
Número de columnas:                             1
Páginas hojas de índice:                       1
Niveles de árbol de índice:                   1
Cardinalidad de primera clave de índice:       2
Cardinalidad de clave completa de índice:     2
Proporción de clúster de índice:               100
Factor de clúster de índice:                   -1.000000
Fecha y hora de creación:                       2006-06-16-11.46.53.857520
Última actualización de estadísticas:           2006-06-26-12.23.45.157028
Páginas secuenciales de índice:                0
Cardinalidad de 2 primeras claves de índice:  -1
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 intervalo: 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:                        2
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. partición de datos: No disponible

```

## EXPLAIN\_GET\_MSGS

```

▶▶ EXPLAIN_GET_MSGS(—peticionario-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.

La función de tabla EXPLAIN\_GET\_MSGS consulta las tablas Explain EXPLAIN\_DIAGNOSTIC y EXPLAIN\_DIAGNOSTIC\_DATA, y devuelve los mensajes con formato.

Cualquiera de los siguientes argumentos de entrada puede ser nulo. Si un argumento es nulo, no se utiliza para limitar la consulta.

### *peticionario-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.

La función devuelve una tabla tal como se muestra más abajo.

*Tabla 201. 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.
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.

Tabla 201. Información devuelta por la función de tabla EXPLAIN\_GET\_MSGS (continuación)

Nombre de columna	Tipo de datos	Descripción
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.

### Ejemplo

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 TIME - 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".

```

## RGET\_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 de parám. de salida
-----
Nombre del parámetro : SNAPSHOTTIMESTAMP
Valor del parámetro  : 2004-02-29-18.31.55.178000

Nombre del parámetro : DATABASESIZE
Valor del parámetro  : 22302720

Nombre del parámetro : DATABASECAPACITY
Valor del parámetro  : 4684793856

Estado de retorno = 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 de parám. de salida
-----
Nombre del parámetro : SNAPSHOTTIMESTAMP
Valor del parámetro  : 2004-02-29-18.33.34.561000

Nombre del parámetro : DATABASESIZE
Valor del parámetro  : 22302720

Nombre del parámetro : DATABASECAPACITY
Valor del parámetro  : 4684859392

Estado de retorno = 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 de parám. de salida
-----
Nombre del parámetro : SNAPSHOTTIMESTAMP
Valor del parámetro  : 2004-02-29-18.33.34.561000

Nombre del parámetro : DATABASESIZE
Valor del parámetro  : 22302720

Nombre del parámetro : DATABASECAPACITY
Valor del parámetro  : 4684859392

Estado de retorno = 0
```

## Notas sobre 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

Privilegio SELECT o CONTROL sobre la vista administrativa NOTIFICATIONLIST y privilegio EXECUTE sobre la función de tabla HEALTH\_GET\_NOTIFICATION\_LIST.

### Ejemplo

Recuperar todos los contactos que recibirán alertas de salud.

```
SELECT * FROM SYSIBMADM.NOTIFICATIONLIST
```

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

```
NAME                TYPE
-----
group3              GROUP
user4               CONTACT
group3              GROUP
```

3 registros(s) seleccionado(s).

### Información devuelta

Tabla 202. 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"> <li>'CONTACT'</li> <li>'GROUP'</li> </ul>

## 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 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. Esto actualmente significa db2diag.log, registro de notificaciones de administración y registros de sucesos rotativos.
- 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 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 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 203. 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"> <li>• ALL: Devuelve registros de todos los recursos</li> <li>• MAIN: Devuelve registros de los registros cronológicos de diagnósticos generales de DB2. Esto actualmente significa db2diag.log, registro de notificaciones de administración y registros de sucesos rotativos</li> <li>• OPTSTATS: Devolver registros relacionados con estadísticas del optimizador</li> </ul>

Tabla 203. Información devuelta por la función de tabla PD\_GET\_DIAG\_HIST (continuación)

Nombre de columna	Tipo de datos	Descripción
RECTYPE	VARCHAR(3)	El tipo de registro. Los valores posibles son: <ul style="list-style-type: none"> <li>• 'DI': Registro de diagnóstico interno</li> <li>• 'DX': Registro de diagnóstico externo</li> <li>• 'EI': Registro de sucesos internos</li> <li>• 'EX': Registro de sucesos externos</li> </ul>
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"> <li>• 'C': Crítico</li> <li>• 'E': Error</li> <li>• 'I': Informativo</li> <li>• 'S': Grave</li> <li>• 'W': Aviso</li> </ul>
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"> <li>• 'CRITICAL'</li> <li>• 'IMMEDIATE'</li> <li>• 'NONE'</li> <li>• 'POTENTIAL'</li> <li>• 'UNLIKELY'</li> </ul>
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.
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'

Tabla 203. Información devuelta por la función de tabla PD\_GET\_DIAG\_HIST (continuación)

Nombre de columna	Tipo de datos	Descripción
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' y 'DB2 UDB'.
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 203. 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"> <li>• 'APM'</li> <li>• 'CATALOG CACHE ENTRY'</li> <li>• 'CFG'</li> <li>• 'CLI'</li> <li>• 'CLP'</li> <li>• 'CONTAINER'</li> <li>• 'COUNTER'</li> <li>• 'DAS'</li> <li>• 'DB2AGENT'</li> <li>• 'DB PART MAP ID'</li> <li>• 'DB PART NUM'</li> <li>• 'DBA'</li> <li>• 'DBM'</li> <li>• 'DMS'</li> <li>• 'DPS'</li> <li>• 'EDU'</li> <li>• 'EVALUATION'</li> <li>• 'EXTENDER'</li> <li>• 'FCM'</li> <li>• 'HISTOGRAM TEMPLATE'</li> <li>• 'INDEX STATS'</li> <li>• 'INITIAL SAMPLING'</li> <li>• 'REDIST DB PART GROUP'</li> <li>• 'REDIST TABLE'</li> <li>• 'RDS'</li> <li>• 'SAMPLING TEST'</li> <li>• 'SERVICE CLASS'</li> <li>• 'STATS'</li> <li>• 'STATS DAEMON'</li> <li>• 'TABLE'</li> <li>• 'TABLE STATS'</li> <li>• 'TABLE AND INDEX STATS'</li> <li>• 'THRESHOLD'</li> <li>• 'UDF'</li> <li>• 'WORK ACTION SET'</li> <li>• 'WORK CLASS SET'</li> <li>• 'WORKLOAD'</li> </ul>
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 203. 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"> <li>• 'ACCEPT'</li> <li>• 'ADD'</li> <li>• 'ALTER'</li> <li>• 'ASSOCIATE'</li> <li>• 'AVAILABLE'</li> <li>• 'BRINGDOWN'</li> <li>• 'CHANGE'</li> <li>• 'CHANGECFG'</li> <li>• 'CLOSE'</li> <li>• 'COLLECT'</li> <li>• 'CONNECT'</li> <li>• 'CREATE'</li> <li>• 'DEPENDENCY'</li> <li>• 'DESTROY'</li> <li>• 'DISASSOCIATE'</li> <li>• 'DISCONNECT'</li> <li>• 'DISPATCH'</li> <li>• 'DROP'</li> <li>• 'FINI'</li> <li>• 'FREE'</li> <li>• 'GET'</li> <li>• 'INIT'</li> <li>• 'INTERRUPT'</li> <li>• 'OPEN','READ'</li> <li>• 'RECV'</li> <li>• 'REPLY'</li> <li>• 'REPORT'</li> <li>• 'REQUEST'</li> <li>• 'RESET'</li> <li>• 'SEND'</li> <li>• 'START'</li> <li>• 'STARTUP'</li> <li>• 'STOP'</li> <li>• 'SWITCH'</li> <li>• 'TERMINATE'</li> <li>• 'TRANSFER'</li> <li>• 'WAIT'</li> <li>• 'WORK'</li> <li>• 'WRITE'</li> </ul>
EVENTDESC	VARCHAR(256)	Descripción breve de los campos clave de este suceso.

Tabla 203. 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"> <li>• 'AT'</li> <li>• 'BY'</li> <li>• 'CONTEXT'</li> <li>• 'DUE TO'</li> <li>• 'FOR'</li> <li>• 'FROM'</li> <li>• 'ON'</li> <li>• 'TO'</li> </ul> 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 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"> <li>• Asíncrono</li> <li>• FABRICATE</li> <li>• FABRICATE PARTIAL</li> <li>• SYNCHRONOUS</li> <li>• SYNCHRONOUS SAMPLED</li> <li>• USER</li> </ul>
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"> <li>• Conflicto</li> <li>• Error</li> <li>• Objeto no disponible</li> <li>• Error de RUNSTATS</li> <li>• Tiempo de espera excedido</li> </ul>
EVENTSTATE	VARCHAR(255)	Estado del objeto o acción como resultado del suceso. También puede contener un porcentaje que indica la progresión del suceso.



Tabla 203. 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"> <li>• 'ASYN'</li> <li>• 'AUTO'</li> <li>• 'CACHED'</li> <li>• 'DIRECT'</li> <li>• 'EXTERNAL'</li> <li>• 'INDIRECT'</li> <li>• 'INTERNAL'</li> <li>• 'LOGICAL'</li> <li>• 'PERMANENT'</li> <li>• 'PHYSICAL'</li> <li>• 'SYNC'</li> <li>• 'TEMPORARY'</li> </ul>
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.
DUMPFIL	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.

### PDLOGMSGS\_LAST24HOURS, vista administrativa

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 204 en la página 733 para ver una lista completa de la información que se puede devolver.

## Autorización

Privilegio SELECT o CONTROL sobre la vista administrativa PDLOGMSGs\_LAST24HOURS y privilegio EXECUTE sobre la función de tabla PD\_GET\_LOG\_MSGS.

## Ejemplo

Obtener todos los mensajes críticos de la anotación cronológica registrados durante las últimas 24 horas, ordenados por el más reciente.

```
SELECT * FROM SYSIBMADM.PDLOGMSGs_LAST24HOURS
WHERE MSGSEVERITY = 'C' ORDER BY TIMESTAMP DESC
```

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

TIMESTAMP	TIMEZONE	INSTANCENAME	...
2005-11-23-21.56.41.240066	-300	svtdbm4	...
			...
			...
			...
			...
			...
			...
			...
2005-11-23-21.56.39.150597	-300	svtdbm4	...
2005-11-23-21.56.37.363384	-300	svtdbm4	...
			...
			...
			...
			...
			...
			...
2005-11-23-21.56.35.880314	-300	svtdbm4	...
			...

4 registro(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

...	DBPARTITIONNUM	DBNAME	PID	PROCESSNAME	...
...	0	CAPTAIN	4239374	db2agent (CAPTAIN)	0 ...
...					...
...					...
...					...
...					...
...					...
...					...
...	0	CAPTAIN	4239374	db2agent (CAPTAIN)	0 ...
...	0	CAPTAIN	4239374	db2agent (CAPTAIN)	0 ...
...					...
...					...
...					...
...					...
...	0	CAPTAIN	4239374	db2agent (CAPTAIN)	0 ...
...					...
...					...

Datos de salida de esta consulta (continuación).

...	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 procesamiento interno inesperado.
...           SE HAN CONCLUIDO TODOS LOS PROCESOS DE DB2
...           ASOCIADOS A ESTA INSTANCIA.
...           Se ha grabado información de
...           diagnóstico. Solicite asistencia técnica
...           al Centro de soporte de IBM.
... 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.

```

## **Función de tabla PD\_GET\_LOG\_MSGS**

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 204 en la página 733 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 la
... 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 la antememoria de paquete. No hay ningún error pero esto indica que la antememoria de paquete ha superado el tamaño máximo configurado. Si esta condición persiste, debería ajustar el parámetro de

```

configuración PCKCACHESZ DB.
2005-06-27-18.41.22.248000 ADM4500W Se ha producido una condición de
desbordamiento de la antememoria de paquete.
No hay ningún error pero esto indica que la
antememoria de paquete ha superado el tamaño
máximo configurado. Si esta condición
persiste, debería 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, DBPARTITONNUM, 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
...                               DB2 RELACIONADOS CON ESTA
...                               INSTANCIA SE HAN
...                               SUSPENDIDO. 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 la hora.

Tabla 204. 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 ha escrito el mensaje. Para mensajes escritor 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 escritor por aplicaciones que utilizan la API db2AdminMsgWrite, se devuelve "Función de usuario".
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.

Tabla 204. 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
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 205. 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.
INDEX_SCHEMA	VARCHAR(128)	Nombre de esquema de índice.
INDEX_NAME	VARCHAR(128)	Nombre de 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 la grabación de la entrada de índice en una página hoja. Éste es el tamaño promedio de la entrada de índice que, excluida cualquier actividad general, y que se calcula desde la longitud de columna media de todas las columnas que participan en el índice.
NONLEAF_RECSIZE	BIGINT	Tamaño de la grabación de la entrada de índice en una página no hoja. Éste es el tamaño promedio de la entrada de índice, excluida cualquier actividad general, y que se calcula desde la longitud de columna media de todas las columnas que participan en el índice, excepto cualquier columna INCLUDE.
LEAF_PAGE_OVERHEAD	BIGINT	Espacio reservado en la página hoja de índice para uso interno.

Tabla 205. Información devuelta por el procedimiento REORGCHK\_IX\_STATS (continuación)

Nombre de columna	Tipo de datos	Descripción
NONLEAF_PAGE_OVERHEAD	BIGINT	Espacio reservado en la página no hoja de índice para uso interno use
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 206. 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.
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.
ACTIVE_BLOCKS	BIGINT	Número total de bloques activos de una tabla de clustering de múltiples dimensiones (MDC). Sólo se puede aplicar este campo a las tablas definidas mediante 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.

```
1
-----...--
SQL0551N  "" no tiene el privilegio necesario para realizar
una operación "" en el objeto "".
```

## SYSINSTALLOBJECTS

El procedimiento SYSINSTALLOBJECTS crea o descarta los objetos de base de datos requeridos para una herramienta específica.

### Sintaxis

►►SYSINSTALLOBJECTS(—*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*

Reservado para su utilización en el futuro. El esquema SYSTOOLS siempre se utiliza, independientemente del nombre pasado a este parámetro.

## **Ejemplo**

```
CALL SYSPROC.SYSINSTALLOBJECTS('AM', 'C', CAST (NULL AS VARCHAR(128)),  
CAST (NULL AS VARCHAR(128)))
```





## Capítulo 4. Rutinas administrativas de SQL obsoletas y sus vistas o rutinas de sustitución

Con el fin de ofrecer un mayor soporte en DB2 Versión 9.5 para las rutinas administrativas existentes, algunas de las rutinas de DB2 UDB para Linux, UNIX y Windows Versión 9.1 se han sustituido por otras rutinas o vistas nuevas y más completas.

Las aplicaciones que utilizan las funciones de la tabla DB2 UDB para Linux, UNIX y Windows Versión 9 se deben modificar para utilizar 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, \_V95). 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 207. Rutinas administrativas de SQL obsoletas y sus vistas o rutinas de sustitución para DB2 UDB para Linux, UNIX y Windows Versión 9.5*

<b>Función obsoleta en DB2 UDB para Linux, UNIX y Windows Versión 9.1</b>	<b>Función o vista nueva en DB2 Versión 9.5</b>
“Tabla de función ADMIN_GET_TAB_INFO - Recuperar información de tamaño y estado para las tablas” en la página 747	“Vista administrativa ADMIN_TABINFO y función de tabla ADMIN_GET_TAB_INFO_V95 – Recuperar información de tamaño y estado para las tablas” en la página 220
“Función de tabla SNAP_GET_APPL – Recuperar información de instantáneas del grupo de datos lógico appl” en la página 759	“Vista administrativa SNAPAPPL y función de tabla SNAP_GET_APPL_V95 – Recuperar información de instantáneas del grupo de datos lógicos appl” en la página 360
“Función de tabla SNAP_GET_APPL_INFO – Recuperación de información de instantáneas del grupo de datos lógico appl_info” en la página 767	“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 352
“Función de tabla SNAP_GET_BP – Recuperación de la información de instantáneas del grupo de datos lógico bufferpool” en la página 773	“Vista administrativa SNAPBP y función de tabla SNAP_GET_BP_V95 – Recuperar información de instantáneas del grupo lógico bufferpool” en la página 369
“Función de tabla SNAP_GET_DB_V91 - Recuperar información de instantánea desde el grupo lógico dbase” en la página 790	“Vista administrativa SNAPDB y función de tabla SNAP_GET_DB_V95 – Recuperar información de instantánea desde el grupo lógico dbase” en la página 382

Tabla 207. Rutinas administrativas de SQL obsoletas y sus vistas o rutinas de sustitución para DB2 UDB para Linux, UNIX y Windows Versión 9.5 (continuación)

<b>Función obsoleta en DB2 UDB para Linux, UNIX y Windows Versión 9.1</b>	<b>Función o vista nueva en DB2 Versión 9.5</b>
“Función de tabla SNAP_GET_DBM – Recuperación de la información de instantáneas del grupo de datos lógicos dbm” en la página 787	“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 399
“Función de tabla SNAP_GET_DYN_SQL_V91 - Recuperar información de la instantánea del grupo de datos lógicos dynsql” en la página 801	“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 409

En el release anterior, DB2 UDB para Linux, UNIX y Windows Versión 9, también había funciones nuevas que sustituían a las funciones de la DB2 UDB para Linux, UNIX y Windows Versión 8.

Tabla 208. Rutinas administrativas de SQL obsoletas y sus vistas o rutinas de sustitución para DB2 UDB para Linux, UNIX y Windows Versión 9.1

<b>Función obsoleta en DB2 UDB para Linux, UNIX y Windows Versión 8</b>	<b>Vista o función nueva en DB2 Versión 9</b>
“RGET_DB_CONFIG” en la página 757	“Vista administrativa DBCFG - Recuperar información sobre los parámetros de configuración de la base de datos” en la página 237
“GET_DBM_CONFIG” en la página 758	“Vista administrativa DBMCFG - Recuperar información sobre los parámetros de configuración del gestor de bases de datos” en la página 239
“SNAP_GET_CONTAINER” en la página 778	“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 378
“SNAP_GET_DB” en la página 779	“Función de tabla SNAP_GET_DB_V91 - Recuperar información de instantánea desde el grupo lógico dbase” en la página 790
SNAP_GET_DETAILLOG (1)	“Vista administrativa SNAPDETAILLOG y función de tabla SNAP_GET_DETAILLOG_V91 – Recuperar información de instantánea desde el grupo de datos lógicos detail_log” en la página 406
“SNAP_GET_DYN_SQL” en la página 805	“Función de tabla SNAP_GET_DYN_SQL_V91 - Recuperar información de la instantánea del grupo de datos lógicos dynsql” en la página 801
“SNAP_GET_STO_PATHS” en la página 807	“Vista administrativa SNAPSTORAGE_PATHS y función de tabla SNAP_GET_STORAGE_PATHS – Recuperar información de vía de acceso del almacenamiento automático” en la página 442
“SNAP_GET_TAB” en la página 808	“Vista administrativa SNAPTAB y función de tabla SNAP_GET_TAB_V91 – Recuperar información de instantáneas del grupo de datos lógicos table” en la página 451

Tabla 208. Rutinas administrativas de SQL obsoletas y sus vistas o rutinas de sustitución para DB2 UDB para Linux, UNIX y Windows Versión 9.1 (continuación)

Función obsoleta en DB2 UDB para Linux, UNIX y Windows Versión 8	Vista o función nueva en DB2 Versión 9
"SNAP_GET_TBSP" en la página 809	"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 459
"SNAP_GET_TBSP_PART" en la página 813	"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 466
"SNAPSHOT_AGENT" en la página 815	"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 346
"SNAPSHOT_APPL" en la página 816	"Función de tabla SNAP_GET_APPL – Recuperar información de instantáneas del grupo de datos lógico appl" en la página 759
"SNAPSHOT_APPL_INFO" en la página 822	"Función de tabla SNAP_GET_APPL_INFO – Recuperación de información de instantáneas del grupo de datos lógico appl_info" en la página 767
"SNAPSHOT_BP" en la página 824	"Función de tabla SNAP_GET_BP – Recuperación de la información de instantáneas del grupo de datos lógico bufferpool" en la página 773
"SNAPSHOT_CONTAINER" en la página 827	"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 378
"SNAPSHOT_DATABASE" en la página 828	"Función de tabla SNAP_GET_DB_V91 - Recuperar información de instantánea desde el grupo lógico dbase" en la página 790
"SNAPSHOT_DBM" en la página 835	"Función de tabla SNAP_GET_DBM – Recuperación de la información de instantáneas del grupo de datos lógicos dbm" en la página 787
"SNAPSHOT_DYN_SQL" en la página 837	"Función de tabla SNAP_GET_DYN_SQL_V91 - Recuperar información de la instantánea del grupo de datos lógicos dynsql" en la página 801
"SNAPSHOT_FCM" en la página 838	"Vista administrativa SNAPFCM y función de tabla SNAP_GET_FCM – Recuperar la información de instantáneas del grupo de datos lógicos fcm" en la página 415
"SNAPSHOT_FCMNODE" en la página 840	"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 417
"SNAPSHOT_FILEW" en la página 840	"Procedimiento SNAP_WRITE_FILE" en la página 484
"SNAPSHOT_LOCK" en la página 841	"Vista administrativa SNAPLOCK y función de tabla SNAP_GET_LOCK – Recuperar información de instantáneas del grupo de datos lógicos lock" en la página 423

Tabla 208. Rutinas administrativas de SQL obsoletas y sus vistas o rutinas de sustitución para DB2 UDB para Linux, UNIX y Windows Versión 9.1 (continuación)

Función obsoleta en DB2 UDB para Linux, UNIX y Windows Versión 8	Vista o función nueva en DB2 Versión 9
"SNAPSHOT_LOCKWAIT" en la página 842	"Vista administrativa SNAPLOCKWAIT y función de tabla SNAP_GET_LOCKWAIT – Recuperar información de instantáneas del grupo de datos lógicos lockwait" en la página 429
"SNAPSHOT QUIESCERS" en la página 844	"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 471
"SNAPSHOT_RANGES" en la página 845	"Vista administrativa SNAPTbsp_RANGE y función de tabla SNAP_GET_TBSP_RANGE – Recuperar información acerca de la instantánea de rango" en la página 474
"SNAPSHOT_STATEMENT" en la página 846	"Vista administrativa SNAPSTMT y función de tabla SNAP_GET_STMT – Recuperar información de instantáneas de sentencias" en la página 435
"SNAPSHOT_SUBSECT" en la página 848	"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 444
"SNAPSHOT_SWITCHES" en la página 850	"Vista administrativa SNAPSWITCHES y función de tabla SNAP_GET_SWITCHES – Recuperar información acerca del estado del conmutador de instantánea de base de datos" en la página 448
"SNAPSHOT_TABLE" en la página 852	"Vista administrativa SNAPTAB y función de tabla SNAP_GET_TAB_V91 – Recuperar información de instantáneas del grupo de datos lógicos table" en la página 451
"SNAPSHOT_TBREORG" en la página 853	"Vista administrativa SNAPTAB_REORG y función de tabla SNAP_GET_TAB_REORG - Recuperar información sobre instantáneas de reorganización de tabla" en la página 454
"SNAPSHOT_TBS" en la página 854	"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 459
"SNAPSHOT_TBS_CFG" en la página 857	"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 466
SNAPSHOT_UTIL (1)	"Vista administrativa SNAPUTIL y función de tabla SNAP_GET_UTIL – Recuperar información acerca de la instantánea de grupo de datos lógicos utility_info" en la página 478
SNAPSHOT_UTIL_PROG (1)	"Vista administrativa SNAPUTIL_PROGRESS y función de tabla SNAP_GET_UTIL_PROGRESS – Recuperar información acerca de la instantánea de grupo de datos lógicos de progreso" en la página 481

Tabla 208. Rutinas administrativas de SQL obsoletas y sus vistas o rutinas de sustitución para DB2 UDB para Linux, UNIX y Windows Versión 9.1 (continuación)

Función obsoleta en DB2 UDB para Linux, UNIX y Windows Versión 8	Vista o función nueva en DB2 Versión 9
"SQLCACHE_SNAPSHOT" en la página 859	"Función de tabla SNAP_GET_DYN_SQL_V91 - Recuperar información de la instantánea del grupo de datos lógicos dynsql" en la página 801. La información que devuelve la función de tabla SQLCACHE_SNAPSHOT se incluye ahora en esta función de tabla y vista nuevas.
SYSFUN.GROUPS (1)	Este procedimiento ha quedado obsoleto.
SYSFUN.GROUPS_FOR_USER (1)	"Función de tabla AUTH_LIST_GROUPS_FOR_AUTHID - Recuperación lista de pertenencia al grupo para un ID de autorización determinado" en la página 315
SYSFUN.USER_GROUPS (1)	Este procedimiento ha quedado obsoleto.
SYSFUN.USERS (1)	Este procedimiento ha quedado obsoleto.
"RSYSINSTALLROUTINES" en la página 860	Este procedimiento ha quedado obsoleto.

**Nota:** (1) Estas funciones estaban presentes en DB2 UDB para Linux, UNIX y Windows Versión 8, pero se omitieron en la documentación.

## 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\_V95 - Recuperar información de tamaño y estado para las tablas" en la página 220.

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 sobre 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 209. Metadatos de la función de tabla ADMIN\_GET\_TAB\_INFO

Nombre de columna	Tipo de datos	Descripción
TABSHEMA	VARCHAR(128)	Nombre de esquema.
TABNAME	VARCHAR(128)	Nombre de tabla.
TABTYPE	CHAR(1)	Tipo de tabla: <ul style="list-style-type: none"> <li>• 'H' = tabla de jerarquía</li> <li>• 'S' = tabla de consultas materializadas</li> <li>• 'T' = tabla</li> </ul>
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"> <li>• '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.</li> <li>• 'Y' = la tabla está disponible.</li> </ul> <p><b>Nota:</b> La realización de una recuperación en avance en una carga no recuperable establecerá una tabla en estado de no disponibilidad.</p>

Tabla 209. 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 209. Metadatos de la función de tabla ADMIN\_GET\_TAB\_INFO (continuación)

Nombre de columna	Tipo de datos	Descripción
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. Este valor sólo se informa para las tablas no particionadas. Para las tablas particionadas, este valor será 0.</p>
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. Este valor sólo se informa para las tablas no particionadas. Para las tablas particionadas, este valor será 0.</p>

Tabla 209. Metadatos de la función de tabla ADMIN\_GET\_TAB\_INFO (continuación)

Nombre de columna	Tipo de datos	Descripción
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>

Tabla 209. Metadatos de la función de tabla ADMIN\_GET\_TAB\_INFO (continuación)

Nombre de columna	Tipo de datos	Descripción
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>
LOB_OBJECT_P_SIZE	BIGINT	<p>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.</p>

Tabla 209. Metadatos de la función de tabla ADMIN\_GET\_TAB\_INFO (continuación)

Nombre de columna	Tipo de datos	Descripción
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"> <li>• 1 si están utilizándose índices de tipo 1.</li> <li>• 2 si están utilizándose índices de tipo 2.</li> </ul>
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'.

Tabla 209. 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"> <li>• ABORTED (en estado PAUSED, pero incapaz de RESUME; se necesita STOP)</li> <li>• EXECUTING</li> <li>• NULL (si no se ha realizado ninguna reorganización in situ en la tabla)</li> <li>• PAUSED</li> </ul>
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"> <li>• IN_PROGRESS</li> <li>• NULL (si no existe ninguna carga en progreso para la tabla y si la tabla no está en estado de carga pendiente)</li> <li>• PENDING</li> </ul>
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'.

Tabla 209. Metadatos de la función de tabla ADMIN\_GET\_TAB\_INFO (continuación)

Nombre de columna	Tipo de datos	Descripción
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)	'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).

Tabla 209. Metadatos de la función de tabla ADMIN\_GET\_TAB\_INFO (continuación)

Nombre de columna	Tipo de datos	Descripción
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.

## 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 237.

►►—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 210. 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 tabla 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 239.

▶▶ 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 211. 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).

## 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 360.

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 212 en la página 761 para ver una lista completa de la información que se puede devolver.

## Sintaxis

```
▶▶ SNAP_GET_APPL ( ( nombre_bd [ , núm-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.

### *núm-partición-bd*

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. Si *nombrebd* no se establece en NULL y *núm-partición-bd* se establece en NULL, implícitamente se establece -1 para *núm-partición-bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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 212. 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 la 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
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

Tabla 212. Información devuelta por la función de tabla SNAP\_GET\_APPL (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_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 base de datos
DIRECT_WRITES	BIGINT	direct_writes - Grabaciones directas en 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 escalas de bloqueo

Tabla 212. Información devuelta por la función de tabla SNAP\_GET\_APPL (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
X_LOCK_ESCALS	BIGINT	x_lock_escals - Escalas de bloqueo exclusivas
DEADLOCKS	BIGINT	deadlocks - Puntos muertos detectados
TOTAL_SORTS	BIGINT	total_sorts - Total de clasificaciones
TOTAL_SORT_TIME	BIGINT	total_sort_time - Tiempo total de clasificación
SORT_OVERFLOWS	BIGINT	sort_overflows - Clasificar desbordamientos
COMMIT_SQL_STMTS	BIGINT	commit_sql_stmts - Intento de sentencias de confirmación
ROLLBACK_SQL_STMTS	BIGINT	rollback_sql_stmts - Intento de sentencias de retrotracción
DYNAMIC_SQL_STMTS	BIGINT	dynamic_sql_stmts - Intento de sentencias de SQL dinámico
STATIC_SQL_STMTS	BIGINT	static_sql_stmts - Intento de sentencias de SQL estático
FAILED_SQL_STMTS	BIGINT	failed_sql_stmts - Las operaciones de sentencia han fallado
SELECT_SQL_STMTS	BIGINT	select_sql_stmts - Se han ejecutado sentencias Select de SQL
DDL_SQL_STMTS	BIGINT	ddl_sql_stmts - Sentencias DDL (Data definition language) de SQL
UID_SQL_STMTS	BIGINT	uid_sql_stmts - Se han ejecutado sentencias UPDATE/INSERT/DELETE de SQL
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 - Intento de vinculaciones/precompilaciones

Tabla 212. Información devuelta por la función de tabla SNAP\_GET\_APPL (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
OPEN_REM_CURS	BIGINT	open_rem_curs - Abrir cursores remotos
OPEN_REM_CURS_BLK	BIGINT	open_rem_curs_blk - Abrir cursores remotos con bloqueo
REJ_CURS_BLK	BIGINT	rej_curs_blk - Peticiones de bloquear cursor rechazadas
ACC_CURS_BLK	BIGINT	acc_curs_blk - Petición de bloquear cursor aceptadas
SQL_REQS_SINCE_COMMIT	BIGINT	sql_reqs_since_commit - Peticiones de SQL desde 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 - Abrir cursores locales
OPEN_LOC_CURS_BLK	BIGINT	open_loc_curs_blk - Abrir cursores locales con bloqueo
PKG_CACHE_LOOKUPS	BIGINT	pkg_cache_lookups - Paquete de búsquedas de antememoria
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 que trabaja 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 del agente de la 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"> <li>• DYNAMIC_PRIORITY</li> <li>• FIXED_PRIORITY</li> </ul>
PREFETCH_WAIT_TIME	BIGINT	prefetch_wait_time - Tiempo de espera para captación previa

Tabla 212. Información devuelta por la función de tabla SNAP\_GET\_APPL (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
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 de generación aleatoria
TOTAL_HASH_LOOPS	BIGINT	total_hash_loops - Total de bucles de generación aleatoria
HASH_JOIN_OVERFLOWS	BIGINT	hash_join_overflows - Desbordamientos de uniones de generación aleatoria
HASH_JOIN_SMALL_OVERFLOWS	BIGINT	hash_join_small_overflows - Pequeños desbordamientos de uniones de generación aleatoria
APPL_IDLE_TIME	BIGINT	appl_idle_time - Tipo de prioridad de la aplicación
UOW_LOCK_WAIT_TIME	BIGINT	uow_lock_wait_time - Unidad de trabajo de tiempo total de espera en bloqueos
UOW_COMP_STATUS	VARCHAR(14)	uow_comp_status - Unidad de estado de finalización 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"> <li>• APPL_END</li> <li>• UOWABEND</li> <li>• UOWCOMMIT</li> <li>• UOWDEADLOCK</li> <li>• UOWLOCKTIMEOUT</li> <li>• UOWROLLBACK</li> <li>• UOWUNKNOWN</li> </ul>
AGENT_USR_CPU_TIME_S	BIGINT	agent_usr_cpu_time - Tiempo de CPU de usuario que el agente utiliza
AGENT_USR_CPU_TIME_MS	BIGINT	agent_usr_cpu_time - Tiempo de CPU de usuario que el agente utiliza
AGENT_SYS_CPU_TIME_S	BIGINT	agent_sys_cpu_time - Tiempo de CPU de sistema que el agente utiliza
AGENT_SYS_CPU_TIME_MS	BIGINT	agent_sys_cpu_time - Tiempo de CPU de sistema que el agente utiliza

Tabla 212. Información devuelta por la función de tabla SNAP\_GET\_APPL (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
APPL_CON_TIME	TIMESTAMP	appl_con_time - Indicación de la hora de inicio de petición de conexión
CONN_COMPLETE_TIME	TIMESTAMP	conn_complete_time - Indicación de la hora de finalización de petición de conexión
LAST_RESET	TIMESTAMP	last_reset - Indicación de la hora de la última restauración
UOW_START_TIME	TIMESTAMP	uow_start_time - Unidad de indicación de la hora de inicio de trabajo
UOW_STOP_TIME	TIMESTAMP	uow_stop_time - Unidad de indicación de la hora de detención de trabajo
PREV_UOW_STOP_TIME	TIMESTAMP	prev_uow_stop_time - Unidad anterior de indicación de la hora de finalización del trabajo
UOW_ELAPSED_TIME_S	BIGINT	uow_elapsed_time - Unidad más reciente de tiempo de trabajo transcurrido
UOW_ELAPSED_TIME_MS	BIGINT	uow_elapsed_time - Unidad más reciente de tiempo de trabajo transcurrido
ELAPSED_EXEC_TIME_S	BIGINT	elapsed_exec_time - Tiempo transcurrido de ejecución de la sentencia
ELAPSED_EXEC_TIME_MS	BIGINT	elapsed_exec_time - Tiempo transcurrido de ejecución de la sentencia
INBOUND_COMM_ADDRESS	VARCHAR(32)	inbound_comm_address - Dirección de comunicaciones de entrada
LOCK_TIMEOUT_VAL	BIGINT	lock_timeout_val - Tiempo 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



Tabla 212. Información devuelta por la función de tabla SNAP\_GET\_APPL (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
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 - Nivel límite 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 352.

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 213 en la página 769 para ver una lista completa de la información que se puede devolver.

### Sintaxis

```

▶▶ SNAP_GET_APPL_INFO ( ( nombre-bd [ , nm-particin-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 vlido 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-partición-bd*

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. Si *nombrebd* no se establece en NULL y *núm\_partición\_bd* se establece en NULL, implícitamente se establece -1 para *núm\_partición\_bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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 SELECT 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 registros(s) seleccionado(s).

## Información devuelta

Tabla 213. 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 <code>sqlmon.h</code>, y es uno de los siguientes:</p> <ul style="list-style-type: none"> <li>• BACKUP</li> <li>• COMMIT_ACT</li> <li>• COMP</li> <li>• CONNECTED</li> <li>• CONNECTPEND</li> <li>• CREATE_DB</li> <li>• DECOUPLED</li> <li>• DISCONNECTPEND</li> <li>• INTR</li> <li>• IOERROR_WAIT</li> <li>• LOAD</li> <li>• LOCKWAIT</li> <li>• QUIESCE_TABLESPACE</li> <li>• RECOMP</li> <li>• REMOTE_RQST</li> <li>• RESTART</li> <li>• RESTORE</li> <li>• ROLLBACK_ACT</li> <li>• ROLLBACK_TO_SAVEPOINT</li> <li>• TEND</li> <li>• THABRT</li> <li>• THCOMT</li> <li>• TPREP</li> <li>• UNLOAD</li> <li>• UOWEXEC</li> <li>• UOWWAIT</li> <li>• WAITFOR_REMOTE</li> </ul>

Tabla 213. Información devuelta por la función de tabla SNAP\_GET\_APPL\_INFO (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
CODEPAGE_ID	BIGINT	codepage_id - ID de página de códigos utilizada por la aplicación
NUM_ASSOC_AGENTS	BIGINT	num_assoc_agents - Número de agentes asociados
COORD_NODE_NUM	SMALLINT	coord_node - Coordinación del nodo
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 bases 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"> <li>• BINDADD</li> <li>• CONNECT</li> <li>• CREATE_EXT_RT</li> <li>• CREATE_NOT_FENC</li> <li>• CREATETAB</li> <li>• DBADM</li> <li>• IMPLICIT_SCHEMA</li> <li>• LOAD</li> <li>• LIBADM</li> <li>• QUIESCE_CONN</li> <li>• SECADM</li> <li>• SYSADM</li> <li>• SYSCTRL</li> <li>• SYSMANT</li> <li>• SYSMON</li> <li>• SYSQUIESCE</li> </ul> <p>Valores posibles para "source":</p> <ul style="list-style-type: none"> <li>• USER – autorización otorgada al usuario o a una función otorgada al usuario.</li> <li>• GROUP – autorización otorgada a un grupo al que pertenece el usuario o a una función otorgada al grupo al que pertenece el usuario.</li> </ul>

Tabla 213. 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_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 la 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"> <li>• AIX</li> <li>• AIX64</li> <li>• AS400_DRDA</li> <li>• DOS</li> <li>• DYNIX</li> <li>• HP</li> <li>• HP64</li> <li>• HPIA</li> <li>• HPIA64</li> <li>• LINUX</li> <li>• LINUX390</li> <li>• LINUXIA64</li> <li>• LINUXPPC</li> <li>• LINUXPPC64</li> <li>• LINUXX8664</li> <li>• LINUXZ64</li> <li>• MAC</li> <li>• MVS_DRDA</li> <li>• NT</li> <li>• NT64</li> <li>• OS2</li> <li>• OS390</li> <li>• SCO</li> <li>• SGI</li> <li>• SNI</li> <li>• SUN</li> <li>• SUN64</li> <li>• UNKNOWN</li> <li>• UNKNOWN_DRDA</li> <li>• VM_DRDA</li> <li>• VSE_DRDA</li> <li>• WINDOWS</li> <li>• WINDOWS95</li> </ul>

Tabla 213. 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"> <li>• CPIC</li> <li>• LOCAL</li> <li>• NETBIOS</li> <li>• NPIPE</li> <li>• TCPIP (para DB2 UDB)</li> <li>• TCPIP4</li> <li>• TCPIP6</li> </ul>
TERRITORY_CODE	SMALLINT	territory_code - Código de territorio de base de datos
APPL_NAME	VARCHAR(256)	appl_name - Nombre de la 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 está obsoleto. El valor devuelto no es un valor válido.
CLIENT_PRDID	VARCHAR(128)	client_prdid - ID de producto/versión del 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 la aplicación
DB_NAME	VARCHAR(128)	db_name - Nombre de la 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 - Señal de correlación de 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 213. 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 369.

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 214 en la página 774 para ver una lista completa de la información que se puede devolver.

### Sintaxis

```

▶▶ SNAP_GET_BP ( ( nombre-bd
                  [ , nm-particin-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 vlido 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 vaca para tomar la instantnea desde la base de datos conectada actualmente. Especifique un valor NULL para tomar la instantnea desde todas las bases de datos que se encuentran en la misma instancia que la base de datos conectada actualmente.

*nm-particin-bd*

Argumento de entrada opcional de tipo INTEGER que especifica un nmero

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. Si *nombrebd* no se establece en NULL y *núm\_partición\_bd* se establece en NULL, implícitamente se establece -1 para *núm\_partición\_bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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 registros(s) seleccionado(s).

Datos de salida de esta consulta (continuación).

```
... TOTAL_PHYSICAL_READS DBPARTITIONNUM
... -----
...                0                0
...                0                0
...                0                0
```

### Información devuelta

Tabla 214. 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.
BP_NAME	VARCHAR(128)	bp_name - Nombre de la agrupación de almacenamientos intermedios



Tabla 214. Información devuelta por la función de tabla SNAP\_GET\_BP (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
DB_NAME	VARCHAR(128)	db_name - Nombre de la 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 la agrupación de almacenamientos intermedios
POOL_ASYNC_DATA_WRITES	BIGINT	pool_async_data_writes - Grabaciones de datos asíncronas de la agrupación de almacenamientos intermedios

Tabla 214. 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 índices asíncronas de la agrupación de almacenamientos intermedios
POOL_ASYNC_INDEX_WRITES	BIGINT	pool_async_index_writes - Grabaciones de índices asíncronas de la agrupación de almacenamientos intermedios
POOL_ASYNC_XDA_READS	BIGINT	pool_async_xda_reads - Lecturas de datos XDA asíncronas de la agrupación de almacenamientos intermedios
POOL_ASYNC_XDA_WRITES	BIGINT	pool_async_xda_writes - Grabaciones de datos XDA asíncronas de la agrupación de almacenamientos intermedios
POOL_ASYNC_READ_TIME	BIGINT	pool_async_read_time - Tiempo de lectura asíncrona de la agrupación de almacenamientos intermedios
POOL_ASYNC_WRITE_TIME	BIGINT	pool_async_write_time - Tiempo de grabación asíncrona de la agrupación de almacenamientos intermedios
POOL_ASYNC_DATA_READ_REQS	BIGINT	pool_async_data_read_reqs - Peticiones de lectura asíncrona de la agrupación de almacenamientos intermedios
POOL_ASYNC_INDEX_READ_REQS	BIGINT	pool_async_index_read_reqs - Peticiones de lectura de índices asíncronos de la agrupación de almacenamientos intermedios
POOL_ASYNC_XDA_READ_REQS	BIGINT	pool_async_xda_reqs - Peticiones de lectura XDA asíncronas de la agrupación de almacenamientos intermedios
DIRECT_READS	BIGINT	direct_reads - Lecturas directas de base de datos
DIRECT_WRITES	BIGINT	direct_writes - Grabaciones directas en 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 214. 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 - Almacenamientos intermedios 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 está suspendido. El sistema devuelve un valor NULL para el elemento del supervisor suspendido.
VECTORED_IOS	BIGINT	vectored_ios - Número de peticiones de E/S de vector
DBPARTITIONNUM	SMALLINT	La 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 378

►►—SNAP\_GET\_CONTAINER—(—nombre-bd—,—núm-partición-bd—)—————►►

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.

*núm-partición-bd*

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 215. 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 del espacio de tablas
CONTAINER_NAME	VARCHAR(256)	container_name - Nombre del contenedor
CONTAINER_ID	BIGINT	container_id - Identificación de contenedor
CONTAINER_TYPE	SMALLINT	container_type - Tipo de contenedor
TOTAL_PAGES	BIGINT	container_total_pages - Páginas totales en contenedor
USABLE_PAGES	BIGINT	container_usable_pages - Páginas útiles en contenedor

Tabla 215. Información devuelta por la función de tabla SNAP\_GET\_CONTAINER (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
ACCESSIBLE	SMALLINT	container_accessible - Accesibilidad del 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 790

▶▶—SNAP\_GET\_DB—(—nombre-bd—,—núm-partición-bd—)————▶▶

El esquema es SYSPROC.

La función de tabla SNAP\_GET\_DB devuelve información de instantáneas de la base de datos.

*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.

*núm-partición-bd*

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. 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 216. 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 la base de datos

Tabla 216. Información devuelta por la función de tabla SNAP\_GET\_DB (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
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 del nodo del catálogo
SERVER_PLATFORM	INTEGER	server_platform - Sistema operativo del servidor
DB_LOCATION	INTEGER	db_location - Ubicación de base de datos
DB_CONN_TIME	TIMESTAMP	db_conn_time - Indicación de la hora de activación de base de datos
LAST_RESET	TIMESTAMP	last_reset - Indicación de la hora de la última restauración
LAST_BACKUP	TIMESTAMP	last_backup - Última hora de copia de seguridad
CONNECTIONS_TOP	BIGINT	connections_top - Número máximo de conexiones simultáneas
TOTAL_CONS	BIGINT	total_cons - Conexiones desde la activación de la base
TOTAL_SEC_CONS	BIGINT	total_sec_cons - Conexiones secundarias
APPLS_CUR_CONS	BIGINT	appls_cur_cons - Aplicaciones conectadas en la actualidad
APPLS_IN_DB2	BIGINT	appls_in_db2 - Aplicaciones que se ejecutan actualmente en la base de datos
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 escalas de bloqueo

Tabla 216. Información devuelta por la función de tabla SNAP\_GET\_DB (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
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 - Número total de pila de clasificación asignada
SORT_SHRHEAP_ALLOCATED	BIGINT	sort_shrheap_allocated - Clasificar pila de compartimiento asignada en la actualidad
SORT_SHRHEAP_TOP	BIGINT	sort_shrheap_top - Clasificar nivel límite de pila de compartimiento
TOTAL_SORTS	BIGINT	total_sorts - Total de clasificaciones
TOTAL_SORT_TIME	BIGINT	total_sort_time - Tiempo total de clasificación
SORT_OVERFLOWS	BIGINT	sort_overflows - Clasificar desbordamientos
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 la 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 la agrupación de almacenamientos intermedios

Tabla 216. 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_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_INDEX_WRITES	BIGINT	pool_index_writes - Grabaciones de índices de la agrupación de almacenamientos intermedios
POOL_ASYNC_INDEX_READS	BIGINT	pool_async_index_reads - Lecturas de índices asíncronas de la agrupación de almacenamientos intermedios
POOL_ASYNC_INDEX_WRITES	BIGINT	pool_async_index_writes - Grabaciones de índices asíncronas 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 la agrupación de almacenamientos intermedios
POOL_ASYNC_WRITE_TIME	BIGINT	pool_async_write_time - Tiempo de grabación asíncrona de la agrupación de almacenamientos intermedios
POOL_ASYNC_DATA_READ_REQS	BIGINT	pool_async_data_read_reqs - Peticiones de lectura asíncrona de la agrupación de almacenamientos intermedios
POOL_ASYNC_INDEX_READ_REQS	BIGINT	pool_async_index_read_reqs - Peticiones de lectura de índices asíncronos de la agrupación de almacenamientos intermedios
POOL_NO_VICTIM_BUFFER	BIGINT	pool_no_victim_buffer - Almacenamientos intermedios sin víctimas de la agrupación de almacenamientos intermedios



Tabla 216. Información devuelta por la función de tabla SNAP\_GET\_DB (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 la agrupación de almacenamientos intermedios activados
POOL_DRTY_PG_STEAL_CLNS	BIGINT	pool_drty_pg_steal_clns - Limpiadores de páginas víctima de la agrupación de almacenamientos intermedios activados
POOL_DRTY_PG_THRSH_CLNS	BIGINT	pool_drty_pg_thrsh_clns - Limpiadores de umbral de la agrupación de almacenamientos intermedios activados
PREFETCH_WAIT_TIME	BIGINT	prefetch_wait_time - Tiempo de espera para captación previa
UNREAD_PREFETCH_PAGES	BIGINT	unread_prefetch_pages - Páginas de captación previa no leídas
DIRECT_READS	BIGINT	direct_reads - Lecturas directas de base de datos
DIRECT_WRITES	BIGINT	direct_writes - Grabaciones directas en 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
POOL_DATA_TO_ESTORE	BIGINT	El elemento del supervisor ESTORE pool_data_to_estore está suspendido. El sistema devolverá un valor NULL para el elemento del supervisor suspendido.
POOL_INDEX_TO_ESTORE	BIGINT	El elemento del supervisor ESTORE pool_index_to_estore está suspendido. El sistema devolverá un valor NULL para el elemento del supervisor suspendido.
POOL_INDEX_FROM_ESTORE	BIGINT	El elemento del supervisor ESTORE pool_index_from_estore está suspendido. El sistema devolverá un valor NULL para el elemento del supervisor suspendido.
POOL_DATA_FROM_ESTORE	BIGINT	El elemento del supervisor ESTORE pool_data_from_estore está suspendido. El sistema devolverá un valor NULL para el elemento del supervisor suspendido.

Tabla 216. Información devuelta por la función de tabla SNAP\_GET\_DB (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 la sentencia
ELAPSED_EXEC_TIME_MS	BIGINT	elapsed_exec_time - Tiempo transcurrido de ejecución de la sentencia
COMMIT_SQL_STMTS	BIGINT	commit_sql_stmts - Intento de sentencias de confirmación
ROLLBACK_SQL_STMTS	BIGINT	rollback_sql_stmts - Intento de sentencias de retrotracción
DYNAMIC_SQL_STMTS	BIGINT	dynamic_sql_stmts - Intento de sentencias de SQL dinámico
STATIC_SQL_STMTS	BIGINT	static_sql_stmts - Intento de sentencias de SQL estático
FAILED_SQL_STMTS	BIGINT	failed_sql_stmts - Las operaciones de sentencia han fallado
SELECT_SQL_STMTS	BIGINT	select_sql_stmts - Se han ejecutado sentencias Select de SQL
UID_SQL_STMTS	BIGINT	uid_sql_stmts - Se han ejecutado sentencias UPDATE/INSERT/DELETE de SQL
DDL_SQL_STMTS	BIGINT	ddl_sql_stmts - Sentencias DDL (Data definition language) 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_rollbacks - Retrotracciones internas
INT_DEADLOCK_ROLLBACKS	BIGINT	int_deadlock_rollbacks - 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 - Intento de vinculaciones/precompilaciones

Tabla 216. Información devuelta por la función de tabla SNAP\_GET\_DB (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
TOTAL_LOG_AVAILABLE	BIGINT	total_log_available - Registro total disponible
TOTAL_LOG_USED	BIGINT	total_log_used - Espacio de registro total utilizado
SEC_LOG_USED_TOP	BIGINT	sec_log_used_top - Espacio máximo de anotación cronológica utilizado
TOT_LOG_USED_TOP	BIGINT	tot_log_used_top - Espacio máximo de anotación cronológica total utilizado
SEC_LOGS_ALLOCATED	BIGINT	sec_logs_allocated - Anotaciones cronológicas secundarias asignadas en la actualidad
LOG_READS	BIGINT	log_reads - Número de páginas de anotación cronológica leídas
LOG_READ_TIME_S	BIGINT	log_read_time - Tiempo de lectura de anotación cronológica
LOG_READ_TIME_NS	BIGINT	log_read_time - Tiempo de lectura de anotación cronológica
LOG_WRITES	BIGINT	log_writes - Número de páginas de anotación cronológica escritas
LOG_WRITE_TIME_S	BIGINT	log_write_time - Tiempo de grabación de anotación cronológica
LOG_WRITE_TIME_NS	BIGINT	log_write_time - Tiempo de grabación de anotación cronológica
NUM_LOG_WRITE_IO	BIGINT	num_log_write_io - Número de grabaciones de anotación cronológica
NUM_LOG_READ_IO	BIGINT	num_log_read_io - Número de lecturas de anotación cronológica
NUM_LOG_PART_PAGE_IO	BIGINT	num_log_part_page_io - Número de grabaciones de página de anotación cronológica 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 anotación cronológica 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 registro por rehacer para la recuperación
LOG_HELD_BY_DIRTY_PAGES	BIGINT	log_held_by_dirty_pages - Cantidad de espacio de registro calculado por las páginas desfasadas

Tabla 216. Información devuelta por la función de tabla SNAP\_GET\_DB (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
PKG_CACHE_LOOKUPS	BIGINT	pkg_cache_lookups - Paquete de búsquedas de antememoria
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 - Nivel límite 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 - Nivel límite 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 de generación aleatoria

Tabla 216. Información devuelta por la función de tabla SNAP\_GET\_DB (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
TOTAL_HASH_LOOPS	BIGINT	total_hash_loops - Total de bucles de generación aleatoria
HASH_JOIN_OVERFLOWS	BIGINT	hash_join_overflows - Desbordamientos de uniones de generación aleatoria
HASH_JOIN_SMALL_OVERFLOWS	BIGINT	hash_join_small_overflows - Pequeños desbordamientos de uniones de generación aleatoria
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 399.

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 217 en la página 788 para ver una lista completa de la información que se puede devolver.

### Sintaxis

```

>> SNAP_GET_DBM ( [ 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 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. Si no se utiliza esta opción de entrada, todas las particiones de la base de datos devolverán datos.

Si *núm-partición-bd* se establece en NULL, los datos se intentarán leer desde el archivo que ha creado 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 217. 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 - Número total de pila de clasificación asignada
POST_THRESHOLD_SORTS	BIGINT	post_threshold_sorts - Clasificaciones posteriores a un umbral
PIPED_SORTS_REQUESTED	BIGINT	pipedsorts_requested - Clasificaciones mediante conexiones solicitadas
PIPED_SORTS_ACCEPTED	BIGINT	pipedsorts_accepted - Clasificaciones mediante conexiones aceptadas
REM_CONS_IN	BIGINT	rem_cons_in - Conexiones remotas con el gestor de bases de datos
REM_CONS_IN_EXEC	BIGINT	rem_cons_in_exec - Conexiones remotas que se ejecutan en el elemento del supervisor del 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 del supervisor del 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

Tabla 217. Información devuelta por la tabla de función SNAP\_GET\_DBM (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
AGENTS_WAITING_ON_TOKEN	BIGINT	agents_waiting_on_token - Agentes esperando 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"> <li>• ACTIVE</li> <li>• QUIESCE_PEND</li> <li>• QUIESCED</li> </ul>
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 desde el agrupamiento
AGENTS_CREATED_EMPTY_POOL	BIGINT	agents_created_empty_pool - Agentes creados debido a una agrupación de agentes 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 agentes
AGENTS_STOLEN	BIGINT	agents_stolen - Agentes robados
GW_TOTAL_CONS	BIGINT	gw_total_cons - Número total de intentos de conexiones 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 que están a la espera de respuesta
GW_CONS_WAIT_CLIENT	BIGINT	gw_cons_wait_client - Número de conexiones a la espera de que el cliente envíe una petición
POST_THRESHOLD_HASH_JOINS	BIGINT	post_threshold_hash_joins - Umbral de uniones de generación aleatoria
NUM_GW_CONN_SWITCHES	BIGINT	num_gw_conn_switches - Conmutadores de conexión
DB2START_TIME	TIMESTAMP	db2start_time - Iniciar indicación de la hora del gestor de bases de datos

Tabla 217. Información devuelta por la tabla de función SNAP\_GET\_DBM (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
LAST_RESET	TIMESTAMP	last_reset - Indicación de la hora de la última restauración
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 - Clasificar nivel límite de pila privada
DBPARTITIONNUM	SMALLINT	La 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 382.

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 218 en la página 792 encontrará una lista completa de la información que se devuelve.

### Sintaxis

```

▶▶ SNAP_GET_DB_V91 ( ( nombre-bd
                    [ , nóm-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.

*núm-partición-bd*

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. Si *nombrebd* no se establece en NULL y *núm-partición-bd* se establece en NULL, implícitamente se establece -1 para *núm-partición-bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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

- Autorización SYSMON
- Privilegio EXECUTE para la función de tabla SNAP\_GET\_DB\_V91.

### 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 218. 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 la 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"> <li>• ACTIVE</li> <li>• QUIESCE_PEND</li> <li>• QUIESCED</li> <li>• ROLLFWD</li> </ul>
CATALOG_PARTITION	SMALLINT	catalog_node - Número de nodo de catálogo
CATALOG_PARTITION_NAME	VARCHAR(128)	catalog_node_name - Nombre de red del nodo del catálogo

Tabla 218. 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"> <li>• AIX</li> <li>• AIX64</li> <li>• AS400_DRDA</li> <li>• DOS</li> <li>• DYNIX</li> <li>• HP</li> <li>• HP64</li> <li>• HPIA</li> <li>• HPIA64</li> <li>• LINUX</li> <li>• LINUX390</li> <li>• LINUXIA64</li> <li>• LINUXPPC</li> <li>• LINUXPPC64</li> <li>• LINUXX8664</li> <li>• LINUXZ64</li> <li>• MAC</li> <li>• MVS_DRDA</li> <li>• NT</li> <li>• NT64</li> <li>• OS2</li> <li>• OS390</li> <li>• SCO</li> <li>• SGI</li> <li>• SNI</li> <li>• SUN</li> <li>• SUN64</li> <li>• UNKNOWN</li> <li>• UNKNOWN_DRDA</li> <li>• VM_DRDA</li> <li>• VSE_DRDA</li> <li>• WINDOWS</li> <li>• WINDOWS95</li> </ul>
DB_LOCATION	VARCHAR(12)	<p>db_location - Ubicación 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:</p> <ul style="list-style-type: none"> <li>• LOCAL</li> <li>• REMOTE</li> </ul>

Tabla 218. 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_CONN_TIME	TIMESTAMP	db_conn_time - Indicación de la hora de activación de base de datos
LAST_RESET	TIMESTAMP	last_reset - Indicación de la hora de la última restauración
LAST_BACKUP	TIMESTAMP	last_backup - Última hora de copia de seguridad
CONNECTIONS_TOP	BIGINT	connections_top - Número máximo de conexiones simultáneas
TOTAL_CONS	BIGINT	total_cons - Conexiones desde la activación de la base
TOTAL_SEC_CONS	BIGINT	total_sec_cons - Conexiones secundarias
APPLS_CUR_CONS	BIGINT	appls_cur_cons - Aplicaciones conectadas en la actualidad
APPLS_IN_DB2	BIGINT	appls_in_db2 - Aplicaciones que se ejecutan actualmente en la base de datos
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 escalas 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 - Número total de pila de clasificación asignada
SORT_SHRHEAP_ALLOCATED	BIGINT	sort_shrheap_allocated - Clasificar pila de compartimiento asignada en la actualidad

Tabla 218. Información devuelta por la función de tabla SNAP\_GET\_DB\_V91 (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
SORT_SHRHEAP_TOP	BIGINT	sort_shrheap_top - Clasificar nivel límite de pila de compartimiento
POST_SHRTHRESHOLD_SORTS	BIGINT	post_shrthreshold_sorts - Clasificaciones posteriores a un umbral compartidas
TOTAL_SORTS	BIGINT	total_sorts - Total de clasificaciones
TOTAL_SORT_TIME	BIGINT	total_sort_time - Tiempo total de clasificación
SORT_OVERFLOWS	BIGINT	sort_overflows - Clasificar desbordamientos
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 la 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 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

Tabla 218. 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_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 índices asíncronas de la 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 índices asíncronas 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_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 la agrupación de almacenamientos intermedios
POOL_ASYNC_XDA_WRITES	BIGINT	pool_async_xda_writes - Grabaciones de datos XDA asíncronas 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_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 la agrupación de almacenamientos intermedios

Tabla 218. 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_ASYNC_WRITE_TIME	BIGINT	pool_async_write_time - Tiempo de grabación asíncrona de la agrupación de almacenamientos intermedios
POOL_ASYNC_DATA_READ_REQS	BIGINT	pool_async_data_read_reqs - Peticiones de lectura asíncrona de la agrupación de almacenamientos intermedios
POOL_ASYNC_INDEX_READ_REQS	BIGINT	pool_async_index_read_reqs - Peticiones de lectura de índices asíncronos de la agrupación de almacenamientos intermedios
POOL_ASYNC_XDA_READ_REQS	BIGINT	pool_async_xda_reqs - Peticiones de lectura XDA asíncronas de la agrupación de almacenamientos intermedios
POOL_NO_VICTIM_BUFFER	BIGINT	pool_no_victim_buffer - Almacenamientos intermedios sin víctimas de la agrupación de almacenamientos intermedios
POOL_LSN_GAP_CLNS	BIGINT	pool_lsn_gap_clns - Limpiadores de espacio de anotación cronológica de la agrupación de almacenamientos intermedios activados
POOL_DRTY_PG_STEAL_CLNS	BIGINT	pool_drty_pg_steal_clns - Limpiadores de páginas víctima de la agrupación de almacenamientos intermedios activados
POOL_DRTY_PG_THRSH_CLNS	BIGINT	pool_drty_pg_thrsh_clns - Limpiadores de umbral de la agrupación de almacenamientos intermedios activados
PREFETCH_WAIT_TIME	BIGINT	prefetch_wait_time - Tiempo de espera para captación previa
UNREAD_PREFETCH_PAGES	BIGINT	unread_prefetch_pages - Páginas de captación previa no leídas
DIRECT_READS	BIGINT	direct_reads - Lecturas directas de base de datos
DIRECT_WRITES	BIGINT	direct_writes - Grabaciones directas en 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

Tabla 218. Información devuelta por la función de tabla SNAP\_GET\_DB\_V91 (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
FILES_CLOSED	BIGINT	files_closed - Archivos de base de datos cerrados
ELAPSED_EXEC_TIME_S	BIGINT	elapsed_exec_time - Tiempo transcurrido de ejecución de la sentencia
ELAPSED_EXEC_TIME_MS	BIGINT	elapsed_exec_time - Tiempo transcurrido de ejecución de la sentencia
COMMIT_SQL_STMTS	BIGINT	commit_sql_stmts - Intento de sentencias de confirmación
ROLLBACK_SQL_STMTS	BIGINT	rollback_sql_stmts - Intento de sentencias de retrotracción
DYNAMIC_SQL_STMTS	BIGINT	dynamic_sql_stmts - Intento de sentencias de SQL dinámico
STATIC_SQL_STMTS	BIGINT	static_sql_stmts - Intento de sentencias de SQL estático
FAILED_SQL_STMTS	BIGINT	failed_sql_stmts - Las operaciones de sentencia han fallado
SELECT_SQL_STMTS	BIGINT	select_sql_stmts - Se han ejecutado sentencias Select de SQL
UID_SQL_STMTS	BIGINT	uid_sql_stmts - Se han ejecutado sentencias UPDATE/INSERT/DELETE de SQL
DDL_SQL_STMTS	BIGINT	ddl_sql_stmts - Sentencias DDL (Data definition language) 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



Tabla 218. Información devuelta por la función de tabla SNAP\_GET\_DB\_V91 (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
BINDS_PRECOMPILES	BIGINT	binds_precompiles - Intento de vinculaciones/precompilaciones
TOTAL_LOG_AVAILABLE	BIGINT	total_log_available - Registro total disponible
TOTAL_LOG_USED	BIGINT	total_log_used - Espacio de registro total utilizado
SEC_LOG_USED_TOP	BIGINT	sec_log_used_top - Espacio máximo de anotación cronológica utilizado
TOT_LOG_USED_TOP	BIGINT	tot_log_used_top - Espacio máximo de anotación cronológica total utilizado
SEC_LOGS_ALLOCATED	BIGINT	sec_logs_allocated - Anotaciones cronológicas secundarias asignadas en la actualidad
LOG_READS	BIGINT	log_reads - Número de páginas de anotación cronológica leídas
LOG_READ_TIME_S	BIGINT	log_read_time - Tiempo de lectura de anotación cronológica
LOG_READ_TIME_NS	BIGINT	log_read_time - Tiempo de lectura de anotación cronológica
LOG_WRITES	BIGINT	log_writes - Número de páginas de anotación cronológica escritas
LOG_WRITE_TIME_S	BIGINT	log_write_time - Tiempo de grabación de anotación cronológica
LOG_WRITE_TIME_NS	BIGINT	log_write_time - Tiempo de grabación de anotación cronológica
NUM_LOG_WRITE_IO	BIGINT	num_log_write_io - Número de grabaciones de anotación cronológica
NUM_LOG_READ_IO	BIGINT	num_log_read_io - Número de lecturas de anotación cronológica
NUM_LOG_PART_PAGE_IO	BIGINT	num_log_part_page_io - Número de grabaciones de página de anotación cronológica 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 anotación cronológica 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 registro por rehacer para la recuperación

Tabla 218. Información devuelta por la función de tabla SNAP\_GET\_DB\_V91 (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
LOG_HELD_BY_DIRTY_PAGES	BIGINT	log_held_by_dirty_pages - Cantidad de espacio de registro calculado por las páginas desfasadas
PKG_CACHE_LOOKUPS	BIGINT	pkg_cache_lookups - Paquete de búsquedas de antememoria
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 - Nivel límite 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 - Nivel límite 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

Tabla 218. Información devuelta por la función de tabla SNAP\_GET\_DB\_V91 (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
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 de generación aleatoria
TOTAL_HASH_LOOPS	BIGINT	total_hash_loops - Total de bucles de generación aleatoria
HASH_JOIN_OVERFLOWS	BIGINT	hash_join_overflows - Desbordamientos de uniones de generación aleatoria
HASH_JOIN_SMALL_OVERFLOWS	BIGINT	hash_join_small_overflows - Pequeños desbordamientos de uniones de generación aleatoria
POST_SHRTHRESHOLD_HASH_JOINS	BIGINT	post_shrthreshold_hash_joins - Uniones de generación aleatoria del umbral posterior
ACTIVE_HASH_JOINS	BIGINT	active_hash_joins - Uniones de generación aleatoria activas
NUM_DB_STORAGE_PATHS	BIGINT	num_db_storage_paths - Número de vías de acceso de almacenamiento automático
DBPARTITIONNUM	SMALLINT	La 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 registro 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 409.

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 219 en la página 803 para ver una lista completa de la información que se puede devolver.

### Sintaxis

►► SNAP\_GET\_DYN\_SQL\_V91 (—*nombre-bd*—, *núm-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 NULL o una serie vacía para tomar la instantánea desde la base de datos conectada actualmente.

### *núm-partición-bd*

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. Si *nombrebd* no se establece en NULL y *núm\_partición\_bd* se establece en NULL, implícitamente se establece -1 para *núm\_partición\_bd*. Si no se utiliza esta opción de entrada, es decir, solamente se proporciona *nombre-bd*, se devuelven los datos de todas las particiones de base de datos.

Si *nombre-bd* y *núm-partición-bd* se establecen en NULL, los datos se intentarán leer desde el archivo que ha creado 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 219. 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 sentencias
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_OVERFLOW	BIGINT	sort_overflows - Clasificar desbordamientos
TOTAL_SORT_TIME	BIGINT	total_sort_time - Tiempo total de clasificación
POOL_DATA_L_READS	BIGINT	pool_data_l_reads - Lecturas lógicas de datos de la agrupación de almacenamientos intermedios

Tabla 219. 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_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
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 de ejecución de sentencia transcurrido
TOTAL_EXEC_TIME_MS	BIGINT	total_exec_time - Tiempo de ejecución de sentencia transcurrido
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

Tabla 219. 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
TOTAL_SYS_CPU_TIME	BIGINT	total_sys_cpu_time - Total de CPU del sistema para una sentencia
TOTAL_SYS_CPU_TIME_MS	BIGINT	total_sys_cpu_time - Total de CPU del sistema para una sentencia
STMT_TEXT	CLOB(2 M)	stmt_text - Texto de sentencia de SQL
DBPARTITIONNUM	SMALLINT	La 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 801

►►—SNAP\_GET\_DYN\_SQL—(—nombre-bd—,—núm-partición-bd—)—————►

El esquema es SYSPROC.

La función de tabla SNAP\_GET\_DYN\_SQL devuelve información de instantáneas desde el grupo de datos lógicos dynsql.

*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.

*núm-partición-bd*

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. 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 220. 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.

Tabla 220. Información devuelta por la función de tabla SNAP\_GET\_DYN\_SQL (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
NUM_EXECUTIONS	BIGINT	num_executions - Ejecuciones de sentencia
NUM_COMPILATIONS	BIGINT	num_compilations - Compilaciones de sentencias
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_OVERFLOW	BIGINT	sort_overflows - Clasificar desbordamientos
TOTAL_SORT_TIME	BIGINT	total_sort_time - Tiempo total de clasificación
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



Tabla 220. Información devuelta por la función de tabla SNAP\_GET\_DYN\_SQL (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
TOTAL_EXEC_TIME	BIGINT	total_exec_time - Tiempo de ejecución de sentencia transcurrido
TOTAL_EXEC_TIME_MS	BIGINT	total_exec_time - Tiempo de ejecución de sentencia transcurrido
TOTAL_USR_TIME	BIGINT	total_usr_cpu_time - Total de CPU de usuario para una sentencia
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 del sistema para una sentencia
TOTAL_SYS_TIME_MS	BIGINT	total_sys_cpu_time - Total de CPU del 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 442

►►—SNAP\_GET\_STO\_PATHS—(—nombre-bd—,—núm-partición-bd—)—————◄◄

El esquema es SYSPROC.

La función de tabla SNAP\_GET\_STO\_PATHS devuelve información de instantáneas del grupo de datos lógicos storage\_paths.

*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.

*núm-partición-bd*

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. 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 221. Información devuelta por la función de tabla `SNAP_GET_STO_PATHS`

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>DB_NAME</code>	<code>VARCHAR(128)</code>	<code>db_name</code> - Nombre de la base de datos
<code>DB_STORAGE_PATH</code>	<code>VARCHAR(256)</code>	<code>db_storage_path</code> - 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 451

►► `SNAP_GET_TAB` (—*nombre-bd*—, —*núm-partición-bd*—) ◀◀

El esquema es `SYSPROC`.

La función de tabla `SNAP_GET_TAB` devuelve información de instantáneas desde el grupo de datos lógicos table.

*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.

*núm-partición-bd*

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. 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 222. 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.
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 los 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 459

►►—SNAP\_GET\_TBSP—(—nombre-bd—,—núm-partición-bd—)—————►►

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.

*núm-partición-bd*

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 223. 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 del 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 tabla
TBSP_EXTENT_SIZE	BIGINT	tablespace_extent_size - Tamaño de extensión de espacio de tabla
TBSP_PREFETCH_SIZE	BIGINT	tablespace_prefetch_size - Tamaño de captación previa de espacio de tabla
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 <sup>1</sup>	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

Tabla 223. 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_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 la 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 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
POOL_ASYNC_INDEX_READS	BIGINT	pool_async_index_reads - Lecturas de índices asíncronas de la 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 índices asíncronas 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

Tabla 223. 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_READ_TIME	BIGINT	pool_async_read_time - Tiempo de lectura asíncrona de la agrupación de almacenamientos intermedios
POOL_ASYNC_WRITE_TIME	BIGINT	pool_async_write_time - Tiempo de grabación asíncrona de la agrupación de almacenamientos intermedios
POOL_ASYNC_DATA_READ_REQS	BIGINT	pool_async_data_read_reqs - Peticiones de lectura asíncrona de la agrupación de almacenamientos intermedios
POOL_ASYNC_INDEX_READ_REQS	BIGINT	pool_async_index_read_reqs - Peticiones de lectura de índices asíncronos de la agrupación de almacenamientos intermedios
POOL_NO_VICTIM_BUFFER	BIGINT	pool_no_victim_buffer - Almacenamientos intermedios sin víctimas de la agrupación de almacenamientos intermedios
DIRECT_READS	BIGINT	direct_reads - Lecturas directas de base de datos
DIRECT_WRITES	BIGINT	direct_writes - Grabaciones directas en 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 está suspendido. El sistema devolverá un valor NULL para el elemento del supervisor suspendido.
POOL_INDEX_TO_ESTORE	BIGINT	El elemento del supervisor ESTORE pool_index_to_estore está suspendido. El sistema devolverá un valor NULL para el elemento del supervisor suspendido.
POOL_INDEX_FROM_ESTORE	BIGINT	El elemento del supervisor ESTORE pool_index_from_estore está suspendido. El sistema devolverá un valor NULL para el elemento del supervisor suspendido.

Tabla 223. 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 está suspendido. El sistema devolverá un valor NULL para el elemento del supervisor suspendido.
TBSP_REBALANCER_MODE	BIGINT	tablespace_rebalancer_mode - Modalidad de reequilibrador
TBSP_USING_AUTO_STORAGE	SMALLINT	tablespace_using_auto_storage - Uso del almacenamiento automático
TBSP_AUTO_RESIZE_ENABLED	SMALLINT	tablespace_auto_resize_enabled - Dimensionamiento automático habilitado

<sup>1</sup> 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 466

►►—SNAP\_GET\_TBSP\_PART—(—nombre-bd—,—núm-partición-bd—)—————►►

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.

*núm-partición-bd*

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 224. 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 del espacio de tablas
TBSP_STATE	BIGINT	tablespace_state - Estado de espacio de tabla
TBSP_PREFETCH_SIZE	BIGINT	tablespace_prefetch_size - Tamaño de captación previa de espacio de tabla
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 tabla de cambio de estado
TBSP_MIN_RECOVERY_TIME	TIMESTAMP	tablespace_min_recovery_time - Tiempo de recuperación mínimo para avance
TBSP_TOTAL_PAGES	BIGINT	tablespace_total_pages - Páginas totales en espacio de tabla
TBSP_USABLE_PAGES	BIGINT	tablespace_usable_pages - Páginas útiles en espacio de tabla
TBSP_USED_PAGES	BIGINT	tablespace_used_pages - Páginas utilizadas en espacio de tabla
TBSP_FREE_PAGES	BIGINT	tablespace_free_pages - Páginas libres en espacio de tabla
TBSP_PENDING_FREE_PAGES	BIGINT	tablespace_pending_free_pages - Páginas libres pendientes en espacio de tabla
TBSP_PAGE_TOP	BIGINT	tablespace_page_top - Nivel límite de espacio de tabla
REBALANCER_MODE	BIGINT	tablespace_rebalancer_mode - Modalidad de reequilibrador
REBALANCER_EXTENTS_REMAINING	BIGINT	tablespace_rebalancer_extents_remaining - Número total de extensiones que ha de procesar el reequilibrador
REBALANCER_EXTENTS_PROCESSED	BIGINT	tablespace_rebalancer_extents_processed - Número de extensiones que el reequilibrador ha procesado
REBALANCER_PRIORITY	BIGINT	tablespace_rebalancer_priority - Prioridad del reequilibrador actual
REBALANCER_START_TIME	TIMESTAMP	tablespace_rebalancer_start_time - Hora de inicio de reequilibrador



Tabla 224. 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_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 en la correlación del espacio de tabla
TBSP_NUM_CONTAINERS	BIGINT	tablespace_num_containers - Número de contenedores en espacio de tabla
TBSP_INITIAL_SIZE	BIGINT	tablespace_initial_size - Tamaño inicial de espacio de tabla
TBSP_CURRENT_SIZE	BIGINT	tablespace_current_size - Tamaño actual de espacio de tabla
TBSP_MAX_SIZE	BIGINT	tablespace_max_size - Tamaño máximo de espacio de tabla
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 346.

▶▶—SNAPSHOT\_AGENT—(—*nombrebd*—,—*númparticiónbd*—)—▶▶

El esquema es SYSPROC.

La función SNAPSHOT\_AGENT devuelve información sobre los agentes de una instantánea de aplicación.

*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. 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 225. Información devuelta por la función de tabla SNAPSHOT\_AGENT

Nombre de columna	Tipo de datos	Descripción o elemento de 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

**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 759.

►►—SNAPSHOT\_APPL—(—nombred—,—númparticiónbd—)—————◄◄

El esquema es SYSPROC.

La función SNAPSHOT\_APPL devuelve información general de una instantánea de aplicación.

*nombred*

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. 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 226. Información devuelta por la función de tabla SNAPSHOT\_APPL

Nombre de columna	Tipo de datos	Descripción o elemento de 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 - Espacio de anotaciones cronológicas de unidad 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 almacenamiento intermedio
POOL_DATA_P_READS	BIGINT	pool_data_p_reads - Lecturas físicas de datos de agrupación de almacenamiento intermedio
POOL_DATA_WRITES	BIGINT	pool_data_writes - Grabaciones de datos de agrupación de almacenamiento intermedio
POOL_INDEX_L_READS	BIGINT	pool_index_l_reads - Lecturas lógicas de índice de agrupación de almacenamiento intermedio
POOL_INDEX_P_READS	BIGINT	pool_index_p_reads - Lecturas físicas de índice de agrupación de almacenamiento intermedio
POOL_INDEX_WRITES	BIGINT	pool_index_writes - Grabaciones de índice de agrupación de almacenamiento intermedio
POOL_READ_TIME	BIGINT	pool_read_time - Tiempo total de lectura física de agrupación de almacenamiento intermedio
POOL_WRITE_TIME	BIGINT	pool_write_time - Tiempo total de grabación física de agrupación de almacenamiento intermedio
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 226. Información devuelta por la función de tabla *SNAPSHOT\_APPL* (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento de 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
POOL_DATA_TO_ESTORE	BIGINT	El elemento de supervisor de 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 de 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 de 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 de 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 mantenidos
LOCK_WAITS	BIGINT	lock_waits - Esperas de bloqueo
LOCK_WAIT_TIME	BIGINT	lock_wait_time - Tiempo que se ha esperado en los bloqueos
LOCK_ESCALS	BIGINT	lock_escals - Número de escaladas de bloqueo
X_LOCK_ESCALS	BIGINT	x_lock_escals - Escaladas 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

Tabla 226. Información devuelta por la función de tabla *SNAPSHOT\_APPL* (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento de supervisor correspondiente
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 falladas
SELECT_SQL_STMTS	BIGINT	select_sql_stmts - Selección de sentencias de SQL ejecutada
DDL_SQL_STMTS	BIGINT	ddl_sql_stmts - Sentencias de SQL de lenguaje de definición de datos (DDL)
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 un 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 - Abrir cursores remotos
OPEN_REM_CURS_BLK	BIGINT	open_rem_curs_blk - Abrir cursores remotos 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

Tabla 226. Información devuelta por la función de tabla *SNAPSHOT\_APPL* (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento de supervisor correspondiente
SQL_REQS_SINCE_COMMIT	BIGINT	sql_reqs_since_commit - Peticiones de SQL desde 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 - Abrir cursores locales 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 durante el que se ha esperado la 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 esperando en bloqueos
TOTAL_HASH_JOINS	BIGINT	total_hash_joins - Uniones hash totales
TOTAL_HASH_LOOPS	BIGINT	total_hash_loops - Bucles hash totales

Tabla 226. Información devuelta por la función de tabla *SNAPSHOT\_APPL* (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento de supervisor correspondiente
HASH_JOIN_OVERFLOW	BIGINT	hash_join_overflows - Desbordamientos de uniones hash
HASH_JOIN_SMALL_OVERFLOW	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 terminación de unidad de trabajo
AGENT_USR_CPU_TIME_S	BIGINT	agent_usr_cpu_time - Tiempo de CPU de usuario utilizado por agente
AGENT_USR_CPU_TIME_MS	BIGINT	agent_usr_cpu_time - Tiempo de CPU de usuario utilizado por agente
AGENT_SYS_CPU_TIME_S	BIGINT	agent_sys_cpu_time - Tiempo de CPU de sistema utilizado por agente
AGENT_SYS_CPU_TIME_MS	BIGINT	agent_sys_cpu_time - Tiempo de CPU de sistema utilizado por 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 terminación 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 terminación 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 226. Información devuelta por la función de tabla *SNAPSHOT\_APPL* (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento de 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

## SNAPSHOT\_APPL\_INFO

**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 767.

►►—SNAPSHOT\_APPL\_INFO—(—nombre-bd—,—núm-partición-bd—)——►►

El esquema es SYSPROC.

La función *SNAPSHOT\_APPL\_INFO* devuelve información general de una instantánea de aplicación.

*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 todas las bases de datos bajo la instancia de la base de datos.

*núm-partición-bd*

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. 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 227. 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.



Tabla 227. Información devuelta por la función de tabla  
SNAPSHOT\_APPL\_INFO (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)
APPL_STATUS	BIGINT	appl_status - Estado de aplicación
CODEPAGE_ID	BIGINT	codepage_id - ID de página de códigos utilizada por la aplicación
NUM_ASSOC_AGENTS	BIGINT	num_assoc_agents - Número de agentes asociados
COORD_PARTITION_NUM	BIGINT	coord_node - Coordinación del nodo
AUTHORITY_LVL	BIGINT	authority_lvl - Nivel de autorización del 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 la 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 la 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 del supervisor client_nname está obsoleto. El valor devuelto no es un valor válido.
CLIENT_PRDID	VARCHAR(128)	client_prdid - ID de producto/versión del 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 la aplicación
DB_NAME	VARCHAR(128)	db_name - Nombre de la base de datos

Tabla 227. Información devuelta por la función de tabla *SNAPSHOT\_APPL\_INFO* (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
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 - Señal de correlación de 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

**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 773.

►►—SNAPSHOT\_BP—(—*nombrebd*—,—*númparticiónbd*—)—◄◄

El esquema es SYSPROC.

La función SNAPSHOT\_BP devuelve información de una instantánea de agrupación de almacenamientos intermedios.

*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. 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 228. Información devuelta por la función de tabla `SNAPSHOT_BP`*

Nombre de columna	Tipo de datos	Descripción o elemento de supervisor correspondiente
<code>SNAPSHOT_TIMESTAMP</code>	<code>TIMESTAMP</code>	La fecha y la hora en que se tomó la instantánea.
<code>POOL_DATA_L_READS</code>	<code>BIGINT</code>	<code>pool_data_l_reads</code> - Lecturas lógicas de datos de agrupación de almacenamiento intermedio
<code>POOL_DATA_P_READS</code>	<code>BIGINT</code>	<code>pool_data_p_reads</code> - Lecturas físicas de datos de agrupación de almacenamiento intermedio
<code>POOL_DATA_WRITES</code>	<code>BIGINT</code>	<code>pool_data_writes</code> - Grabaciones de datos de agrupación de almacenamiento intermedio
<code>POOL_INDEX_L_READS</code>	<code>BIGINT</code>	<code>pool_index_l_reads</code> - Lecturas lógicas de índice de agrupación de almacenamiento intermedio
<code>POOL_INDEX_P_READS</code>	<code>BIGINT</code>	<code>pool_index_p_reads</code> - Lecturas físicas de índice de agrupación de almacenamiento intermedio
<code>POOL_INDEX_WRITES</code>	<code>BIGINT</code>	<code>pool_index_writes</code> - Grabaciones de índice de agrupación de almacenamiento intermedio
<code>POOL_READ_TIME</code>	<code>BIGINT</code>	<code>pool_read_time</code> - Tiempo total de lectura física de agrupación de almacenamiento intermedio
<code>POOL_WRITE_TIME</code>	<code>BIGINT</code>	<code>pool_write_time</code> - Tiempo total de grabación física de agrupación de almacenamiento intermedio
<code>POOL_ASYNC_DATA_READS</code>	<code>BIGINT</code>	<code>pool_async_data_reads</code> - Lecturas de datos asíncronas de agrupación de almacenamiento intermedio
<code>POOL_ASYNC_DATA_WRITES</code>	<code>BIGINT</code>	<code>pool_async_data_writes</code> - Grabaciones de datos asíncronas de agrupación de almacenamiento intermedio
<code>POOL_ASYNC_INDEX_WRITES</code>	<code>BIGINT</code>	<code>pool_async_index_writes</code> - Grabaciones de índice asíncronas de agrupación de almacenamiento intermedio
<code>POOL_ASYNC_READ_TIME</code>	<code>BIGINT</code>	<code>pool_async_read_time</code> - Tiempo de lectura asíncrona de agrupación de almacenamiento intermedio
<code>POOL_ASYNC_WRITE_TIME</code>	<code>BIGINT</code>	<code>pool_async_write_time</code> - Tiempo de grabación asíncrona de agrupación de almacenamiento intermedio

Tabla 228. Información devuelta por la función de tabla *SNAPSHOT\_BP* (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento de supervisor correspondiente
POOL_ASYNC_DATA_READ_REQS	BIGINT	pool_async_data_read_reqs - Peticiones de lectura asíncrona de agrupación de almacenamiento intermedio
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 almacenamiento intermedio
POOL_DATA_TO_ESTORE	BIGINT	El elemento de supervisor de 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 de 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 de 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 de 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 base de datos cerrados
BP_NAME	VARCHAR(128)	bp_name - Nombre de agrupación de almacenamiento intermedio

Tabla 228. Información devuelta por la función de tabla *SNAPSHOT\_BP* (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento de supervisor correspondiente
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

**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 378

▶▶—SNAPSHOT\_CONTAINER—(—nombrebd—,—númparticiónbd—)————▶▶

El esquema es SYSPROC.

La función SNAPSHOT\_CONTAINER devuelve información de configuración de contenedor de una instantánea de espacio 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 todas las particiones de base de datos. 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 229. Información devuelta por la función de tabla *SNAPSHOT\_CONTAINER*

Nombre de columna	Tipo de datos	Descripción o elemento de 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 tabla

Tabla 229. Información devuelta por la función de tabla *SNAPSHOT\_CONTAINER* (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento de supervisor correspondiente
TABLESPACE_NAME	VARCHAR(128)	tablespace_name - Nombre de espacio de tabla
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

**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 790

►►—SNAPSHOT\_DATABASE—(—nombrebd—,—númparticiónbd—)—————►

El esquema es SYSPROC.

La función SNAPSHOT\_DATABASE devuelve información de una instantánea de 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 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. 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 230. Información devuelta por la función de tabla `SNAPSHOT_DATABASE`*

Nombre de columna	Tipo de datos	Descripción o elemento de supervisor correspondiente
<code>SNAPSHOT_TIMESTAMP</code>	<code>TIMESTAMP</code>	La fecha y la hora en que se tomó la instantánea.
<code>SEC_LOG_USED_TOP</code>	<code>BIGINT</code>	<code>sec_log_used_top</code> - Espacio máximo utilizado de anotaciones cronológicas secundarias
<code>TOT_LOG_USED_TOP</code>	<code>BIGINT</code>	<code>tot_log_used_top</code> - Espacio máximo utilizado de anotaciones cronológicas totales
<code>TOTAL_LOG_USED</code>	<code>BIGINT</code>	<code>total_log_used</code> - Espacio utilizado de anotaciones cronológicas totales
<code>TOTAL_LOG_AVAILABLE</code>	<code>BIGINT</code>	<code>total_log_available</code> - Anotaciones cronológicas totales disponibles
<code>ROWS_READ</code>	<code>BIGINT</code>	<code>rows_read</code> - Filas leídas
<code>POOL_DATA_L_READS</code>	<code>BIGINT</code>	<code>pool_data_l_reads</code> - Lecturas lógicas de datos de agrupación de almacenamiento intermedio
<code>POOL_DATA_P_READS</code>	<code>BIGINT</code>	<code>pool_data_p_reads</code> - Lecturas físicas de datos de agrupación de almacenamiento intermedio
<code>POOL_DATA_WRITES</code>	<code>BIGINT</code>	<code>pool_data_writes</code> - Grabaciones de datos de agrupación de almacenamiento intermedio
<code>POOL_INDEX_L_READS</code>	<code>BIGINT</code>	<code>pool_index_l_reads</code> - Lecturas lógicas de índice de agrupación de almacenamiento intermedio
<code>POOL_INDEX_P_READS</code>	<code>BIGINT</code>	<code>pool_index_p_reads</code> - Lecturas físicas de índice de agrupación de almacenamiento intermedio
<code>POOL_INDEX_WRITES</code>	<code>BIGINT</code>	<code>pool_index_writes</code> - Grabaciones de índice de agrupación de almacenamiento intermedio
<code>POOL_READ_TIME</code>	<code>BIGINT</code>	<code>pool_read_time</code> - Tiempo total de lectura física de agrupación de almacenamiento intermedio
<code>POOL_WRITE_TIME</code>	<code>BIGINT</code>	<code>pool_write_time</code> - Tiempo total de grabación física de agrupación de almacenamiento intermedio
<code>POOL_ASYNC_INDEX_READS</code>	<code>BIGINT</code>	<code>pool_async_index_reads</code> - Lecturas de índice asíncronas de agrupación de almacenamiento intermedio

Tabla 230. Información devuelta por la función de tabla *SNAPSHOT\_DATABASE* (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento de supervisor correspondiente
POOL_DATA_TO_ESTORE	BIGINT	El elemento de supervisor de 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 de 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 de 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 de 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 almacenamiento intermedio
POOL_ASYNC_DATA_WRITES	BIGINT	pool_async_data_writes - Grabaciones de datos asíncronas de agrupación de almacenamiento intermedio
POOL_ASYNC_INDEX_WRITES	BIGINT	pool_async_index_writes - Grabaciones de índice asíncronas de agrupación de almacenamiento intermedio
POOL_ASYNC_READ_TIME	BIGINT	pool_async_read_time - Tiempo de lectura asíncrona de agrupación de almacenamiento intermedio
POOL_ASYNC_WRITE_TIME	BIGINT	pool_async_write_time - Tiempo de grabación asíncrona de agrupación de almacenamiento intermedio
POOL_ASYNC_DATA_READ_REQS	BIGINT	pool_async_data_read_reqs - Peticiones de lectura asíncrona de agrupación de almacenamiento intermedio
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 230. Información devuelta por la función de tabla  
SNAPSHOT\_DATABASE (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento de 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
FILES_CLOSED	BIGINT	files_closed - Archivos de base de datos cerrados
POOL_LSN_GAP_CLNS	BIGINT	pool_lsn_gap_clns - Limpiadores de espacio de anotaciones cronológicas de agrupación de almacenamiento intermedio activados
POOL_DRTY_PG_STEAL_CLNS	BIGINT	pool_drty_pg_steal_clns - Limpiadores de páginas víctimas de agrupación de almacenamiento intermedio activados
POOL_DRTY_PG_THRSH_CLNS	BIGINT	pool_drty_pg_thrsh_clns - Limpiadores de umbrales de agrupación de almacenamiento intermedio activados
LOCKS_HELD	BIGINT	locks_held - Bloqueos mantenidos
LOCK_WAITS	BIGINT	lock_waits - Esperas de bloqueo
LOCK_WAIT_TIME	BIGINT	lock_wait_time - Tiempo que se ha esperado en los bloqueos
LOCK_LIST_IN_USE	BIGINT	lock_list_in_use - Memoria de lista de bloqueos total en uso
DEADLOCKS	BIGINT	deadlocks - Puntos muertos detectados
LOCK_ESCALS	BIGINT	lock_escals - Número de escaladas de bloqueo
X_LOCK_ESCALS	BIGINT	x_lock_escals - Escaladas de bloqueo exclusivas
LOCKS_WAITING	BIGINT	locks_waiting - Agentes esperando en bloqueos
SORT_HEAP_ALLOCATED	BIGINT	sort_heap_allocated - Total de pila de clasificación asignada
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

Tabla 230. Información devuelta por la función de tabla  
SNAPSHOT\_DATABASE (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento de supervisor correspondiente
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 falladas
SELECT_SQL_STMTS	BIGINT	select_sql_stmts - Selección de sentencias de SQL ejecutada
DDL_SQL_STMTS	BIGINT	ddl_sql_stmts - Sentencias de SQL de lenguaje de definición de datos (DDL)
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 un 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
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

Tabla 230. Información devuelta por la función de tabla  
SNAPSHOT\_DATABASE (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento de supervisor correspondiente
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
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 durante el que se ha esperado la captación previa
APPL_SECTION_LOOKUPS	BIGINT	appl_section_lookups - Búsquedas de sección

Tabla 230. Información devuelta por la función de tabla  
SNAPSHOT\_DATABASE (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento de supervisor correspondiente
APPL_SECTION_INSERTS	BIGINT	appl_section_inserts - Inserciones de sección
TOTAL_HASH_JOINS	BIGINT	total_hash_joins - Uniones hash totales
TOTAL_HASH_LOOPS	BIGINT	total_hash_loops - Bucles hash totales
HASH_JOIN_OVERFLOW	BIGINT	hash_join_overflows - Desbordamientos de uniones hash
HASH_JOIN_SMALL_OVERFLOW	BIGINT	hash_join_small_overflows - Desbordamientos pequeños de uniones hash
PKG_CACHE_NUM_OVERFLOW	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 transacción más antigua
CATALOG_PARTITION_NAME	VARCHAR(128)	catalog_node_name - Nombre de red de nodo de catálogo
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

**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 787.

►►—SNAPSHOT\_DBM—(—númparticiónbd—)—————►►

El esquema es SYSPROC.

La función SNAPSHOT\_DBM devuelve información de una instantánea del gestor de bases de datos DB2.

*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. 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 231. Información devuelta por la función de tabla SNAPSHOT\_DBM

Nombre de columna	Tipo de datos	Descripción o elemento de 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 - Enviar clasificaciones de umbral
PIPED_SORTS_REQUESTED	BIGINT	piped_sorts_requested - Clasificaciones conducidas solicitada
PIPED_SORTS_ACCEPTED	BIGINT	piped_sorts_accepted - Clasificaciones conducidas aceptadas
REM_CONS_IN	BIGINT	rem_cons_in - Conexiones remotas al 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

Tabla 231. Información devuelta por la función de tabla *SNAPSHOT\_DBM* (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento de supervisor correspondiente
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 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 de respuesta del sistema principal
GW_CONS_WAIT_CLIENT	BIGINT	gw_cons_wait_client - Número de conexiones en espera de 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 bases de datos

Tabla 231. Información devuelta por la función de tabla *SNAPSHOT\_DBM* (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento de supervisor correspondiente
LAST_RESET	TIMESTAMP	last_reset - Indicación de fecha y hora de último restablecimiento

## SNAPSHOT\_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 801

►—SNAPSHOT\_DYN\_SQL—(—*nombrebd*—,—*númparticiónbd*—)—————►

El esquema es SYSPROC.

La función *SNAPSHOT\_DYN\_SQL* devuelve información de una instantánea de SQL dinámico. Sustituye la función *SQLCACHE\_SNAPSHOT*, que sigue estando disponible por razones de compatibilidad.

*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. 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 232. Información devuelta por la función de tabla *SNAPSHOT\_DYN\_SQL*

Nombre de columna	Tipo de datos	Descripción o elemento de supervisor correspondiente
SNAPSHOT_TIMESTAMP	TIMESTAMP	La fecha y la hora en que se tomó la instantánea.
ROWS_READ	BIGINT	rows_read - Filas leídas
ROWS_WRITTEN	BIGINT	rows_written - Filas grabadas
NUM_EXECUTIONS	BIGINT	num_executions - Ejecuciones de sentencia
NUM_COMPILATIONS	BIGINT	num_compilations - Compilaciones de sentencia

Tabla 232. Información devuelta por la función de tabla *SNAPSHOT\_DYN\_SQL* (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento de supervisor correspondiente
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
STMT_SORTS	BIGINT	stmt_sorts - Clasificaciones de sentencias
TOTAL_EXEC_TIME	BIGINT	total_exec_time - Tiempo de ejecución de sentencia transcurrido
TOTAL_SYS_CPU_TIME	BIGINT	total_sys_cpu_time - Total de CPU de sistema para una sentencia
TOTAL_USR_CPU_TIME	BIGINT	total_usr_cpu_time - Total de CPU de usuario para una sentencia
STMT_TEXT	CLOB(16M) <sup>1</sup>	stmt_text - Texto de sentencia SQL

<sup>1</sup> 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 415.

►►—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).

*núm-partición-bd*

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. Si se especifica el valor nulo, se establece -1 de forma implícita.



La función devuelve una tabla tal como se muestra más abajo.

*Tabla 233. 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 del supervisor ma_free está suspendido. El sistema devolverá un valor nulo para el elemento del supervisor suspendido.
MA_FREE_BOTTOM	BIGINT	El elemento del supervisor ma_free_bottom está suspendido. El sistema devolverá un valor nulo para el elemento del supervisor suspendido.
CE_FREE	BIGINT	El elemento del supervisor ce_free está suspendido. El sistema devolverá un valor nulo para el elemento del supervisor suspendido.
CE_FREE_BOTTOM	BIGINT	El elemento del supervisor ce_free_bottom está suspendido. El sistema devolverá un valor nulo para el elemento del supervisor suspendido.
RB_FREE	BIGINT	El elemento del supervisor rb_free está suspendido. El sistema devolverá un valor nulo para el elemento del supervisor suspendido.
RB_FREE_BOTTOM	BIGINT	El elemento del supervisor rb_free_bottom está suspendido. El sistema devolverá un valor nulo para el elemento del supervisor suspendido.
PARTITION_NUMBER	SMALLINT	node_number - Número de nodo

---

## SNAPSHOT\_FCMNODE

**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 417.

►►—SNAPSHOT\_FCMNODE—(—númparticiónbd—)—————►►

El esquema es SYSPROC.

La función SNAPSHOT\_FCMNODE devuelve información de una instantánea del gestor de comunicaciones rápidas que se encuentra en el gestor 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. 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 234. Información devuelta por la función de tabla SNAPSHOT\_FCMNODE

Nombre de columna	Tipo de datos	Descripción o elemento de supervisor correspondiente
SNAPSHOT_TIMESTAMP	TIMESTAMP	La fecha y la hora en que se tomó la instantánea.
CONNECTION_STATUS	BIGINT	connection_status - Estado de conexión
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
PARTITION_NUMBER	SMALLINT	node_number - Número de nodo

---

## SNAPSHOT\_FILEW

**Nota:** Este procedimiento ha quedado obsoleto y se ha sustituido por la “Procedimiento SNAP\_WRITE\_FILE” en la página 484.

►►—SNAPSHOT\_FILEW—(—tipo-petición—,—nombre-bd—,—núm-partición-bd—)—————►►

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`.

*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 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.

*núm-partición-bd*

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. 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

**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 423.

►►—SNAPSHOT\_LOCK—(—nombrebd—, —númparticiónbd—)—————►►

El esquema es `SYSPROC`.

La función `SNAPSHOT_LOCK` devuelve información de una instantánea de bloqueo.

*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. 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 235. Información devuelta por la función de tabla SNAPSHOT\_LOCK

Nombre de columna	Tipo de datos	Descripción o elemento de 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)
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 - Escalada 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 tabla

## SNAPSHOT\_LOCKWAIT

**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 429.

►►—SNAPSHOT\_LOCKWAIT—(—nombrebd—, —númparticiónbd—)—————►►

El esquema es SYSPROC.

La función `SNAPSHOT_LOCKWAIT` devuelve información sobre las esperas de bloqueo de una instantánea de aplicación.

*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. 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 236. Información devuelta por la función de tabla `SNAPSHOT_LOCKWAIT`*

Nombre de columna	Tipo de datos	Descripción o elemento de supervisor correspondiente
<code>SNAPSHOT_TIMESTAMP</code>	<code>TIMESTAMP</code>	La fecha y la hora en que se tomó la instantánea.
<code>AGENT_ID</code>	<code>BIGINT</code>	<code>agent_id</code> - Descriptor de contexto de aplicación (ID de agente)
<code>SUBSECTION_NUMBER</code>	<code>BIGINT</code>	<code>ss_number</code> - Número de subsección
<code>LOCK_MODE</code>	<code>BIGINT</code>	<code>lock_mode</code> - Modalidad de bloqueo
<code>LOCK_OBJECT_TYPE</code>	<code>BIGINT</code>	<code>lock_object_type</code> - Tipo de objeto de bloqueo esperado
<code>AGENT_ID_HOLDING_LK</code>	<code>BIGINT</code>	<code>agent_id_holding_lock</code> - ID de agente que mantiene el bloqueo
<code>LOCK_WAIT_START_TIME</code>	<code>TIMESTAMP</code>	<code>lock_wait_start_time</code> - Indicación de fecha y hora de inicio de espera de bloqueo
<code>LOCK_MODE_REQUESTED</code>	<code>BIGINT</code>	<code>lock_mode_requested</code> - Modalidad de bloqueo solicitada
<code>PARTITION_NUMBER</code>	<code>SMALLINT</code>	<code>node_number</code> - Número de nodo
<code>LOCK_ESCALATION</code>	<code>SMALLINT</code>	<code>lock_escalation</code> - Escalada de bloqueos
<code>TABLE_NAME</code>	<code>VARCHAR(128)</code>	<code>table_name</code> - Nombre de tabla

Tabla 236. Información devuelta por la función de tabla *SNAPSHOT\_LOCKWAIT* (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento de supervisor correspondiente
TABLE_SCHEMA	VARCHAR(128)	table_schema - Nombre de esquema de tabla
TABLESPACE_NAME	VARCHAR(128)	tablespace_name - Nombre de espacio de tabla
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 471.

►► *SNAPSHOT\_QUIESCERS* (—*nombre-bd*—, —*núm-partición-bd*—) ◀◀

El esquema es *SYSPROC*.

La función *SNAPSHOT\_QUIESCERS* devuelve información sobre inmovilizadores desde una instantánea de espacio de tabla.

*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.

*núm-partición-bd*

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. Si se especifica el valor nulo, se establece -1 de forma implícita.

La función devuelve una tabla tal como se muestra más abajo.

Tabla 237. 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 del espacio de tablas de inmovilizador

Tabla 237. Información devuelta por la función de tabla *SNAPSHOT QUIESCERS* (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
QUIESCER_OBJ_ID	BIGINT	quiescer_obj_id - Identificación del 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 474.

►►—SNAPSHOT\_RANGES—(—nombre-bd—,—núm-partición-bd—)—————►►

El esquema es SYSPROC.

La función SNAPSHOT\_RANGES devuelve información de una instantánea de rango.

*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.

*núm-partición-bd*

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. Si se especifica el valor nulo, se establece -1 de forma implícita.

La función devuelve una tabla tal como se muestra más abajo.

Tabla 238. 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 del espacio de tablas

Tabla 238. Información devuelta por la función de tabla *SNAPSHOT\_RANGES* (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
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
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

**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 435.

►►—SNAPSHOT\_STATEMENT—(—*nombrebd*—,—*númparticiónbd*—)—◄◄

El esquema es SYSPROC.

La función SNAPSHOT\_STATEMENT devuelve información sobre las sentencias de una instantánea de aplicación.

*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. 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 239. Información devuelta por la función de tabla SNAPSHOT\_STATEMENT*

Nombre de columna	Tipo de datos	Descripción o elemento de 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)
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 - Consultar estimación de coste
QUERY_CARD_ESTIMATE	BIGINT	query_card_estimate - Consultar estimación 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 capturas satisfactorias
STMT_START	TIMESTAMP	stmt_start - Indicación de fecha y hora de inicio de operación de sentencia

Tabla 239. Información devuelta por la función de tabla *SNAPSHOT\_STATEMENT* (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento de supervisor correspondiente
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_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) <sup>1</sup>	stmt_text - Texto de sentencia SQL

<sup>1</sup> 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

**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 444.

►►—SNAPSHOT\_SUBSECT—(—nombrebd—,—númparticiónbd—)—————►►

El esquema es SYSPROC.

La función SNAPSHOT\_SUBSECT devuelve información sobre las subsecciones de los planes de acceso 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. 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 240. Información devuelta por la función de tabla SNAPSHOT\_SUBSECT*

Nombre de columna	Tipo de datos	Descripción o elemento de supervisor correspondiente
SNAPSHOT_TIMESTAMP	TIMESTAMP	La fecha y la hora en que se tomó la instantánea.
STMT_TEXT	CLOB(16M) <sup>1</sup>	stmt_text - Texto de sentencia 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	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

Tabla 240. Información devuelta por la función de tabla *SNAPSHOT\_SUBSECT* (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento de supervisor correspondiente
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
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

<sup>1</sup> 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

**Nota:** Esta función de tabla ha quedado obsoleta y se ha sustituido por la “Vista administrativa SNAPSWITCHES y función de tabla SNAP\_GET\_SWITCHES – Recuperar información acerca del estado del conmutador de instantánea de base de datos” en la página 448.

►—SNAPSHOT\_SWITCHES—(—núm-partición-bd—)—————►

El esquema es SYSPROC.

La función SNAPSHOT\_SWITCHES devuelve información acerca del estado de conmutación de las instantáneas de la base de datos.

*núm-partición-bd*

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. Si se especifica el valor nulo, se establece -1 de forma implícita.

La función devuelve una tabla tal como se muestra más abajo.

Tabla 241. 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 241. 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

**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 451

►►—SNAPSHOT\_TABLE—(—nombrebd—, —númparticiónbd—)—————►►

El esquema es SYSPROC.

La función SNAPSHOT\_TABLE devuelve información sobre las actividades de una instantánea 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 todas las particiones de base de datos. 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 242. Información devuelta por la función de tabla SNAPSHOT\_TABLE

Nombre de columna	Tipo de datos	Descripción o elemento de 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 454.

►—SNAPSHOT\_TBREORG—(—nombre-bd—,—núm-partición-bd—)—————►

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.

*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.

*núm-partición-bd*

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. 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 243. 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 máxima de reorganización de tabla
REORG_CURRENT_COUNTER	BIGINT	reorg_current_counter - Progreso de reorganización de tabla

Tabla 243. Información devuelta por la función de tabla *SNAPSHOT\_TBREORG* (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento del supervisor correspondiente
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 que se utiliza para reorganizar la tabla
REORG_TBSPC_ID	BIGINT	reorg_tbspc_id - Espacio de tabla donde se reorganiza la tabla
PARTITION_NUMBER	SMALLINT	node_number - Número de nodo

## SNAPSHOT\_TBS

**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 459

►►—SNAPSHOT\_TBS—(—nombrebd—,—númparticiónbd—)—————►►

El esquema es SYSPROC.

La función SNAPSHOT\_TBS devuelve información sobre las actividades de una instantánea de espacio 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 todas las particiones de base de datos. 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 244. Información devuelta por la función de tabla SNAPSHOT\_TBS*

Nombre de columna	Tipo de datos	Descripción o elemento de 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 almacenamiento intermedio
POOL_DATA_P_READS	BIGINT	pool_data_p_reads - Lecturas físicas de datos de agrupación de almacenamiento intermedio
POOL_ASYNC_DATA_READS	BIGINT	pool_async_data_reads - Lecturas de datos asíncronas de agrupación de almacenamiento intermedio
POOL_DATA_WRITES	BIGINT	pool_data_writes - Grabaciones de datos de agrupación de almacenamiento intermedio
POOL_ASYNC_DATA_WRITES	BIGINT	pool_async_data_writes - Grabaciones de datos asíncronas de agrupación de almacenamiento intermedio
POOL_INDEX_L_READS	BIGINT	pool_index_l_reads - Lecturas lógicas de índice de agrupación de almacenamiento intermedio
POOL_INDEX_P_READS	BIGINT	pool_index_p_reads - Lecturas físicas de índice de agrupación de almacenamiento intermedio
POOL_INDEX_WRITES	BIGINT	pool_index_writes - Grabaciones de índice de agrupación de almacenamiento intermedio
POOL_ASYNC_INDEX_WRITES	BIGINT	pool_async_index_writes - Grabaciones de índice asíncronas de agrupación de almacenamiento intermedio
POOL_READ_TIME	BIGINT	pool_read_time - Tiempo total de lectura física de agrupación de almacenamiento intermedio
POOL_WRITE_TIME	BIGINT	pool_write_time - Tiempo total de grabación física de agrupación de almacenamiento intermedio
POOL_ASYNC_READ_TIME	BIGINT	pool_async_read_time - Tiempo de lectura asíncrona de agrupación de almacenamiento intermedio

Tabla 244. Información devuelta por la función de tabla *SNAPSHOT\_TBS* (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento de supervisor correspondiente
POOL_ASYNC_WRITE_TIME	BIGINT	pool_async_write_time - Tiempo de grabación asíncrona de agrupación de almacenamiento intermedio
POOL_ASYNC_DATA_READ_REQS	BIGINT	pool_async_data_read_reqs - Peticiones de lectura asíncrona de agrupación de almacenamiento intermedio
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 almacenamiento intermedio
POOL_DATA_TO_ESTORE	BIGINT	El elemento de supervisor de 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 de 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 de 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 de 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.
FILES_CLOSED	BIGINT	files_closed - Archivos de base de datos cerrados

Tabla 244. Información devuelta por la función de tabla *SNAPSHOT\_TBS* (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento de supervisor correspondiente
TABLESPACE_NAME	VARCHAR(128)	tablespace_name - Nombre de espacio de tabla

## 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 466

►—SNAPSHOT\_TBS\_CFG—(—*nombrebd*—, —*númparticiónbd*—)—————►

El esquema es SYSPROC.

La función *SNAPSHOT\_TBS\_CFG* devuelve información sobre la configuración de una instantánea de espacio 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 todas las particiones de base de datos. 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 245. Información devuelta por la función de tabla *SNAPSHOT\_TBS\_CFG*

Nombre de columna	Tipo de datos	Descripción o elemento de 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 tabla
TABLESPACE_NAME	VARCHAR (128)	tablespace_name - Nombre de espacio de tabla
TABLESPACE_TYPE	SMALLINT	tablespace_type - Tipo de espacio de tabla
TABLESPACE_STATE	BIGINT	tablespace_state - Estado de espacio de tabla

Tabla 245. Información devuelta por la función de tabla  
SNAPSHOT\_TBS\_CFG (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento de supervisor correspondiente
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 tabla 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 tabla
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 tabla
EXTENT_SIZE	BIGINT	tablespace_extent_size - Tamaño de extensión de espacio de tabla
PREFETCH_SIZE	BIGINT	tablespace_prefetch_size - Tamaño de captación previa de espacio de tabla
TOTAL_PAGES	BIGINT	tablespace_total_pages - Total de páginas en espacio de tabla
USABLE_PAGES	BIGINT	tablespace_usable_pages - Páginas utilizables en espacio de tabla
USED_PAGES	BIGINT	tablespace_used_pages - Páginas utilizadas en espacio de tabla
FREE_PAGES	BIGINT	tablespace_free_pages - Páginas libres en espacio de tabla
PENDING_FREE_PAGES	BIGINT	tablespace_pending_free_pages - Páginas libres pendientes en espacio de tabla
HIGH_WATER_MARK	BIGINT	pool_watermark - Marca de límite de agrupación de memoria
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 actual de reequilibrador

Tabla 245. Información devuelta por la función de tabla *SNAPSHOT\_TBS\_CFG* (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento de 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
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 tabla
NUM_CONTAINERS	BIGINT	tablespace_num_containers - Número de contenedores del espacio de tabla

## 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 801

►►—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 246. Información devuelta por la función de tabla *SQLCACHE\_SNAPSHOT*

Nombre de columna	Tipo de datos	Descripción o elemento de supervisor correspondiente
NUM_EXECUTIONS	INTEGER	num_executions - Ejecuciones de sentencia
NUM_COMPILATIONS	INTEGER	num_compilations - Compilaciones de sentencia
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

Tabla 246. Información devuelta por la función de tabla *SQLCACHE\_SNAPSHOT* (continuación)

Nombre de columna	Tipo de datos	Descripción o elemento de supervisor correspondiente
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 de ejecución de sentencia transcurrido
TOTAL_EXEC_TIME_MS	INTEGER	total_exec_time - Tiempo de ejecución de sentencia transcurrido
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) <sup>1</sup>	stmt_text - Texto de sentencia SQL

<sup>1</sup> 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.

---

## Parte 2. Apéndices





---

## 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 en 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](http://ibm.com).

Puede acceder a información técnica adicional de DB2 como, por ejemplo, notas técnicas, documentos técnicos y publicaciones en línea IBM Redbooks en [ibm.com](http://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](mailto:db2docs@ca.ibm.com). El personal encargado de la documentación de DB2 lee todos los comentarios de los usuarios, pero no puede responder directamente a cada uno. 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](http://www.ibm.com/shop/publications/order). Los manuales de DB2 Versión 9.5 en inglés en formato PDF y las versiones traducidas se pueden descargar del sitio [www.ibm.com/support/docview.wss?rs=71&uid=swg2700947](http://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.

Tabla 247. Información técnica de DB2

Nombre	Número de documento	Copia impresa disponible
<i>Consulta de las API administrativas</i>	SC11-3505-00	Sí
<i>Rutinas y vistas administrativas</i>	SC11-3507-00	No
<i>Call Level Interface Guide and Reference, Volume 1</i>	SC23-5844-00	Sí
<i>Call Level Interface Guide and Reference, Volume 2</i>	SC23-5845-00	Sí
<i>Consulta de mandatos</i>	SC11-3506-00	Sí
<i>Data Movement Utilities Guide and Reference</i>	SC23-5847-00	Sí
<i>Data Recovery and High Availability Guide and Reference</i>	SC23-5848-00	Sí
<i>Data Servers, Databases, and Database Objects Guide</i>	SC23-5849-00	Sí
<i>Database Security Guide</i>	SC23-5850-00	Sí
<i>Desarrollo de aplicaciones ADO.NET y OLE DB</i>	SC11-3499-00	Sí
<i>Desarrollo de aplicaciones de SQL incorporado</i>	SC11-3500-00	Sí
<i>Desarrollo de aplicaciones Java</i>	SC11-3501-00	Sí
<i>Desarrollo de aplicaciones Perl y PHP</i>	SC11-3502-00	No
<i>Desarrollo de rutinas definidas por el usuario (SQL y externas)</i>	SC11-3503-00	Sí
<i>Iniciación al desarrollo de aplicaciones de bases de datos</i>	GC11-3504-00	Sí
<i>Iniciación a la instalación y administración de DB2 en Linux y Windows</i>	GC11-3511-00	Sí
<i>Internationalization Guide</i>	SC23-5858-00	Sí
<i>Consulta de mensajes, Volumen 1</i>	GI11-7823-00	No
<i>Consulta de mensajes, Volumen 2</i>	GI11-7824-00	No
<i>Guía de migración</i>	GC11-3510-00	Sí

Tabla 247. Información técnica de DB2 (continuación)

Nombre	Número de documento	Copia impresa disponible
Net Search Extender Guía de administración y del usuario <b>Nota:</b> El contenido de este documento no está incluido en el Centro de información de DB2	SC11-3615-00	Sí
Partitioning and Clustering Guide	SC23-5860-00	Sí
Query Patroller Administration and User's Guide	SC23-8507-00	Sí
Guía rápida de iniciación para clientes IBM Data Server	GC11-3513-00	No
Guía rápida de iniciación para servidores DB2	GC11-3512-00	Sí
Spatial Extender and Geodetic Data Management Feature Guía del usuario y manual de consulta	SC11-3614-00	Sí
Consulta de SQL, Volumen 1	SC11-3508-00	Sí
Consulta de SQL, Volumen 2	SC11-3509-00	Sí
System Monitor Guide and Reference	SC23-5865-00	Sí
Text Search Guide	SC23-5866-00	Sí
Troubleshooting Guide	GI11-7857-00	No
Tuning Database Performance	SC23-5867-00	Sí
Guía de aprendizaje de Visual Explain	SC11-3518-00	No
Novedades	SC11-3517-00	Sí
Workload Manager Guide and Reference	SC23-5870-00	Sí
pureXML Guide	SC23-5871-00	Sí
XQuery Reference	SC23-5872-00	No

Tabla 248. Información técnica específica de DB2 Connect

Nombre	Número de documento	Copia impresa disponible
Guía rápida de iniciación para DB2 Connect Personal Edition	GC11-3515-00	Sí
Guía rápida de iniciación para servidores DB2 Connect	GC11-3516-00	Sí
Guía del usuario de DB2 Connect	SC11-3514-00	Sí

Tabla 249. Información técnica de Information Integration

Nombre	Número de documento	Copia impresa disponible
Information Integration: Administration Guide for Federated Systems	SC19-1020-01	Sí

Tabla 249. Información técnica de Information Integration (continuación)

Nombre	Número de documento	Copia impresa disponible
<i>Information Integration: ASNCLP Program Reference for Replication and Event Publishing</i>	SC19-1018-02	Sí
<i>Information Integration: Configuration Guide for Federated Data Sources</i>	SC19-1034-01	No
<i>Information Integration: SQL Replication Guide and Reference</i>	SC19-1030-01	Sí
<i>Information Integration: Introduction to Replication and Event Publishing</i>	SC19-1028-01	Sí

## Pedido de manuales de DB2 en copia impresa

Si necesita manuales de DB2 en copia impresa, puede comprarlos en línea en varios, pero no en todos los países o regiones. 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/v9r5>.

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](http://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 pedido, consulte el apartado “Biblioteca técnica de DB2 en copia impresa o en formato PDF” en la página 864.

---

## Visualización de la ayuda para estados de SQL desde el procesador de línea de mandatos

DB2 devuelve 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 invocar 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 9.5 de DB2, el URL del Centro de información de DB2 es <http://publib.boulder.ibm.com/infocenter/db2luw/v9r5/>

Para los temas de la versión 9 de DB2, el URL del Centro de información de DB2 es <http://publib.boulder.ibm.com/infocenter/db2luw/v9/>

Para los temas de la versión 8 de DB2, vaya al URL del Centro de información de la versión 8 en: <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, puede que también tenga que 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

Si ha instalado el Centro de información de DB2, puede descargar e instalar las actualizaciones que IBM pueda tener disponibles.

Para actualizar 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 descargar y aplicar actualizaciones.
2. Utilice la función Actualizar para ver qué actualizaciones están disponibles. Si hay actualizaciones que quisiera instalar, puede utilizar la función Actualizar para descargarlas y actualizarlas.

**Nota:** Si su entorno requiere la instalación de actualizaciones del Centro de información de DB2 en una máquina no conectada a Internet, debe duplicar 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 descargar 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 Vista, los mandatos listados más abajo se deben ejecutar como administrador. Para iniciar 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 en **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/db2icdv95 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. De forma predeterminada, el Centro de información de DB2 se instala en el directorio <Archivos de programa>\IBM\Centro de información de DB2\Versión 9.5, 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. De forma predeterminada, el Centro de información de DB2 se instala en el directorio /opt/ibm/db2ic/V9.5.
    - b. Navegue desde el directorio de instalación al directorio doc/bin.
    - c. Ejecute el script help\_start:  
`help_start`

Se inicia el navegador Web por omisión de los sistemas para visualizar el Centro de información autónomo.

3. Pulse en el botón Actualizar (🔄). 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 descarga, compruebe las selecciones que desea descargar, después pulse en Instalar actualizaciones.
5. Cuando finalice el proceso de descarga e instalación, pulse en Finalizar.
6. Detenga el Centro de información autónomo.
  - En Windows, navegue hasta el directorio doc\bin y ejecute el archivo help\_end.bat:  
`help_end.bat`

**Nota:** El archivo help\_end de proceso por lotes contiene los mandatos necesarios para concluir 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 concluir 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 concluir sin peligro los procesos que se iniciaron mediante el script help\_start. No utilice ningún otro método para concluir el script help\_start.

7. Reinicie el Centro de información de DB2:
  - En Windows, pulse en **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/db2icdv95 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 *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 DB2.

### Documentación de DB2

Puede encontrar información sobre la resolución de problemas en la publicación DB2 Troubleshooting Guide o en la sección Soporte y resolución de problemas 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), fix packs 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/udb/support.html>

---

## 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

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 doble byte (DBCS), puede ponerse en contacto con el Departamento de Propiedad Intelectual de IBM de su país/región o escribir a:

IBM World Trade Asia Corporation  
Licensing  
2-31 Roppongi 3-chome, Minato-ku  
Tokio 106, 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.

Este documento puede proporcionar enlaces o referencias a sitios y recursos que no son de IBM. IBM no representa, no da garantías, ni se compromete con los recursos de terceros ni con los recursos que no son de IBM a los cuales se puede hacer referencia, acceder desde o enlazarse con desde este documento. Un enlace a un sitio que no es de IBM no implica que IBM apruebe el contenido o la utilización de dicho sitio Web o a su propietario. Además, IBM no forma parte ni es responsable de ninguna transacción que el usuario pueda realizar con terceros, aún cuando llegue a conocerlos (o utilice un enlace a ellas) desde un sitio de IBM. De acuerdo a esto, el usuario reconoce y acepta que IBM no es responsable de la disponibilidad de dichos recursos o sitios externos ni tampoco es responsable de ningún contenido, servicio, producto u otros materiales que estén o se encuentren disponibles desde dichos sitios o recursos. Cualquier software que proporcionen terceras partes, estarán sujetos a los términos y condiciones de licencia que acompañen al software.

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 licenciarios 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 puede contener 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.

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**

Los nombres de empresas, productos o servicios identificados en los documentos de la biblioteca de documentación de DB2 Versión 9.5 pueden ser marcas registradas o marcas de servicio de International Business Machines Corporation o de otras empresas. La información sobre marcas registradas de IBM Corporation en los Estados Unidos y/o en otros países está ubicada en <http://www.ibm.com/legal/copytrade.shtml>.

Los términos siguientes son marcas registradas de otras empresas y se han utilizado como mínimo en uno de los documentos de la biblioteca de documentación de DB2:

Microsoft, Windows, Windows NT y el logotipo de Windows son marcas registradas de Microsoft Corporation en los Estados Unidos y/o en otros países.

Intel, el logotipo de Intel, 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 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.

Linux es una marca registrada de Linus Torvalds en los Estados Unidos y/o en otros países.

Adobe, el logotipo de Adobe, PostScript y el logotipo de PostScript son marcas registradas o marcas comerciales de Adobe Systems Incorporated 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 868  
  Centro de información de DB2 868  
ADMIN\_CMD, procedimiento 36  
  eliminación de mensajes 219  
  mandatos soportados  
    ADD CONTACT 38  
    ADD CONTACTGROUP 39  
    AUTOCONFIGURE 40  
    BACKUP DATABASE 45  
    DESCRIBE 53  
    DROP CONTACT 63  
    DROP CONTACTGROUP 64  
    EXPORT 65  
    FORCE APPLICATION 77  
    GET STMM TUNING DBPARTITIONNUM 78  
    IMPORT 80  
    INITIALIZE TAPE 107  
    LOAD 108  
    PRUNE HISTORY/LOGFILE 149  
    QUIESCE DATABASE 151  
    QUIESCE TABLESPACES FOR TABLE 153  
    REDISTRIBUTE DATABASE PARTITION GROUP 156  
    REORG INDEXES/TABLE 164  
    RESET ALERT CONFIGURATION 175  
    RESET DATABASE CONFIGURATION 176  
    RESET DATABASE MANAGER  
      CONFIGURATION 178  
    REWIND TAPE 180  
    RUNSTATS 180  
    SET TAPE POSITION 192  
    UNQUIESCE DATABASE 193  
    UPDATE ALERT CONFIGURATION 194  
    UPDATE CONTACT 200  
    UPDATE CONTACTGROUP 201  
    UPDATE DATABASE CONFIGURATION 202  
    UPDATE DATABASE MANAGER  
      CONFIGURATION 205  
    UPDATE HEALTH NOTIFICATION CONTACT  
      LIST 207  
    UPDATE HISTORY 208  
    UPDATE STMM TUNING DBPARTITIONNUM 211  
  recuperación de mensajes 213  
ADMIN\_COPY\_SCHEMA, procedimiento 687  
ADMIN\_DROP\_SCHEMA, procedimiento 692  
admin\_get\_dbp\_mem\_usage  
  función de tabla 212  
ADMIN\_GET\_MSGS, función de tabla 213  
ADMIN\_GET\_TAB\_INFO, función de tabla 747  
ADMIN\_REMOVE\_MSGS, procedimiento 219  
ADMINTABCOMPRESSINFO 215  
ADMINTABINFO, vista administrativa 220  
alertas de salud  
  recuperación de configuración de acciones de alertas 275  
  recuperación de configuración de alertas 278  
ALTOBJ, procedimiento 694  
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  
APPL\_PERFORMANCE, vista administrativa 322  
APPLICATION\_ID, función escalar 696  
APPLICATIONS, vista administrativa 323  
archivo histórico  
  recuperar información 701  
AUDIT\_ARCHIVE 228  
AUDIT\_DELIM\_EXTRACT 229  
AUDIT\_LIST\_LOGS 231  
AUTH\_LIST\_GROUPS\_FOR\_AUTHID, función de tabla 315  
AUTH\_LIST\_ROLES\_FOR\_AUTHID 316  
AUTHORIZATIONIDS, vista administrativa 318  
autorizaciones  
  para vistas administrativas 3  
  recuperación de ID de autorización 318  
  recuperación de pertenencia a grupo 315  
avisos 873  
ayuda  
  para sentencias de SQL 867  
  visualización 867

## B

BP\_HITRATIO, vista administrativa 327  
BP\_READ\_IO, vista administrativa 329  
BP\_WRITE\_IO, vista administrativa 331

## C

Centro de información  
  actualización 868  
  versiones 867  
  visualización en distintos idiomas 867  
Centro de información de DB2  
  actualización 868  
  versiones 867  
  visualización en distintos idiomas 867  
COMPILATION\_ENV, función de tabla 697  
configuración de base de datos  
  recuperación de parámetros 237  
configuración del gestor de bases de datos  
  recuperación de parámetros 239  
CONTACTGROUPS, vista administrativa 699  
contactos  
  recuperación de grupos de contactos 699  
  recuperación de listas de contactos 700  
CONTACTS, vista administrativa 700  
CONTAINER\_UTILIZATION, vista administrativa 333  
copia  
  esquemas y objetos 687

## D

DB\_HISTORY, vista administrativa 701  
DB\_PARTITIONS, función de tabla 236  
DBCFCG, vista administrativa 237  
DBMCFG, vista administrativa 239

DBPATHS, vista administrativa 706  
 descartar  
   esquemas y objetos 692  
 determinación de problemas  
   guías de aprendizaje 870  
   información en línea 870  
   recuperación mensajes de anotación cronológica de  
     notificación de DB2 727  
 dividir duplicación  
   recuperación de vías de acceso de base de datos 706  
 documentación  
   PDF o impresa 864  
   términos y condiciones de uso 871

## E

ENV\_INST\_INFO, vista administrativa 243  
 ENV\_PROD\_INFO, vista administrativa 245  
 ENV\_SYS\_INFO, vista administrativa 247  
 esquemas  
   copia de esquemas y objetos 687  
   eliminación de esquemas y objetos 692  
 EXPLAIN\_GET\_MSGS, función de tabla 714

## F

Función de tabla ADMIN\_GET\_TAB\_COMPRESS\_INFO 215  
 función de tabla ADMIN\_GET\_TAB\_INFO\_V95 220  
 Función de tabla  
   AUTH\_LIST\_AUTHORITIES\_FOR\_AUTHID 311  
 función de tabla PD\_GET\_DIAG\_HIST 720  
 función de tabla SNAP\_GET\_APPL\_INFO\_V95 352, 492  
 función de tabla SNAP\_GET\_APPL\_V95 360, 499  
 función de tabla SNAP\_GET\_BP\_V95 369, 508  
 función de tabla SNAP\_GET\_DB\_V95 382, 521  
 función de tabla SNAP\_GET\_DBM\_V95 399, 538  
 función de tabla SNAP\_GET\_DYN\_SQL\_V95 409, 548  
 función de tabla SNAP\_GET\_STORAGE\_PATHS 442, 581  
 función de tabla WLM\_GET\_ACTIVITY\_DETAILS 650  
 función de tabla WLM\_GET\_QUEUE\_STATS 657  
 función de tabla WLM\_GET\_SERVICE\_CLASS\_AGENTS 661  
 función de tabla  
   WLM\_GET\_SERVICE\_CLASS\_WORKLOAD\_OCCURRENCES 666  
 función de tabla  
   WLM\_GET\_SERVICE\_SUBCLASS\_STATS 671  
 función de tabla  
   WLM\_GET\_SERVICE\_SUPERCLASS\_STATS 676  
 función de tabla  
   WLM\_GET\_WORK\_ACTION\_SET\_STATS 678  
 función de tabla  
   WLM\_GET\_WORKLOAD\_OCCURRENCE\_ACTIVITIES 680  
 función de tabla WLM\_GET\_WORKLOAD\_STATS 684  
 Función escalar EXPLAIN\_FORMAT\_STATS 710  
 funciones  
   AUDIT\_LIST\_LOGS 231  
   escalar  
     APPLICATION\_ID 696  
     GET\_ROUTINE\_OPTS 629  
   escalares  
     MQPUBLISH 293  
     MQREAD 294  
     MQREADCLOB 299  
     MQRECEIVE 300  
     MQRECEIVECLOB 306  
     MQSEND 307  
     MQSUBSCRIBE 309

funciones (*continuación*)  
 escalares (*continuación*)  
   MQUNSUBSCRIBE 310  
   SQLERRM 738  
 funciones de tabla 1  
   ADMIN\_GET\_MSGS 213  
   ADMIN\_GET\_TAB\_COMPRESS\_INFO 215  
   ADMIN\_GET\_TAB\_INFO 747  
   ADMIN\_GET\_TAB\_INFO\_V95 220  
   AM\_BASE\_RPT\_RECOMS 21  
   AM\_BASE\_RPTS 22  
   AUDIT\_ARCHIVE 228  
   AUTH\_LIST\_AUTHORITIES\_FOR\_AUTHID 311  
   AUTH\_LIST\_GROUPS\_FOR\_AUTHID 315  
   comparadas con las vistas administrativas 5  
   COMPILATION\_ENV 697  
   DB\_PARTITIONS 236  
   EXPLAIN\_GET\_MSGS 714  
   GET\_DB\_CONFIG 757  
   GET\_DBM\_CONFIG 758  
   HEALTH\_CONT\_HI 252  
   HEALTH\_CONT\_HI\_HIS 254  
   HEALTH\_CONT\_INFO 256  
   HEALTH\_DB\_HI 257  
   HEALTH\_DB\_HI\_HIS 261  
   HEALTH\_DB\_HIC 264  
   HEALTH\_DB\_HIC\_HIS 266  
   HEALTH\_DB\_INFO 269  
   HEALTH\_DBM\_HI 270  
   HEALTH\_DBM\_HI\_HIS 271  
   HEALTH\_DBM\_INFO 274  
   HEALTH\_GET\_ALERT\_ACTION\_CFG 275  
   HEALTH\_GET\_ALERT\_CFG 278  
   HEALTH\_GET\_IND\_DEFINITION 281  
   HEALTH\_TBS\_HI 285  
   HEALTH\_TBS\_HI\_HIS 287  
   HEALTH\_TBS\_INFO 291  
   MQREADALL 295  
   MQREADALLCLOB 297  
   MQRECEIVEALL 302  
   MQRECEIVEALLCLOB 304  
   obsoletas 743  
   PD\_GET\_DIAG\_HIST 720  
   PD\_GET\_LOG\_MSGS 727  
   SNAP\_GET\_AGENT 346, 485  
   SNAP\_GET\_AGENT\_MEMORY\_POOL 349, 488  
   SNAP\_GET\_APPL 759  
   SNAP\_GET\_APPL\_INFO 767  
   SNAP\_GET\_APPL\_INFO\_V95 352, 492  
   SNAP\_GET\_APPL\_V95 360, 499  
   SNAP\_GET\_BP 773  
   SNAP\_GET\_BP\_PART 374, 513  
   SNAP\_GET\_BP\_V95 369, 508  
   SNAP\_GET\_CONTAINER (obsoleta) 778  
   SNAP\_GET\_CONTAINER\_V91 378, 517  
   SNAP\_GET\_DB (obsoleta) 779  
   SNAP\_GET\_DB\_MEMORY\_POOL 395, 534  
   SNAP\_GET\_DB\_V91 790  
   SNAP\_GET\_DB\_V95 382, 521  
   SNAP\_GET\_DBM 787  
   SNAP\_GET\_DBM\_MEMORY\_POOL 403, 542  
   SNAP\_GET\_DBM\_V95 399, 538  
   SNAP\_GET\_DETAIL\_LOG\_V91 406, 545  
   SNAP\_GET\_DYN\_SQL (obsoleta) 805  
   SNAP\_GET\_DYN\_SQL\_V91 801  
   SNAP\_GET\_DYN\_SQL\_V95 409, 548  
   SNAP\_GET\_FCM 415, 554



## funciones (continuación)

### funciones de tabla (continuación)

SNAP\_GET\_FCM\_PART 417, 556  
SNAP\_GET\_HADR 419, 558  
SNAP\_GET\_LOCK 423, 562  
SNAP\_GET\_LOCKWAIT 429, 568  
SNAP\_GET\_STMT 435, 574  
SNAP\_GET\_STO\_PATHS (obsoleta) 807  
SNAP\_GET\_STORAGE\_PATHS 442, 581  
SNAP\_GET\_SUBSECTION 444, 583  
SNAP\_GET\_SWITCHES 448, 587  
SNAP\_GET\_TAB (obsoleta) 808  
SNAP\_GET\_TAB\_REORG 454, 593  
SNAP\_GET\_TAB\_V91 451, 590  
SNAP\_GET\_TBSP (obsoleta) 809  
SNAP\_GET\_TBSP\_PART (obsoleta) 813  
SNAP\_GET\_TBSP\_PART\_V91 466, 605  
SNAP\_GET\_TBSP QUIESCER 471, 610  
SNAP\_GET\_TBSP\_RANGE 474, 613  
SNAP\_GET\_TBSP\_V91 459, 598  
SNAP\_GET\_UTIL 478, 617  
SNAP\_GET\_UTIL\_PROGRESS 481, 620  
SNAPSHOT\_AGENT (obsoleta) 815  
SNAPSHOT\_APPL (obsoleta) 816  
SNAPSHOT\_APPL\_INFO (obsoleta) 822  
SNAPSHOT\_BP (obsoleta) 824  
SNAPSHOT\_CONTAINER (obsoleta) 827  
SNAPSHOT\_DATABASE (obsoleta) 828  
SNAPSHOT\_DBM (obsoleta) 835  
SNAPSHOT\_DYN\_SQL (obsoleta) 837  
SNAPSHOT\_FCM (obsoleta) 838  
SNAPSHOT\_FCMNODE (obsoleta) 840  
SNAPSHOT\_LOCK (obsoleta) 841  
SNAPSHOT\_LOCKWAIT (obsoleta) 842  
SNAPSHOT QUIESCERS (obsoleta) 844  
SNAPSHOT\_RANGES (obsoleta) 845  
SNAPSHOT\_STATEMENT (obsoleta) 846  
SNAPSHOT\_SUBSECT (obsoleta) 848  
SNAPSHOT\_SWITCHES (obsoleta) 850  
SNAPSHOT\_TABLE (obsoleta) 852  
SNAPSHOT\_TBREORG (obsoleta) 853  
SNAPSHOT\_TBS (obsoleta) 854  
SNAPSHOT\_TBS\_CFG (obsoleta) 857  
soportadas 7  
SQLCACHE\_SNAPSHOT (obsoleta) 859  
WLM\_GET\_ACTIVITY\_DETAILS 650  
WLM\_GET\_QUEUE\_STATS 657  
WLM\_GET\_SERVICE\_CLASS\_AGENTS 661  
WLM\_GET\_SERVICE\_CLASS\_WORKLOAD  
\_OCCURRENCES 666  
WLM\_GET\_SERVICE\_SUBCLASS\_STATS 671  
WLM\_GET\_SERVICE\_SUPERCLASS\_STATS 676  
WLM\_GET\_WORK\_ACTION\_SET\_STATS 678  
WLM\_GET\_WORKLOAD\_  
OCCURRENCE\_ACTIVITIES 680  
WLM\_GET\_WORKLOAD\_STATS 684

### funciones escalar

EXPLAIN\_FORMAT\_STATS 710

### soportadas 7

### funciones de tabla

admin\_get\_dbp\_mem\_usage 212  
ADMIN\_GET\_MSGS 213  
ADMIN\_GET\_TAB\_INFO\_V95 220  
AUDIT\_ARCHIVE 228  
AUTH\_LIST\_GROUPS\_FOR\_AUTHID 315  
comparadas con vistas administrativas 5  
HEALTH\_GET\_ALERT\_ACTION\_CFG 275

## funciones de tabla (continuación)

HEALTH\_GET\_ALERT\_CFG 278  
HEALTH\_GET\_IND\_DEFINITION 281  
obsoletas 743  
ADMIN\_GET\_TAB\_INFO 747  
SNAP\_GET\_APPL\_INFO 767  
SNAP\_GET\_BP 773  
SNAP\_GET\_DB\_V91 790  
SNAP\_GET\_DBM 787  
SNAP\_GET\_DYN\_SQL\_V91 801  
PD\_GET\_DIAG\_HIST 720  
PD\_GET\_LOG\_MSGS 727  
SNAP\_GET\_AGENT 346, 485  
SNAP\_GET\_AGENT\_MEMORY\_POOL 349, 488  
SNAP\_GET\_APPL\_INFO\_V95 352, 492  
SNAP\_GET\_APPL\_V95 360, 499  
SNAP\_GET\_BP\_PART 374, 513  
SNAP\_GET\_BP\_V95 369, 508  
SNAP\_GET\_CONTAINER\_V91 378, 517  
SNAP\_GET\_DB\_MEMORY\_POOL 395, 534  
SNAP\_GET\_DB\_V95 382, 521  
SNAP\_GET\_DBM\_MEMORY\_POOL 403, 542  
SNAP\_GET\_DBM\_V95 399, 538  
SNAP\_GET\_DETAIL\_LOG\_V91 406, 545  
SNAP\_GET\_DYN\_SQL\_V95 409, 548  
SNAP\_GET\_FCM 415, 554  
SNAP\_GET\_FCM\_PART 417, 556  
SNAP\_GET\_HADR 419, 558  
SNAP\_GET\_LOCK 423, 562  
SNAP\_GET\_LOCKWAIT 429, 568  
SNAP\_GET\_STMT 435, 574  
SNAP\_GET\_STORAGE\_PATHS 442, 581  
SNAP\_GET\_SUBSECTION 444, 583  
SNAP\_GET\_SWITCHES 448, 587  
SNAP\_GET\_TAB\_REORG 454, 593  
SNAP\_GET\_TAB\_V91 451, 590  
SNAP\_GET\_TBSP\_PART\_V91 466, 605  
SNAP\_GET\_TBSP QUIESCER 471, 610  
SNAP\_GET\_TBSP\_RANGE 474, 613  
SNAP\_GET\_TBSP\_V91 459, 598  
SNAP\_GET\_UTIL 478, 617  
SNAP\_GET\_UTIL\_PROGRESS 481, 620  
soportadas 7  
WLM\_GET\_QUEUE\_STATS 657

### funciones escalares

SQLERRM 738

### funciones soportadas 7

## G

GET\_DB\_CONFIG, función de tabla 757  
GET\_DBM\_CONFIG, función de tabla 758  
GET\_DBSIZE\_INFO, procedimiento 717  
GET\_ROUTINE\_OPTS, función escalar 629  
GET\_ROUTINE\_SAR, procedimiento 630  
grupos  
recuperación de pertenencia a grupo 315  
guías de aprendizaje  
resolución de problemas y determinación de  
problemas 870  
Visual Explain 870

## H

HEALTH\_CONT\_HI, función de tabla 252  
HEALTH\_CONT\_HI\_HIS, función de tabla 254

HEALTH\_CONT\_INFO, función de tabla 256  
 HEALTH\_DB\_HI, función de tabla 257  
 HEALTH\_DB\_HI\_HIS, función de tabla 261  
 HEALTH\_DB\_HIC, función de tabla 264  
 HEALTH\_DB\_HIC\_HIS, función de tabla 266  
 HEALTH\_DB\_INFO, función de tabla 269  
 HEALTH\_DBM\_HI, función de tabla 270  
 HEALTH\_DBM\_HI\_HIS, función de tabla 271  
 HEALTH\_DBM\_INFO, función de tabla 274  
 HEALTH\_GET\_ALERT\_ACTION\_CFG, función de tabla 275  
 HEALTH\_GET\_ALERT\_CFG, función de tabla 278  
 HEALTH\_GET\_IND\_DEFINITION, función de tabla 281  
 HEALTH\_HI\_REC, procedimiento 283  
 HEALTH\_TBS\_HI, función de tabla 285  
 HEALTH\_TBS\_HI\_HIS, función de tabla 287  
 HEALTH\_TBS\_INFO, función de tabla 291  
 herramienta de gestión de almacenamiento  
   procedimientos almacenados 643, 644, 645

## I

IMPORT, mandato  
   ADMIN\_CMD, utilización 80  
 indicadores de salud  
   recuperación de definiciones 281  
 información del sistema  
   recuperación 247  
 instalación  
   devolución de información de licencia del producto  
     DB2 246, 248  
   recuperación de información del producto DB2 245  
 instancias  
   recuperación de información de instancia actual 243

## L

listas de contactos  
   recuperación de contactos 700  
   recuperación de listas de grupos de contactos 699  
 listas de notificaciones  
   recuperación de lista de contactos 719  
 LOCKS\_HELD, vista administrativa 335  
 LOCKWAIT, vista administrativa 337  
 LOG\_UTILIZATION, vista administrativa 341  
 LONG\_RUNNING\_SQL, vista administrativa 342

## M

mandato ADD CONTACT  
   mediante ADMIN\_CMD 38  
 mandato ADD CONTACTGROUP  
   mediante ADMIN\_CMD 39  
 mandato AUTOCONFIGURE  
   mediante ADMIN\_CMD 40  
 mandato BACKUP DATABASE  
   mediante ADMIN\_CMD 45  
 mandato DESCRIBE  
   mediante ADMIN\_CMD 53  
 mandato DROP CONTACT  
   mediante ADMIN\_CMD 63  
 mandato DROP CONTACTGROUP  
   mediante ADMIN\_CMD 64  
 mandato EXPORT  
   mediante ADMIN\_CMD 65  
 mandato FORCE APPLICATION  
   mediante ADMIN\_CMD 77

mandato GET STMM TUNING DBPARTITIONNUM  
   mediante ADMIN\_CMD 78  
 mandato INITIALIZE TAPE  
   mediante ADMIN\_CMD 107  
 mandato LOAD  
   mediante ADMIN\_CMD 108  
 mandato PRUNE HISTORY/LOGFILE  
   mediante ADMIN\_CMD 149  
 mandato QUIESCE DATABASE  
   mediante ADMIN\_CMD 151  
 mandato QUIESCE TABLESPACES FOR TABLE  
   mediante ADMIN\_CMD 153  
 mandato RESET ALERT CONFIGURATION  
   mediante ADMIN\_CMD 175  
 mandato RESET DATABASE CONFIGURATION  
   mediante ADMIN\_CMD 176  
 mandato RESET DATABASE MANAGER CONFIGURATION  
   mediante ADMIN\_CMD 178  
 mandato REWIND TAPE  
   mediante ADMIN\_CMD 180  
 mandato RUNSTATS  
   mediante ADMIN\_CMD 180  
 mandato SET TAPE POSITION  
   mediante ADMIN\_CMD 192  
 mandato UNQUIESCE DATABASE  
   mediante ADMIN\_CMD 193  
 mandato UPDATE ALERT CONFIGURATION  
   mediante ADMIN\_CMD 194  
 mandato UPDATE CONTACT  
   mediante ADMIN\_CMD 200  
 mandato UPDATE CONTACTGROUP  
   mediante ADMIN\_CMD 201  
 mandato UPDATE HEALTH NOTIFICATION CONTACT LIST  
   mediante ADMIN\_CMD 207  
 mandato UPDATE STMM TUNING DBPARTITIONNUM  
   mediante ADMIN\_CMD 211  
 mandatos  
   ADD CONTACT 38  
   ADD CONTACTGROUP 39  
   AUTOCONFIGURE 40  
   BACKUP DATABASE 45  
   DESCRIBE 53  
   DROP CONTACT 63  
   DROP CONTACTGROUP 64  
   EXPORT 65  
   FORCE APPLICATION 77  
   GET STMM TUNING DBPARTITIONNUM 78  
   IMPORT 80  
   INITIALIZE TAPE 107  
   invocar desde procedimiento 36  
   LOAD 108  
   PRUNE HISTORY/LOGFILE 149  
   QUIESCE DATABASE 151  
   QUIESCE TABLESPACES FOR TABLE 153  
   REDISTRIBUTE DATABASE PARTITION GROUP 156  
   REORG INDEXES/TABLE 164  
   RESET ALERT CONFIGURATION 175  
   RESET DATABASE CONFIGURATION 176  
   RESET DATABASE MANAGER CONFIGURATION 178  
   REWIND TAPE 180  
   RUNSTATS 180  
   SET TAPE POSITION 192  
   UNQUIESCE DATABASE 193  
   UPDATE ALERT CONFIGURATION 194  
   UPDATE CONTACT 200  
   UPDATE CONTACTGROUP 201  
   UPDATE DATABASE CONFIGURATION 202

mandatos (*continuación*)  
 UPDATE DATABASE MANAGER CONFIGURATION 205  
 UPDATE HEALTH NOTIFICATION CONTACT LIST 207  
 UPDATE HISTORY 208  
 UPDATE STMM TUNING DBPARTITIONNUM 211

manuales en copia impresa  
 pedido 866

mensajes de anotación cronológica de notificación  
 recuperación 727

mensajes de error  
 recuperación de información 738

MQPUBLISH, función escalar 293  
 MQREAD, función escalar 294  
 MQREADALL, función de tabla 295  
 MQREADALLCLOB, función de tabla 297  
 MQREADCLOB, función escalar 299  
 MQRECEIVE, función escalar 300  
 MQRECEIVEALL, función de tabla 302  
 MQRECEIVEALLCLOB, función de tabla 304  
 MQRECEIVECLOB, función escalar 306  
 MQSEND, función escalar 307  
 MQSUBSCRIBE, función escalar 309  
 MQUNSUBSCRIBE, función escalar 310

## N

NOTIFICATIONLIST, vista administrativa 719

## O

OBJECTOWNERS, vista administrativa 319

objetos  
 recuperación de propiedad de objeto 319

obsoleta  
 funciones de tabla 822, 838, 844, 845, 850, 853  
 SNAP\_GET\_APPL, función de tabla 759

obsoletas  
 funcionalidad  
 rutinas administrativas de SQL 743  
 funciones de tabla 758, 778, 779, 805, 807, 808, 809, 813,  
 815, 816, 824, 827, 828, 835, 837, 840, 841, 842, 846, 848,  
 852, 854, 857, 859  
 SNAP\_GET\_BP, función de tabla 773  
 SNAP\_GET\_DB\_V91, función de tabla 790  
 SNAP\_GET\_DBM, función de tabla 787

obsoletos  
 procedimientos 757, 840, 860

## P

PD\_GET\_LOG\_MSGS, función de tabla 727  
 PDLOGMSGs\_LAST24HOURS, vista administrativa 727  
 pedido de manuales de DB2 866  
 PRIVILEGES, vista administrativa 320  
 privilegios  
 recuperación 320

Procedimiento ANALYZE\_LOG\_SPACE 633  
 Procedimiento AUTOMAINT\_GET\_POLICY 232  
 Procedimiento AUTOMAINT\_GET\_POLICYFILE 233  
 Procedimiento AUTOMAINT\_SET\_POLICY 234  
 Procedimiento AUTOMAINT\_SET\_POLICYFILE 235  
 Procedimiento CAPTURE\_STORAGEEMGMT\_INFO 643  
 Procedimiento CREATE\_STORAGEEMGMT\_TABLES 644  
 Procedimiento DROP\_STORAGEEMGMT\_TABLES 645  
 Procedimiento GENERATE\_DISTFILE 636  
 Procedimiento GET\_SWRD\_SETTINGS 637

Procedimiento SET\_SWRD\_SETTINGS 639  
 procedimiento WLM\_CANCEL\_ACTIVITY 646  
 procedimiento  
 WLM\_CAPTURE\_ACTIVITY\_IN\_PROGRESS 647

procedimientos 1  
 ADMIN\_CMD 36  
 ADMIN\_COPY\_SCHEMA 687  
 ADMIN\_DROP\_SCHEMA 692  
 ADMIN\_REMOVE\_MSGS 219  
 ALTOBJ 694  
 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 633  
 AUTOMAINT\_GET\_POLICY 232  
 AUTOMAINT\_GET\_POLICYFILE 233  
 AUTOMAINT\_SET\_POLICYFILE 235  
 CAPTURE\_STORAGEEMGMT\_INFO 643  
 CREATE\_STORAGEEMGMT\_TABLES 644  
 DROP\_STORAGEEMGMT\_TABLES 645  
 GENERATE\_DISTFILE 636  
 GET\_DBSIZE\_INFO 717  
 GET\_ROUTINE\_SAR 630  
 GET\_SWRD\_SETTINGS 637  
 HEALTH\_HI\_REC 283  
 obsoletos 743  
 procedimientos almacenados  
 AUDIT\_ARCHIVE 228  
 AUDIT\_DELIM\_EXTRACT 229  
 WLM\_CANCEL\_ACTIVITY 646  
 WLM\_CAPTURE\_ACTIVITY\_IN\_PROGRESS 647  
 PUT\_ROUTINE\_SAR 631  
 REBIND\_ROUTINE\_PACKAGE 632  
 REORGCHK\_IX\_STATS 734  
 REORGCHK\_TB\_STATS 736  
 SET\_ROUTINE\_OPTS 633  
 SET\_SWRD\_SETTINGS 639  
 SNAP\_WRITE\_FILE 484, 623  
 SNAPSHOT\_FILEW (obsoleto) 840  
 soportados 7  
 STEPWISE\_REDISTRIBUTE\_DBPG 642  
 SYSINSTALLOBJECTS 740  
 SYSINSTALLROUTINES 860  
 WLM\_COLLECT\_STATS 649

procedimientos almacenados  
 AUDIT\_ARCHIVE 228  
 AUDIT\_DELIM\_EXTRACT 229  
 PUT\_ROUTINE\_SAR, procedimiento 631

## Q

QUERY\_PREP\_COST, vista administrativa 345

## R

REBIND\_ROUTINE\_PACKAGE, procedimiento 632  
 redistribución de datos  
 procedimientos 633, 636, 637, 639, 642  
 REDISTRIBUTE DATABASE PARTITION GROUP, mandato  
 ADMIN\_CMD, utilización 156  
 REG\_VARIABLES, vista administrativa 241  
 REORG INDEXES/TABLE, mandato  
 ADMIN\_CMD, utilización 164

REORGCHK\_IX\_STATS, procedimiento 734  
 REORGCHK\_TB\_STATS, procedimiento 736  
 resolución de problemas  
   guías de aprendizaje 870  
   información en línea 870  
 Rutina administrativa  
   Procedimiento AUTOMAINT\_SET\_POLICY 234  
 rutinas  
   administrativas de SQL  
     soportadas 7  
   rutinas administrativas de SQL 743  
 rutinas administrativas  
   AUTH\_LIST\_ROLES\_FOR\_AUTHID 316  
 rutinas administrativas de SQL  
   soportadas 7  
 Rutinas administrativas de SQL  
   rutinas obsoletas 743

## S

sentencias SQL  
   visualización de la ayuda 867  
 SET\_ROUTINE\_OPTS, procedimiento 633  
 SNAP\_GET\_AGENT, función de tabla 346, 485  
 SNAP\_GET\_AGENT\_MEMORY\_POOL, función de  
   tabla 349, 488  
 SNAP\_GET\_APPL\_INFO, función de tabla 767  
 SNAP\_GET\_BP\_PART, función de tabla 374, 513  
 SNAP\_GET\_CONTAINER, función de tabla obsoleta 778  
 SNAP\_GET\_CONTAINER\_V91, función de tabla 378, 517  
 SNAP\_GET\_DB, función de tabla obsoleta 779  
 SNAP\_GET\_DB\_MEMORY\_POOL, función de tabla 395, 534  
 SNAP\_GET\_DBM\_MEMORY\_POOL, función de tabla 403,  
   542  
 SNAP\_GET\_DETAIL\_LOG\_V91, función de tabla 406, 545  
 SNAP\_GET\_DYN\_SQL, función de tabla obsoleta 805  
 SNAP\_GET\_DYN\_SQL\_V91, función de tabla 801  
 SNAP\_GET\_FCM, función de tabla 415, 554  
 SNAP\_GET\_FCM\_PART, función de tabla 417, 556  
 SNAP\_GET\_HADR, función de tabla 419, 558  
 SNAP\_GET\_LOCK, función de tabla 423, 562  
 SNAP\_GET\_LOCKWAIT, función de tabla 429, 568  
 SNAP\_GET\_STMT, función de tabla 435, 574  
 SNAP\_GET\_STO\_PATHS, función de tabla obsoleta 807  
 SNAP\_GET\_SUBSECTION, función de tabla 444, 583  
 SNAP\_GET\_SWITCHES, función de tabla 448, 587  
 SNAP\_GET\_TAB, función de tabla obsoleta 808  
 SNAP\_GET\_TAB\_REORG, función de tabla 454, 593  
 SNAP\_GET\_TAB\_V91, función de tabla 451, 590  
 SNAP\_GET\_TBSP, función de tabla obsoleta 809  
 SNAP\_GET\_TBSP\_PART, función de tabla obsoleta 813  
 SNAP\_GET\_TBSP\_PART\_V91, función de tabla 466, 605  
 SNAP\_GET\_TBSP QUIESCER, función de tabla 471, 610  
 SNAP\_GET\_TBSP\_RANGE, función de tabla 474, 613  
 SNAP\_GET\_TBSP\_V91, función de tabla 459, 598  
 SNAP\_GET\_UTIL, función de tabla 478, 617  
 SNAP\_GET\_UTIL\_PROGRESS, función de tabla 481, 620  
 SNAP\_WRITE\_FILE, procedimiento 484, 623  
 SNAPAGENT, vista administrativa 346, 485  
 SNAPAGENT\_MEMORY\_POOL, vista administrativa 349,  
   488  
 SNAPAPPL, vista administrativa 360, 499  
 SNAPAPPL\_INFO, vista administrativa 352, 492  
 SNAPBP, vista administrativa 369, 508  
 SNAPBP\_PART, vista administrativa 374, 513  
 SNAPCONTAINER, vista administrativa 378, 517  
 SNAPDB, vista administrativa 382, 521

SNAPDB\_MEMORY\_POOL, vista administrativa 395, 534  
 SNAPDBM, vista administrativa 399, 538  
 SNAPDBM\_MEMORY\_POOL, vista administrativa 403, 542  
 SNAPDETAILOG, vista administrativa 406, 545  
 SNAPFCM, vista administrativa 415, 554  
 SNAPFCM\_PART, vista administrativa 417, 556  
 SNAPHADR, vista administrativa 419, 558  
 SNAPLOCK, vista administrativa 423, 562  
 SNAPLOCKWAIT, vista administrativa 429, 568  
 SNAPSHOT\_AGENT, función de tabla obsoleta 815  
 SNAPSHOT\_APPL, función de tabla obsoleta 816  
 SNAPSHOT\_APPL\_INFO, función de tabla obsoleta 822  
 SNAPSHOT\_BP, función de tabla obsoleta 824  
 SNAPSHOT\_CONTAINER, función de tabla obsoleta 827  
 SNAPSHOT\_DATABASE, función de tabla obsoleta 828  
 SNAPSHOT\_DBM, función de tabla obsoleta 835  
 SNAPSHOT\_DYN\_SQL, función de tabla obsoleta 837  
 SNAPSHOT\_FCM, función de tabla obsoleta 838  
 SNAPSHOT\_FCMNODE, función de tabla obsoleta 840  
 SNAPSHOT\_FILEW, procedimiento obsoleto 840  
 SNAPSHOT\_LOCK, función de tabla obsoleta 841  
 SNAPSHOT\_LOCKWAIT, función de tabla obsoleta 842  
 SNAPSHOT QUIESCERS, función de tabla obsoleta 844  
 SNAPSHOT\_RANGES, función de tabla obsoleta 845  
 SNAPSHOT\_STATEMENT, función de tabla obsoleta 846  
 SNAPSHOT\_SUBSECT, función de tabla obsoleta 848  
 SNAPSHOT\_SWITCHES, función de tabla obsoleta 850  
 SNAPSHOT\_TABLE, función de tabla obsoleta 852  
 SNAPSHOT\_TBREORG, función de tabla obsoleta 853  
 SNAPSHOT\_TBS, función de tabla obsoleta 854  
 SNAPSHOT\_TBS\_CFG, función de tabla obsoleta 857  
 SNAPSTMT, vista administrativa 435, 574  
 SNAPSUBSECTION, vista administrativa 444, 583  
 SNAPSWITCHES, vista administrativa 448, 587  
 SNAPTAB, vista administrativa 451, 590  
 SNAPTAB\_REORG, vista administrativa 454, 593  
 SNAPTbsp, vista administrativa 459, 598  
 SNAPTbsp QUIESCER, vista administrativa 471, 610  
 SNAPTbsp\_RANGE, vista administrativa 474, 613  
 SNAPTbspPART, vista administrativa 466, 605  
 SNAPUTIL, vista administrativa 478, 617  
 SNAPUTIL\_PROGRESS, vista administrativa 481, 620  
 SQLCACHE\_SNAPSHOT, función de tabla obsoleta 859  
 SQLERRM, función escalar 738  
 STEPWISE\_REDISTRIBUTE\_DBPG, procedimiento 642  
 SYSINSTALLOBJECTS, procedimiento 740  
 SYSINSTALLROUTINES, procedimiento obsoleto 860

## T

tablas  
   recuperación de tamaño y estado 220, 747  
 TBSP\_UTILIZATION, vista administrativa 624  
 términos y condiciones  
   uso de publicaciones 871  
 TOP\_DYNAMIC\_SQL, vista administrativa 628

## U

UPDATE DATABASE CONFIGURATION, mandato  
   ADMIN\_CMD, utilización 202  
 UPDATE DATABASE MANAGER CONFIGURATION,  
   mandato  
   ADMIN\_CMD, utilización 205  
 UPDATE HISTORY, mandato  
   ADMIN\_CMD, utilización 208

## V

### variables de registro

- recuperación de valores en uso 241
- vías de acceso de base de datos
  - recuperación 706
- visión general de la documentación 863
- vista administrativa ENV\_FEATURE\_INFO 246
- Vista administrativa ENV\_SYS\_RESOURCES 248
- vista administrativa SNAPDYN\_SQL 409, 548
- vista administrativa SNAPSTORAGE\_PATHS 442, 581

### vistas

#### vistas administrativas

- ADMINTABCOMPRESSINFO 215
- ADMINTABINFO 220
- APPL\_PERFORMANCE 322
- APPLICATIONS 323
- AUTHORIZATIONIDS 318
- BP\_HITRATIO 327
- BP\_READ\_IO 329
- BP\_WRITE\_IO 331
- CONTACTGROUPS 699
- CONTACTS 700
- CONTAINER\_UTILIZATION 333
- DB\_HISTORY 701
- DBCFCG 237
- DBMCFG 239
- DBPATHS 706
- ENV\_FEATURE\_INFO 246
- ENV\_INST\_INFO 243
- ENV\_PROD\_INFO 245
- ENV\_SYS\_INFO 247
- ENV\_SYS\_RESOURCES 248
- LOCKS\_HELD 335
- LOCKWAIT 337
- LOG\_UTILIZATION 341
- LONG\_RUNNING\_SQL 342
- NOTIFICATIONLIST 719
- OBJECTOWNERS 319
- PDLOGMSG\_LAST24HOURS 727
- PRIVILEGES 320
- QUERY\_PREP\_COST 345
- REG\_VARIABLES 241
- SNAPAGENT 346, 485
- SNAPAGENT\_MEMORY\_POOL 349, 488
- SNAPAPPL 360, 499
- SNAPAPPL\_INFO 352, 492
- SNAPBP 369, 508
- SNAPBP\_PART 374, 513
- SNAPCONTAINER 378, 517
- SNAPDB 382, 521
- SNAPDB\_MEMORY\_POOL 395, 534
- SNAPDBM 399, 538
- SNAPDBM\_MEMORY\_POOL 403, 542
- SNAPDETAILLOG 406, 545
- SNAPDYN\_SQL 409, 548
- SNAPFCM 415, 554
- SNAPFCM\_PART 417, 556
- SNAPHADR 419, 558
- SNAPLOCK 423, 562
- SNAPLOCKWAIT 429, 568
- SNAPSTMT 435, 574
- SNAPSTORAGE\_PATHS 442, 581
- SNAPSUBSECTION 444, 583
- SNAPSWITCHES 448, 587
- SNAPTAB 451, 590
- SNAPTAB\_REORG 454, 593
- SNAPTbsp 459, 598

### vistas (continuación)

#### vistas administrativas (continuación)

- SNAPTbsp QUIESCER 471, 610
- SNAPTbsp RANGE 474, 613
- SNAPTbsp PART 466, 605
- SNAPUTIL 478, 617
- SNAPUTIL\_PROGRESS 481, 620
- Tbsp\_UTILIZATION 624
- TOP\_DYNAMIC\_SQL 628

#### vistas administrativas

- ADMINTABINFO 220
- APPL\_PERFORMANCE 322
- APPLICATIONS 323
- AUTHORIZATIONIDS 318
- autorización 3
- BP\_HITRATIO 327
- BP\_READ\_IO 329
- BP\_WRITE\_IO 331
- comparadas con funciones de tablas 5
- CONTACTGROUPS 699
- CONTACTS 700
- CONTAINER\_UTILIZATION 333
- DB\_HISTORY 701
- DBCFCG 237
- DBMCFG 239
- DBPATHS 706
- ENV\_FEATURE\_INFO 246
- ENV\_INST\_INFO 243
- ENV\_PROD\_INFO 245
- ENV\_SYS\_INFO 247
- ENV\_SYS\_RESOURCES 248
- LOCKS\_HELD 335
- LOCKWAIT 337
- LOG\_UTILIZATION 341
- LONG\_RUNNING\_SQL 342
- NOTIFICATIONLIST 719
- OBJECTOWNERS 319
- PDLOGMSG\_LAST24HOURS 727
- PRIVILEGES 320
- QUERY\_PREP\_COST 345
- REG\_VARIABLES 241
- SNAPAGENT 346, 485
- SNAPAGENT\_MEMORY\_POOL 349, 488
- SNAPAPPL 360, 499
- SNAPAPPL\_INFO 352, 492
- SNAPBP 369, 508
- SNAPBP\_PART 374, 513
- SNAPCONTAINER 378, 517
- SNAPDB 382, 521
- SNAPDB\_MEMORY\_POOL 395, 534
- SNAPDBM 399, 538
- SNAPDBM\_MEMORY\_POOL 403, 542
- SNAPDETAILLOG 406, 545
- SNAPDYN\_SQL 409, 548
- SNAPFCM 415, 554
- SNAPFCM\_PART 417, 556
- SNAPHADR 419, 558
- SNAPLOCK 423, 562
- SNAPLOCKWAIT 429, 568
- SNAPSTMT 435, 574
- SNAPSTORAGE\_PATHS 442, 581
- SNAPSUBSECTION 444, 583
- SNAPSWITCHES 448, 587
- SNAPTAB 451, 590
- SNAPTAB\_REORG 454, 593
- SNAPTbsp 459, 598
- SNAPTbsp QUIESCER 471, 610

vistas administrativas (*continuación*)  
SNAPTbsp\_RANGE 474, 613  
SNAPTbspPART 466, 605  
SNAPUTIL 478, 617  
SNAPUTIL\_PROGRESS 481, 620  
soportadas 7  
Tbsp\_UTILIZATION 624  
TOP\_DYNAMIC\_SQL 628  
visión general 1  
Visual Explain  
guía de aprendizaje 870

## W

WLM\_COLLECT\_STATS 649





SC11-3507-00





Spine information:

DB2 Versión 9.5 for Linux, UNIX, and Windows **Versión 9 Release 5**

**Rutinas y vistas administrativas**

