

Data Warehouse Service

Preguntas frecuentes

Edición 01

Fecha 2023-10-12



Copyright © Huawei Cloud Computing Technologies Co., Ltd. 2023. Todos los derechos reservados.

Quedan terminantemente prohibidas la reproducción y/o la divulgación totales y/o parciales del presente documento de cualquier forma y/o por cualquier medio sin la previa autorización por escrito de Huawei Cloud Computing Technologies Co., Ltd.

Marcas registradas y permisos



El logotipo  y otras marcas registradas de Huawei pertenecen a Huawei Technologies Co., Ltd. Todas las demás marcas registradas y los otros nombres comerciales mencionados en este documento son propiedad de sus respectivos titulares.

Aviso

Es posible que la totalidad o parte de los productos, las funcionalidades y/o los servicios que figuran en el presente documento no se encuentren dentro del alcance de un contrato vigente entre Huawei Cloud y el cliente. Las funcionalidades, los productos y los servicios adquiridos se limitan a los estipulados en el respectivo contrato. A menos que un contrato especifique lo contrario, ninguna de las afirmaciones, informaciones ni recomendaciones contenidas en el presente documento constituye garantía alguna, ni expresa ni implícita.

Huawei está permanentemente preocupada por la calidad de los contenidos de este documento; sin embargo, ninguna declaración, información ni recomendación aquí contenida constituye garantía alguna, ni expresa ni implícita. La información contenida en este documento se encuentra sujeta a cambios sin previo aviso.

Índice

1 Preguntas frecuentes principales.....	1
2 Problemas generales.....	7
2.1 ¿Por qué son necesarios los almacenes de datos?.....	7
2.2 ¿Cuáles son las diferencias entre un almacén de datos y la plataforma de big data de Hadoop?.....	8
2.3 ¿Por qué debería usar la nube pública GaussDB(DWS)?.....	9
2.4 ¿Debo elegir GaussDB(DWS) de la nube pública o RDS?.....	9
2.5 ¿Cuándo debo usar GaussDB(DWS) y MRS?.....	10
2.6 ¿Qué es la cuota de usuario?.....	10
2.7 ¿Cuáles son las diferencias entre usuarios y roles?.....	11
2.8 ¿Cómo verifico el tiempo de creación de un usuario de base de datos?.....	12
2.9 Regiones y AZ.....	13
2.10 ¿Mis datos están seguros en GaussDB(DWS)?.....	15
2.11 ¿Cómo se asegura GaussDB(DWS)?.....	16
2.12 ¿Puedo modificar el grupo de seguridad de un clúster de GaussDB(DWS)?.....	16
2.13 ¿Cómo se relacionan LibrA, GaussDB A y GaussDB(DWS)?.....	16
2.14 ¿Qué es una base de datos/almacén de datos/lago de datos/lakehouse?.....	17
3 Gestión de clúster.....	21
3.1 ¿Qué hago si la creación de un clúster de GaussDB(DWS) ha fallado?.....	21
3.2 ¿Cómo puedo borrar y recuperar el espacio de almacenamiento?.....	21
3.3 ¿Puedo cambiar mis nodos de clúster a otra región después de la compra?.....	22
3.4 ¿Por qué se redujo el almacenamiento usado después de la expansión horizontal?.....	22
3.5 ¿Cómo puedo ver las métricas de nodo (CPU, memoria y uso de disco)?.....	23
3.6 ¿GaussDB(DWS) soporta el nodo único para un entorno de aprendizaje?.....	23
3.7 ¿GaussDB(DWS) es compatible con BMS?.....	23
3.8 ¿Cómo se calcula el espacio en disco o la capacidad de GaussDB(DWS)?.....	23
3.9 ¿Cuáles son las bases de datos gaussdb y postgres de GaussDB(DWS)?.....	24
3.10 ¿Cómo configuro el número máximo de sesiones al agregar una regla de alarma en Cloud Eye?.....	24
3.11 ¿Cómo puedo determinar si un clúster utiliza la arquitectura x86 o Arm?.....	25
3.12 ¿Qué debo hacer si falla la comprobación de expansión horizontal?.....	27
3.13 ¿Cuándo debo agregar CN o ampliar un clúster?.....	28
3.14 ¿Cuáles son los escenarios de cambiar el tamaño de un clúster, cambiar la variante del nodo, ampliar y reducir?.....	29

3.15 ¿Cómo debo seleccionar entre un clúster de muchos nodos con variante pequeña y un clúster de tres nodos de variante grande con los mismos núcleos de CPU y memoria?.....	30
3.16 ¿Cuáles son las diferencias entre las SSD en la nube y las SSD locales?.....	31
3.17 ¿Cuáles son las diferencias entre el almacenamiento de datos en caliente y el almacenamiento de datos en frío?.....	31
4 Conexión de bases de datos.....	33
4.1 ¿Cómo se comunican las aplicaciones con GaussDB(DWS)?.....	33
4.2 ¿GaussDB(DWS) es compatible con clientes de terceros y controladores JDBC y ODBC?.....	38
4.3 ¿Puedo conectarme a nodos de clúster de GaussDB(DWS) usando SSH?.....	38
4.4 ¿Qué debo hacer si no puedo conectarme a un clúster de almacenamiento de datos?.....	38
4.5 ¿Por qué no se me notificó un fallo al desvincular la EIP cuando GaussDB(DWS) está conectado por Internet?.....	39
4.6 ¿Cómo configuro una lista blanca para proteger los clústeres disponibles por una dirección IP pública?.....	40
5 Importación y exportación de datos.....	41
5.1 ¿Cuáles son las diferencias entre los formatos de datos admitidos por las tablas extranjeras OBS y GDS?.....	41
5.2 ¿Cómo puedo importar datos incrementales usando una tabla externa de OBS?.....	41
5.3 ¿Cómo puedo importar datos a GaussDB(DWS)?.....	41
5.4 ¿Cuántos datos de servicio puede almacenar un almacén de datos?.....	42
5.5 ¿Cómo uso \Copy para importar y exportar datos?.....	42
5.6 ¿Cómo puedo implementar la importación de tolerancia a fallos entre diferentes bibliotecas de codificación.....	43
5.7 ¿Puedo exportar/importar datos de GaussDB(DWS) desde/a OBS entre regiones?.....	44
5.8 ¿Cómo importo datos de GaussDB(DWS)/Oracle/MySQL/SQL Server a GaussDB(DWS) (Migración de toda la base de datos)?.....	45
5.9 ¿Puedo importar datos por la red pública/externa con GDS?.....	45
5.10 ¿Cuáles son los factores que afectan al rendimiento de importación de GaussDB(DWS)?.....	45
6 Cuenta, contraseña y permisos.....	46
6.1 ¿Cómo implementa GaussDB(DWS) el aislamiento de la carga de trabajo?.....	46
6.2 ¿Cómo cambio la contraseña de una cuenta de base de datos cuando caduca la contraseña?.....	50
6.3 ¿Cómo puedo conceder permisos de tabla a un usuario?.....	51
6.4 ¿Cómo puedo conceder permisos de esquema a un usuario?.....	55
6.5 ¿Cómo creo un usuario de solo lectura de la base de datos?.....	57
6.6 ¿Cómo puedo crear usuarios y tablas privadas de bases de datos?.....	58
6.7 ¿Cómo revoco el permiso de CONNECT ON DATABASE de un usuario?.....	59
6.8 ¿Cómo puedo ver los permisos de tabla de un usuario?.....	60
6.9 ¿Quién es la usuaria Ruby?.....	62
7 Uso de la base de datos.....	63
7.1 ¿Cómo cambio las columnas de distribución?.....	63
7.2 ¿Cómo puedo ver y establecer la codificación de caracteres de la base de datos?.....	64
7.3 ¿Qué hago si el tipo de fecha se convierte automáticamente en el tipo de marca de tiempo durante la creación de la tabla?.....	66
7.4 ¿Necesito ejecutar VACUUM FULL y ANALYZE en tablas comunes periódicamente?.....	67
7.5 ¿Necesito ajustar una clave de distribución después de establecer una clave principal?.....	69
7.6 ¿Es GaussDB(DWS) compatible con los procedimientos almacenados de PostgreSQL?.....	69
7.7 ¿Qué son las tablas particionadas, las particiones y las claves de partición?.....	69

7.8 ¿Cómo puedo exportar la estructura de la tabla?.....	69
7.9 ¿Cómo puedo eliminar los datos de la tabla de manera eficiente?.....	70
7.10 ¿Cómo puedo ver la información de la tabla extranjera?.....	71
7.11 Si no se especifica ninguna columna de distribución, ¿cómo se almacenarán los datos?.....	71
7.12 ¿Cómo reemplazo el resultado nulo por 0?.....	73
7.13 ¿Cómo puedo comprobar si una tabla está almacenada en filas o en columnas?.....	74
7.14 ¿Cómo puedo consultar la información sobre las tablas de almacén de columnas de GaussDB(DWS)?.....	75
7.15 ¿Por qué a veces los índices de consulta de GaussDB(DWS) se vuelven inválidos?.....	76
7.16 ¿Cómo uso una función definida por el usuario para reescribir la función CRC32()?.....	85
7.17 ¿Cuáles son los esquemas que comienzan con pg_toast_temp* o pg_temp* ?.....	86
7.18 Soluciones a los resultados incoherentes de las consultas a GaussDB(DWS).....	86
7.19 ¿En qué catálogos de sistema no se puede realizar la operación de VACUUM FULL?.....	92
7.20 ¿En qué escenarios estaría una declaración "idle in transaction"?.....	93
7.21 ¿Cómo implementa GaussDB(DWS) las conversiones de fila a columna y de columna a fila?.....	95
7.22 ¿Cuáles son las diferencias entre las restricciones únicas y los índices únicos?.....	98
8 Rendimiento de bases de datos.....	100
8.1 ¿Por qué la ejecución de SQL es lenta después del uso largo de GaussDB(DWS)?.....	100
8.2 ¿Por qué GaussDB(DWS) funciona peor que una base de datos de un solo servidor en los escenarios extremos?..	101
8.3 ¿Cómo puedo ver registros de ejecución de SQL en un cierto período cuando las solicitudes de lectura y escritura están bloqueadas?.....	101
8.4 ¿Qué hago si mi clúster no está disponible debido a la insuficiencia de espacio?.....	102
8.5 ¿Qué es el derrame del operador en GaussDB(DWS)?.....	102
8.6 Gestión de recursos de CPU de GaussDB(DWS).....	103
8.7 ¿Por qué las tareas ejecutadas por un usuario ordinario son más lentas que las ejecutadas por el usuario dbadmin?.....	106
8.8 ¿Cuáles son los factores relacionados con el rendimiento de la consulta de tabla única en GaussDB(DWS)?.....	108
9 Copia de respaldo y restauración de instantáneas.....	110
9.1 ¿Por qué la creación de una instantánea automatizada es tan lenta?.....	110
9.2 ¿Una instantánea de DWS tiene la misma función que una instantánea de EVS?.....	110
10 Facturación.....	112
10.1 ¿Cómo puedo renovar el servicio?.....	112
10.2 ¿Se admite el reembolso?.....	113
10.3 ¿Por qué se me dedujeron las tarifas después de que venciera mi prueba gratuita de GaussDB(DWS)?.....	113
10.4 ¿Por qué no puedo ver un clúster después de suscribirme a una prueba gratuita de GaussDB(DWS)?.....	113
10.5 ¿Por qué no se muestra ningún clúster después de comprar un paquete de descuento?.....	114
10.6 ¿Por qué todavía se deducen las tarifas después de darme de baja de un paquete de descuento?.....	114
10.7 ¿Cómo puedo detener la facturación de GaussDB(DWS)?.....	114
10.8 ¿Se detiene la facturación de pago por uso cuando se detiene mi clúster?.....	115
10.9 ¿Cómo puedo reutilizar un paquete de descuento después de que mi paquete caduque?.....	115
10.10 ¿Por qué el botón de compra no está disponible cuando creo un clúster?.....	116
10.11 Si no uso el paquete de un año que compré y elimino todos los grupos, ¿se deducirá la cuota del paquete?.....	116

10.12 ¿Se solaparán los períodos de validez de los paquetes de GaussDB(DWS) cuando compre otro paquete antes de que expire el paquete actual?.....	116
10.13 ¿Puedo crear varios clústeres cuando compre un paquete que tenga más de seis nodos?.....	117
10.14 ¿Cómo descongelar un clúster?.....	117
10.15 ¿Puedo congelar o cerrar un clúster de GaussDB(DWS) para detener la facturación?.....	117
10.16 ¿Se pueden asociar automáticamente los nodos de descuento adquiridos después de la creación del clúster con los nodos del clúster antiguo?.....	117
10.17 ¿Cómo sé cuándo caduca mi paquete de descuento GaussDB(DWS)?.....	118

1 Preguntas frecuentes principales

Recientemente, hemos recopilado las preguntas frecuentes sobre GaussDB(DWS) con el [Chatbot inteligente](#). Puede encontrar respuestas a sus preguntas en esta página.

Uso de sintaxis

Tabla 1-1 Uso de sintaxis

Clasificación	Preguntas frecuentes	URL de la página
1	<ul style="list-style-type: none">● ¿Cómo puedo consultar el número de bits en una cadena de consulta?● ¿Cómo se trunca una subcadena?● ¿Cómo reemplazo algunas cadenas en el resultado devuelto?● ¿Cómo devuelvo los primeros caracteres de una cadena?● ¿Cómo puedo filtrar el encabezado y la cola de una cadena?● ¿Cómo obtengo el número de bytes de una cadena especificada?	Funciones y operadores de procesamiento de caracteres
2	<ul style="list-style-type: none">● ¿Cómo creo una tabla de particiones?● ¿Cuáles son los tipos de particiones de tabla compatibles?	CREATE TABLE PARTITION
3	<ul style="list-style-type: none">● ¿Cómo puedo ver todos los esquemas?● ¿Cómo puedo ver todas las tablas de un esquema?	Esquema

Clasificación	Preguntas frecuentes	URL de la página
4	<ul style="list-style-type: none"> ● ¿Cómo agrego una columna de tabla? ● ¿Cómo cambio un tipo de datos? ● ¿Cómo agrego una restricción NOT NULL a una columna? ● ¿Cómo configuro la clave principal? ● ¿Cómo se modifican los atributos de la tabla? 	ALTER TABLE
5	<ul style="list-style-type: none"> ● ¿Cómo uso las funciones de fecha? ● ¿Cómo uso pg_sleep()? ● ¿Cómo realizo la sustracción mensual? ● ¿Cómo uso las funciones para cambiar los tipos de fecha? 	Funciones y operadores de procesamiento de hora y fecha
6	<ul style="list-style-type: none"> ● ¿Cómo cambio las columnas de distribución? ● ¿Cómo cambio la columna de distribución a otra columna? ● ¿Por qué no se pueden actualizar los datos de la columna de distribución y se muestra el mensaje "Distributed key column can't be updated in current version"? 	¿Cómo cambio las columnas de distribución?
7	<ul style="list-style-type: none"> ● ¿Cómo gestiono las particiones? ● ¿Cómo agrego o elimino una partición? ● ¿Cómo cambio el nombre de una partición? ● ¿Cómo puedo consultar datos en una partición? 	Creación y gestión de tablas particionadas
8	<ul style="list-style-type: none"> ● ¿Cómo puedo consultar el número de filas en una partición? ● ¿Cómo fusiono dos particiones? 	ALTER TABLE PARTITION
9	¿Cómo llamo al procedimiento almacenado?	CALL
10	<ul style="list-style-type: none"> ● ¿Cómo creo una tabla? ● create table like 	CREATE TABLE

Clasificación	Preguntas frecuentes	URL de la página
11	<ul style="list-style-type: none"> ● ¿Cuál es el comando para la autorización? ● ¿Cómo uso la sintaxis de concesión? ● ¿Cómo otorgo los derechos de un usuario a otros usuarios? ● ¿Cómo puedo conceder permisos de tabla a un usuario? ● ¿Cómo puedo conceder permisos de base de datos a un usuario? ● ¿Cuáles son los permisos de la tabla externa? 	GRANT
12	<ul style="list-style-type: none"> ● ¿Cómo uso la función REPLACE? ● ¿Cómo uso to_timestamp? ● ¿Cómo puedo ocultar una marca de tiempo en un formato especificado? ● ¿Cómo utilizo current_timestamp? ● ¿Cómo uso to_number? 	Funciones de conversión de tipo

Gestión de bases de datos

Tabla 1-2 Gestión de bases de datos

Clasificación	Preguntas frecuentes	URL de la página
1	<ul style="list-style-type: none"> ● ¿Cómo puedo ver las definiciones de tabla? ● ¿Cómo puedo ver el DDL? ● ¿Cómo puedo ver la estructura de las vistas? ● ¿Cómo puedo consultar el tamaño de una base de datos o una tabla? 	Consultar el tamaño de la tabla y la base de datos

Clasificación	Preguntas frecuentes	URL de la página
2	<ul style="list-style-type: none"> ● ¿Cómo se borran los espacios de tablas? ● ¿Cómo uso la sintaxis VACUUM FULL? ● ¿Cómo puedo mejorar el rendimiento de la consulta después de un gran número de operaciones de adición, eliminación y modificación? 	VACUUM
3	<ul style="list-style-type: none"> ● ¿Cómo puedo comprobar si un usuario tiene permisos en la tabla actual? ● ¿Cómo puedo comprobar si un usuario tiene un permiso determinado en una tabla? 	¿Cómo puedo ver los permisos de tabla de un usuario?
4	<ul style="list-style-type: none"> ● ¿Cómo puedo consultar y terminar instrucciones bloqueadas? ● ¿Cómo puedo consultar las sentencias activas? ● ¿Cómo cancelo una sesión? ● ¿Qué debo hacer cuando bloquee los tiempos de espera fuera? 	Análisis de sentencias de SQL que se están ejecutando
5	<ul style="list-style-type: none"> ● ¿Hay una descripción detallada de los planes de ejecución de SQL? ● ¿Cómo puedo ver el plan de ejecución? ● ¿Cuáles son las diferencias entre Nested Loop, Hash Join y Merge Join? 	Inmersión profunda en el plan de ejecución de SQL
6	<ul style="list-style-type: none"> ● ¿Cómo cancelo el estado de solo lectura? ● ¿Qué debo hacer cuando la base de datos entra en el estado de solo lectura? 	Un clúster de GaussDB(DWS) se convierte en solo lectura y está bloqueado y los datos no se pueden escribir
7	<ul style="list-style-type: none"> ● ¿Cómo puedo recopilar estadísticas? 	ANALYZE ANALYSE

Clasificación	Preguntas frecuentes	URL de la página
8	<ul style="list-style-type: none"> ● ¿Cómo configuro el optimizador? ● ¿Cómo puedo habilitar o deshabilitar nestloop? ● ¿Cómo puedo habilitar o deshabilitar mergejoin? ● ¿Cuáles son los parámetros que afectan al plan de ejecución? 	<p>Configuración del método de optimizador</p>
9	<ul style="list-style-type: none"> ● ¿Cómo cambio la zona horaria de la base de datos? ● ¿Cómo cambio la zona horaria? 	<p>¿Cómo cambiar la zona horaria predeterminada de una base de datos cuando su tiempo es diferente del tiempo del sistema?</p>
10	<ul style="list-style-type: none"> ● ¿Cómo puedo consultar definiciones de función? ● ¿Cómo uso las funciones de consulta de privilegios de acceso? ● ¿Cómo puedo consultar definiciones de vista? 	<p>Funciones de información del sistema</p>
11	<ul style="list-style-type: none"> ● ¿Cuáles son las especificaciones técnicas? ● ¿Cuáles son los tamaños de tabla de particiones compatibles? ● ¿Cuál es el volumen máximo de datos en una sola tabla? ● ¿Cuál es el número máximo de columnas soportadas por una tabla? 	<p>Especificaciones técnicas</p>
12	<ul style="list-style-type: none"> ● ¿Cuáles son las operaciones del desarrollador? ● ¿Cómo especifico si se habilita el uso del optimizador de un framework distribuido? 	<p>Opciones de desarrollador</p>

Gestión de clúster

Tabla 1-3 Gestión de clúster

Clasificación	Preguntas frecuentes	URL de la página
1	<ul style="list-style-type: none"> ● ¿Qué debo hacer cuando el uso del disco es demasiado alto? ● ¿Qué debo hacer si el disco está lleno? ● ¿Qué debo hacer si el clúster pasa a ser de solo lectura? ● ¿Qué debo hacer si se genera una alarma de disco? ● ¿Cómo configuro el umbral del disco? ● ¿Cómo puedo activar la suscripción de alarma de disco? 	<p>Un clúster de GaussDB(DWS) se convierte en solo lectura y está bloqueado y los datos no se pueden escribir</p>
2	<ul style="list-style-type: none"> ● ¿Cómo puedo ver la información básica del clúster? ● ¿Cómo puedo ver la EIP de un clúster? ● ¿Cómo puedo ver la dirección de conexión a la base de datos? ● ¿Cuáles son las variantes de clúster? ● ¿Cuál es la información básica del disco? 	<p>Consulta de detalles del clúster</p>
3	<p>¿Cómo puedo comprar un clúster?</p>	<p>Creación de un clúster</p>
4	<ul style="list-style-type: none"> ● ¿Qué debo hacer si el clúster está desequilibrado? ● ¿Cómo se realiza la conmutación del clúster primario/en espera? 	<p>Realización de una conmutación de retorno principal/en espera</p>

2 Problemas generales

2.1 ¿Por qué son necesarios los almacenes de datos?

Situación actual y requisitos

Muchos datos (pedidos, existencias, materiales y pagos) se generan en los sistemas de operación de negocios y en la base de datos (transaccional) de las empresas.

Los responsables de la toma de decisiones categorizan y analizan los datos para la toma de decisiones empresariales.

Dificultades

La categorización y el análisis de datos implican el acceso simultáneo a los datos en múltiples tablas de base de datos. Es decir, múltiples tablas que se actualizan mediante diferentes transacciones pueden bloquearse al mismo tiempo, lo que puede causar complicaciones a los sistemas de base de datos durante las horas pico.

- El bloqueo de varias tablas aumenta la latencia de las consultas complejas.
- Las transacciones que están actualizando las tablas de la base de datos están bloqueadas, causando retrasos o interrupciones.

Solución

Los almacenes de datos sobresalen en la agregación y asociación de datos, por lo que los usuarios extraen más datos, obtienen más información y toman mejores decisiones. La minería requiere consultas complejas que involucren datos en varias tablas.

El proceso ETL copia los datos de las bases de datos de operaciones empresariales a los almacenes de datos para su análisis e informática. Los datos se pueden agregar desde múltiples sistemas de operaciones empresariales en un almacén de datos para una mejor asociación, análisis e información procesable.

Los almacenes de datos y las bases de datos estándar orientadas a las transacciones, como Oracle, Microsoft SQL Server y MySQL, utilizan diferentes modos de diseño. Los almacenes de datos están optimizados en términos de agregación y asociación de datos, pero es posible que no se garanticen las funciones o el rendimiento de la transacción o la adición y

eliminación de datos. Por lo tanto, los almacenes de datos y las bases de datos se aplican a diferentes escenarios. Las bases de datos transaccionales se dedican al procesamiento de transacciones (operación comercial de las empresas), mientras que los almacenes de datos sobresalen en el análisis de datos complejos. En conclusión, las bases de datos se aplican a las actualizaciones de datos, mientras que los almacenes de datos se aplican al análisis de datos.

2.2 ¿Cuáles son las diferencias entre un almacén de datos y la plataforma de big data de Hadoop?

La plataforma de big data de Hadoop puede considerarse como un sistema de almacenamiento de datos de próxima generación. Tiene las características de los almacenes de datos modernos y es ampliamente utilizado por las empresas. Debido a la escalabilidad de MPP, el sistema de almacenamiento de datos basado en MPP a veces se clasifica como una plataforma de big data.

Sin embargo, los almacenes de datos difieren mucho de la plataforma Hadoop en función y experiencia del usuario en diferentes escenarios. Para más detalles, consulte la siguiente tabla.

Tabla 2-1 Comparación de características entre los almacenes de datos y la plataforma de big data de Hadoop

Características	Hadoop	Almacén de datos
Número de nodos informáticos	1000s	Máximo 256
Volumen de datos	Más de 10 PB	Máximo 10 PB
Tipo de datos	Relacional, semi-relacional, no estructurado (voz, imágenes y vídeo)	Solo relacional
Latencia	Mediano a alto	Baja
Ecosistema de aplicaciones	Innovador/IA	Tradicional/BI
API de desarrollo de aplicaciones	SQL y otras API de lenguaje de programación, como MapReduce	Base de datos SQL estándar
Escalabilidad	Ilimitado, con APIs de programación integrales	Limitado, apoyado por UDFs
Soporte de transacciones	Limitado	Datos integrales

Los almacenes de datos y la plataforma Hadoop trabajan juntos en diferentes escenarios. GaussDB(DWS) en la nube pública puede integrarse sin problemas con MRS basado en Hadoop en la nube pública para proporcionar el intercambio de datos SQL sobre Hadoop entre plataformas y servicios. GaussDB(DWS) sirve como un almacén de datos para la gestión de datos masivos mientras disfruta de la apertura, conveniencia e innovación de la plataforma Hadoop. También puede disfrutar de las aplicaciones de capa superior de los

almacenes de datos convencionales, especialmente las aplicaciones de BI, utilizando GaussDB(DWS).

2.3 ¿Por qué debería usar la nube pública GaussDB(DWS)?

Los almacenes de datos convencionales no son prácticos para las empresas más pequeñas debido al alto costo, la selección y adquisición de dispositivos y sistemas que consumen mucho tiempo, y el complejo escalamiento horizontal.

GaussDB(DWS) en la nube pública es la mejor opción:

- Este servicio en la nube de almacenamiento de datos MPP distribuido es muy abierto, eficiente, compatible, escalable, y es fácil de O&M.
- Desarrollado en el núcleo FusionInsight de almacén de datos LibrA, permite a las empresas de la nube pública una experiencia mejor y consistente dentro y fuera de la nube.

FusionInsight LibrA es un sistema de almacenamiento de datos distribuidos de última generación con derechos de propiedad intelectual independientes. Actualmente, es ampliamente utilizado en gobierno, finanzas y operadores. FusionInsight LibrA es compatible con bases de datos Postgres de código abierto, especialmente en sentencias SQL de Oracle y Teradata. Los ingenieros de GaussDB(DWS) han diseñado un núcleo de almacenes híbridos de fila-columna no solo para un análisis más rápido, sino también para el procesamiento de datos, como agregar, eliminar, modificar datos. FusionInsight LibrA cuenta con las tecnologías de optimización de costes y almacén, incluido la computación vectorial de código de máquina y entre/dentro paralelismo para operadores y nodos. Utiliza LLVM para optimizar el código local en los planes de consulta de compilación. Las consultas y análisis de datos más potentes abordan los puntos problemáticos del servicio y mejoran la experiencia del usuario.

- GaussDB(DWS) se puede utilizar fuera de la caja.

La solicitud de GaussDB(DWS) en la nube pública toma solo unos minutos, lo que le libera del lento proceso de búsqueda y compra de almacenes de datos. Esto no solo simplifica la adquisición, sino que también reduce el costo y los requisitos para el uso de almacenes de datos. Las pequeñas y medianas empresas con acceso a GaussDB(DWS) pueden extraer sin problemas los valores de los datos para su desarrollo y formar información procesable.

2.4 ¿Debo elegir GaussDB(DWS) de la nube pública o RDS?

Ambos permiten ejecutar bases de datos relacionales convencionales en la nube y transferir cargas de gestión de bases de datos. Las bases de datos RDS son útiles para OLTP, informes y análisis, pero son menos capaces de manejar operaciones de lectura de una gran cantidad de datos (consultas complejas de solo lectura). GaussDB(DWS) es útil para OLAP al reducir las cargas de trabajo de análisis e informes de grandes conjuntos de datos en un orden de magnitud, gracias a su escala y recursos multinodo y algoritmos optimizados (**almacenamiento de columnas, ejecutores vectorizados y marcos distribuidos**).

Puede escalar un clúster de GaussDB(DWS) para abordar datos y consultas complejos, o para manejar análisis abrumadores e informes de cargas de trabajo que afectan al rendimiento de OLTP.

La siguiente tabla muestra la comparación entre OLTP y OLAP.

Tabla 2-2 Comparación de características entre OLTP y OLAP

Características	RDS para OLTP	GaussDB(DWS) para OLAP
Usuario	Operaciones y gestión de bajo nivel	Tomadores de decisiones y alto gestión
Función	Procesamiento de operación diaria	Análisis y toma de decisiones
Diseño	Por aplicación	Por tema
Colaboración	Último, detallado, bidimensional, discreto	Histórico, integrado, multidimensional, unificado
Acceso	Decenas de registros de lectura y escritura	Millones de registros de lectura
Cobertura	Operaciones sencillas de lectura/escritura	Consultas complejas
Tamaño de la base de datos	Cientos de GB	TB o PB

2.5 ¿Cuándo debo usar GaussDB(DWS) y MRS?

MRS funciona mejor con marcos de procesamiento de big data como Apache Spark, Hadoop y HBase, para procesar y analizar conjuntos de datos ultragrandes a través de código personalizado. Le permite controlar las configuraciones del clúster y el software instalado en el clúster.

GaussDB(DWS) funciona mejor con consultas complejas de una gran cantidad de datos estructurados. Su objetivo es reunir datos de diferentes fuentes, como inventario, finanzas y sistema minorista. Para garantizar la coherencia y la precisión de los informes empresariales, GaussDB(DWS) almacena los datos de una manera altamente estructurada. Esta estructura puede crear directamente la regla de coherencia de datos en la tabla de la base de datos. Además, GaussDB(DWS) es altamente compatible con instrucciones SQL estándar y la sintaxis de bases de datos convencionales compatibles con transacciones.

GaussDB(DWS) se prefiere cuando se desea realizar una consulta compleja de una gran cantidad de datos estructurados con alto rendimiento.

2.6 ¿Qué es la cuota de usuario?

Para los servicios de HUAWEI CLOUD, las cuotas limitan el número de recursos disponibles para los usuarios. Si necesita más, envíe un ticket de servicio para aumentar sus cuotas. Una

vez aprobada, actualizaremos su cuota de recursos en consecuencia y le enviaremos una notificación. Para obtener más información sobre las operaciones de cuota, consulte [Cuotas](#).

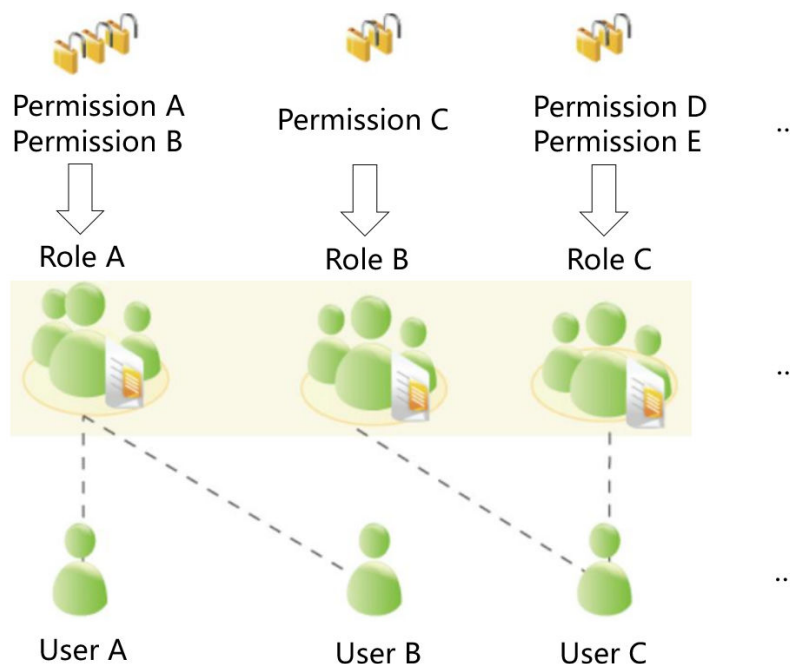
2.7 ¿Cuáles son las diferencias entre usuarios y roles?

Los usuarios y los roles se comparten dentro de todo el clúster, pero sus datos no se comparten. Es decir, un usuario puede conectarse a cualquier base de datos, pero una vez que la conexión es exitosa, cualquier usuario puede acceder solo a la base de datos declarada en la solicitud de conexión.

- Un rol es un grupo de permisos. Por lo general, los roles se utilizan para ordenar los permisos. Los usuarios se utilizan para gestionar permisos y realizar operaciones.
- Un rol puede heredar permisos de otros roles. Todos los usuarios de un grupo de usuarios heredan automáticamente los permisos de operación del rol del grupo.
- En una base de datos, los permisos de los usuarios provienen de roles.
- Un grupo de usuarios es un grupo de usuarios que tienen el mismo permiso.
- Un usuario puede ser considerado como un rol con el permiso de inicio de sesión.
- Un rol puede considerarse como un usuario sin el permiso de inicio de sesión.

Los permisos proporcionados por Gauss (DWS) incluyen los permisos de operación y mantenimiento para los componentes en el plano de gestión. Puede asignar diferentes permisos a los usuarios según sea necesario. El plano de gestión utiliza roles para una mejor gestión de permisos. Puede seleccionar los permisos especificados y asignarlos a los roles de una manera unificada. De esta manera, los permisos se pueden ver y gestionar de manera centralizada.

En la siguiente figura se muestran las relaciones entre permisos, roles y usuarios en la gestión unificada de permisos.



GaussDB(DWS) proporciona varios permisos. Seleccione y asigne permisos a diferentes usuarios en función de los escenarios de servicio. Un rol se le pueden asignar uno o más permisos.

Después de que un rol es otorgado a un usuario a través de **GRANT**, el usuario tendrá todos los permisos del rol. Se recomienda que los roles se utilicen para conceder permisos de manera eficiente. Un usuario solo tiene permisos para sus propias tablas, pero no tiene permisos para las tablas de otros usuarios en sus esquemas.

- Al rol A se le asignan permisos de operación A y B. Después de asignar el rol A al usuario A y al usuario B, el usuario A y el usuario B pueden obtener permisos de operación A y B.
- Al rol B se le asigna el permiso de operación C. Después de asignar el rol B al usuario C, el usuario C puede obtener permisos de operación C.
- Al rol C se le asignan permisos de operación D y E. Después de asignar el rol C al usuario C, el usuario C puede obtener los permisos de operación D y E.

2.8 ¿Cómo verifico el tiempo de creación de un usuario de base de datos?

Método 1:

Cuando se crea un usuario de base de datos de GaussDB(DWS), si el momento en que el usuario entra en vigor (**VALID BEGIN**) es el mismo que el tiempo de creación del usuario, y el momento en que el usuario entra en vigor no se ha cambiado, puede comprobar la columna **valbegin** en la vista **PG_USER** para comprobar la hora de creación del usuario.

A continuación se presenta un ejemplo:

Cree **jerry** de usuario y establezca su hora de inicio de validez en su hora de creación actual.

```
CREATE USER jerry PASSWORD 'password' VALID BEGIN '2022-05-19 10:31:56';
```

Consulte los usuarios en la vista **PG_USER**. La columna **valbegin** indica el momento en que **jerry** entró en vigor, es decir, el momento en que se creó jerry.

```
SELECT * FROM PG_USER;
```

username	usesysid	usecreatedb	usesuper	usecatupd	userepl	passwd
Ruby	10	t	t	t	t	*****
				default_pool	0	
dbadmin	16393	f	f	f	f	*****
				default_pool	0	
jack	451897	f	f	f	f	*****
				default_pool	0	
emma	451910	f	f	f	f	*****
				default_pool	0	

```
jerry | 457386 | f | f | f | f | ***** |
2022-05-19 10:31:56+08 | | default_pool | 0 |
| | | |
(5 rows)
```

Método 2:

Compruebe la columna **passwordtime** en el catálogo del sistema **PG_AUTH_HISTORY**. Esta columna indica la hora en que se creó la contraseña inicial del usuario. Solo los usuarios con permisos de administrador del sistema pueden acceder al catálogo.

```
select roloid, min(passwordtime) as create_time from pg_auth_history group by
roloid order by roloid;
```

A continuación se presenta un ejemplo:

Consulte la vista **PG_USER** para obtener el OID del **jerry** de usuario, que es el nombre **457386**. Consulte la columna **passwordtime** para obtener el tiempo de creación del **jerry** de usuario, que es el de **2022-05-19 10:31:56**.

```
select roloid, min(passwordtime) as create_time from pg_auth_history group by
roloid order by roloid;
 roloid | create_time
-----+-----
      10 | 2022-02-25 09:53:38.711785+08
    16393 | 2022-02-25 09:55:17.992932+08
   451897 | 2022-05-18 09:42:26.897855+08
   451910 | 2022-05-18 09:46:33.152354+08
   457386 | 2022-05-19 10:31:56.037706+08
(5 rows)
```

2.9 Regiones y AZ

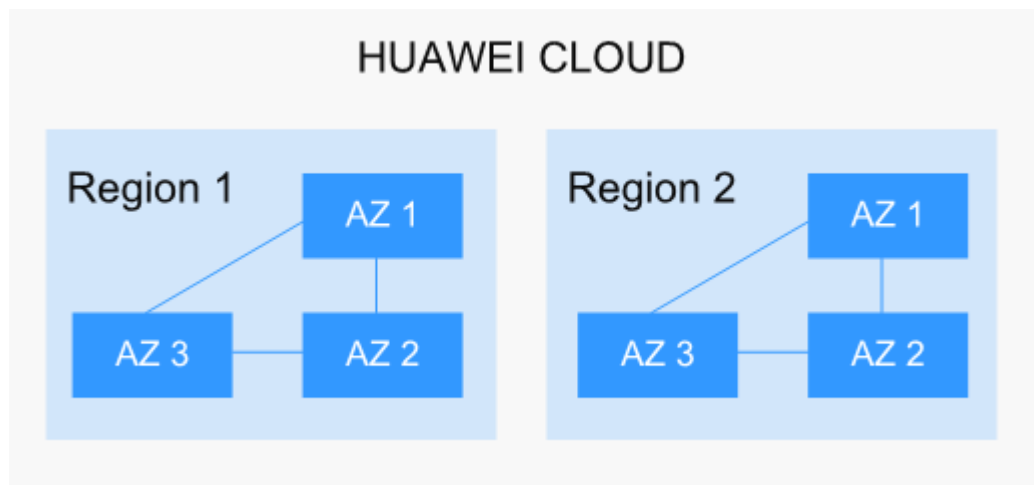
Conceptos

Una región y una zona de disponibilidad (AZ) identifican la ubicación de un centro de datos. Puede crear recursos en regiones y zonas de disponibilidad.

- **Regiones** se definen en términos de ubicación geográfica y latencia de red. Cada región tiene sus propios servicios públicos compartidos (ECS, EVS, OBS, VPC, EIP e IMS). Las regiones son comunes o dedicadas. Una región común proporciona servicios en la nube comunes disponibles para todos los tenants. Una región dedicada proporciona servicios de un tipo específico o solo para tenants específicos.
- Una **AZ** contiene uno o más centros de datos físicos. Cada AZ tiene instalaciones independientes de refrigeración, extinción de incendios, antihumedad y electricidad. La computación, la red, el almacenamiento y otros recursos en una AZ se dividen lógicamente en múltiples clústeres. Las AZ de una región están interconectadas a través de fibra óptica de alta velocidad, por lo que los sistemas implementados en las AZ pueden lograr una mayor disponibilidad.

La **Figura 2-1** muestra la relación entre las regiones y las AZ.

Figura 2-1 Regiones y AZ



Huawei Cloud ofrece servicios en todo el mundo. Puede seleccionar una región y AZ según sea necesario. Para obtener más información, consulte [Productos y servicios globales](#).

¿Cómo selecciono una región?

Al seleccionar una región, tenga en cuenta lo siguiente:

- Localización

Seleccione una región cercana a usted o a sus usuarios de destino para una menor latencia de red y un acceso más rápido. Las regiones en China continental proporcionan la misma infraestructura, calidad de red BGP y operaciones y configuraciones de recursos. Por lo tanto, si sus usuarios objetivo se encuentran en China continental, no es necesario tener en cuenta las diferencias de latencia de la red al seleccionar una región.

Los países y regiones fuera de China continental, como Bangkok y Hong Kong, proporcionan servicios a los usuarios fuera de China continental. Si usted o sus usuarios objetivo están en China continental, estas regiones no se recomiendan debido a la alta latencia de acceso.

- Si usted o sus usuarios objetivo se encuentran en la región de Asia Pacífico (excepto China continental), seleccione la región de **AP-Bangkok** o **AP-Singapore**.
- Si usted o sus usuarios objetivo están en África, seleccione la región de **AF-Johannesburg**.
- Si usted o sus usuarios objetivo están en Europa, seleccione la región **EU-Paris**.

- Precios de recursos

Esto varía según la región. Para obtener más información, consulte los [detalles de precios](#).

¿Cómo selecciono una AZ?

Tenga en cuenta sus requisitos para DR y latencia de red al seleccionar un AZ:

- Despliegue recursos en diferentes AZ en la misma región para fines de recuperación ante desastres.
- Despliegue recursos en la misma AZ para una latencia mínima.

Regiones y puntos de conexión

Cuando usa recursos con invocaciones a API, debe especificar el punto de conexión regional. Para obtener más información sobre las regiones y puntos de conexión de Huawei Cloud, consulte [Regiones y puntos de conexión](#).

2.10 ¿Mis datos están seguros en GaussDB(DWS)?

Sí. En la era del big data, los datos se han convertido en un activo fundamental. La nube pública adherirá al compromiso asumido a lo largo de los años de no tocar sus aplicaciones o datos, ayudándole a proteger sus activos principales. Este es nuestro compromiso con los usuarios y la sociedad, sentando las bases para el éxito empresarial de la nube pública y sus socios.

GaussDB(DWS) es un sistema de almacenamiento de datos con seguridad de clase de telecomunicaciones para salvaguardar sus datos y privacidad. Además, GaussDB(DWS) en la nube pública ofrece calidad de operador, que puede satisfacer los requisitos de seguridad y privacidad de datos de gobiernos, organizaciones financieras y operadores. Por lo tanto, es ampliamente utilizado por diversas industrias. GaussDB(DWS) de la nube pública ganó la siguiente autenticación de seguridad:

- Laboratorio interno de seguridad cibernética (ICSL) en cumplimiento de los estándares de seguridad cibernética emitidos por las autoridades del UK.
- Certificación de Evaluación de Privacidad y Seguridad (PSA) para cumplir con los requisitos de la UE en materia de seguridad de datos y privacidad.

Seguridad de datos de servicio

GaussDB(DWS) se basa en la infraestructura de software de la nube pública, incluidos ECS y OBS. Tanto ECS como OBS ganaron la Certificación de nube confiable emitida por China Data Center Alliance en 2017.

Los datos de servicio de los usuarios de GaussDB(DWS) se almacenan en los ECS del clúster. Ni los usuarios ni los administradores de O&M de la nube pública pueden iniciar sesión en los ECS.

El sistema operativo de los ECS está reforzado para la seguridad, incluidos el refuerzo del núcleo, la instalación del último parche, el control de permisos, la gestión de puertos y el protocolo y el puerto anti-ataque.

GaussDB(DWS) proporciona medidas de seguridad completas, como políticas de contraseñas, autenticación, gestión de sesiones, gestión de permisos de usuario y auditoría de bases de datos.

Seguridad de datos de instantáneas

Las copias de seguridad de GaussDB(DWS) son instantáneas almacenadas en OBS. OBS ha pasado la autenticación de seguridad en la nube de confianza de China Data Center Alliance. Es compatible con funciones de control de permisos de acceso, acceso a claves y encriptación de datos. Los datos de instantáneas de GaussDB(DWS) solo se pueden usar para copias de seguridad y restauración de datos y ningún usuario puede acceder a ellos. Los administradores de GaussDB(DWS) pueden ver el espacio OBS ocupado por los datos de instantáneas en la consola de GaussDB(DWS) y las facturas de la nube pública .

Seguridad de acceso a la red

GaussDB(DWS) está completamente aislado entre las redes de capa 2 y capa 3 para cumplir con los requisitos de seguridad de los usuarios gubernamentales y financieros.

- GaussDB(DWS) se despliega en el entorno de ECS dedicado al tenant, que no se comparte con otros tenants. Por lo tanto, la fuga de datos debido al uso compartido de recursos informáticos es imposible físicamente.
- ECS en un clúster de GaussDB(DWS) se aíslan a través de VPC, evitando que los ECS sean descubiertos e intruidos por otros tenants.
- La red se divide en el plano de servicio y el plano de gestión. Los dos planos están aislados físicamente, lo que garantiza la seguridad de la red.
- Los tenants pueden personalizar de forma flexible el grupo de seguridad y las reglas de acceso.
- Software de aplicaciones externas de acceso GaussDB(DWS) a través de SSL.
- Los datos importados de OBS están cifrados.

2.11 ¿Cómo se asegura GaussDB(DWS)?

GaussDB(DWS) utiliza IAM y VPC para controlar el acceso del usuario y aislar la red del clúster. El acceso al clúster es a través de SSL y paquete de cifrado. Además, GaussDB(DWS) admite la autenticación de certificados digitales bidireccionales.

Los OS de nodo de cada clúster están reforzados para permitir un acceso válido solo a los archivos de sistema operativo.

2.12 ¿Puedo modificar el grupo de seguridad de un clúster de GaussDB(DWS)?

Sí. Después de crear un clúster de almacén de datos, no se puede cambiar su grupo de seguridad. Sin embargo, puede agregar, eliminar o modificar reglas del grupo de seguridad actual.

Para editar el grupo de seguridad del clúster:

1. Inicie sesión en la consola de gestión de GaussDB(DWS).
2. En el árbol de navegación de la izquierda, haga clic en **Clusters**.
3. En la lista de clústeres, busque el clúster objetivo y haga clic en el nombre del clúster. Se muestra **Basic Information**.
4. Busque el parámetro **Grupo de seguridad** y haga clic en el nombre del grupo de seguridad para cambiar a la página de **Grupos de seguridad** en la consola de VPC, en la que puede establecer el grupo de seguridad.

2.13 ¿Cómo se relacionan LibrA, GaussDB A y GaussDB(DWS)?

GaussDB(DWS) es una base de datos de procesamiento de datos en línea construida sobre la infraestructura y plataforma de nube pública. Evolucionó a partir de la GaussDB A de Huawei

(originalmente llamado FusionInsight LibrA). GaussDB A es un software de base de datos implementado en máquinas físicas. Para obtener más información, visite los siguientes sitios web:

- Versión 6.5.1 o anterior <https://support.huawei.com/enterprise/en/cloud-computing/gaussdb-200-pid-21407429>
- Versión 8.0.0: <https://support.huawei.com/enterprise/en/cloud-computing/gaussdb-a-pid-250949677>

2.14 ¿Qué es una base de datos/almacén de datos/lago de datos/lakehouse?

La evolución de Internet y el IoT producen volúmenes masivos de datos. Estos datos necesitan ser gestionados, usando conceptos como base de datos, almacén de datos, lago de datos y lakehouse. ¿Cuáles son estos conceptos? ¿Cuáles son sus relaciones? ¿Cuáles son los productos y soluciones específicos? Este documento le ayuda a entenderlos con la comparación.

Base de datos

Una base de datos es donde los datos están organizados, almacenados y gestionados por la estructura de datos.

Las bases de datos se han utilizado en computadoras desde la década de 1960s, con los dos modelos de datos predominantes (jerárquico y de red), y los datos y aplicaciones eran muy interdependientes. Esto limita las aplicaciones de bases de datos.

Una base de datos generalmente se refiere a una base de datos relacional. Una base de datos relacional organiza los datos con un modelo relacional, es decir, los datos se almacenan en filas y columnas. Por lo tanto, los datos de la base de datos están bien estructurados e independientes con baja redundancia. En 1970, las bases de datos relacionales nacieron para separar completamente los datos de las aplicaciones de software y se han convertido en una parte indispensable de los sistemas informáticos convencionales. Las bases de datos relacionales son la base de los productos de base de datos de todos los proveedores, con soporte de API relacional incluso si una base de datos no es relacional.

Las bases de datos relacionales procesan transacciones básicas y rutinarias usando OLTP, como las transacciones bancarias.

Almacén de datos

El crecimiento de las bases de datos ha facilitado el crecimiento de los datos. OLAP explora la relación entre los datos y extrae más valor de datos. Sin embargo, es difícil compartir datos entre diferentes bases de datos, y la integración y el análisis de datos también enfrentan grandes desafíos.

Para superar estos desafíos para las empresas, Bill Inmon, propuso la idea de almacenamiento de datos en 1990. El almacén de datos se ejecuta en una arquitectura de almacenamiento única para realizar OLAP en una gran cantidad de datos OLTP acumulados a lo largo de los años. De esta manera, las empresas pueden obtener información valiosa de datos masivos de forma rápida y eficaz para tomar decisiones informadas. Gracias a los almacenes de datos, la industria de la información ha evolucionado de sistemas operativos basados en bases de datos relacionales a sistemas de apoyo a la toma de decisiones.

A diferencia de una base de datos, un almacén de datos tiene las siguientes características:

- Un almacén de datos utiliza temas. Está diseñado para soportar varios servicios, con datos procedentes de datos operativos dispersos. Por lo tanto, los datos requeridos deben extraerse de múltiples fuentes de datos heterogéneas, procesarse e integrarse, y reorganizarse por tema.
- Un almacén de datos soporta principalmente el análisis de decisiones empresariales y las operaciones involucradas se centran en la consulta de datos. Por lo tanto, mejora la velocidad de consulta y reduce el costo total de propiedad (TCO) al optimizar las estructuras de tablas y los modos de almacenamiento.

Tabla 2-3 Comparación entre almacenes de datos y bases de datos

Dimensión	Almacén de datos	Base de datos
Escenario de aplicación	OLAP	OLTP
Fuente de datos	Múltiple	Único
Normalización de datos	Esquemas denormalizados	Esquemas estáticos altamente normalizados
Acceso a los datos	Operaciones de lectura optimizadas	Operaciones de escritura optimizadas

Lago de datos

Los datos son un activo importante para las empresas. Los datos de producción y operaciones se guardan y se destilan en políticas de gestión efectivas.

El lago de datos hace eso. Es un gran almacén de datos que almacena de forma centralizada datos estructurados y no estructurados. Puede almacenar datos sin procesar de múltiples fuentes y tipos de datos, lo que significa que se puede acceder a los datos, procesarlos, analizarlos y transmitirlos sin ser estructurados primero. El lago de datos ayuda a las empresas a completar rápidamente el análisis federado de fuentes de datos heterogéneas y a explorar el valor de los datos.

Un lago de datos es en esencia una solución que consiste en una arquitectura de almacenamiento de datos y herramientas de procesamiento de datos.

- La **arquitectura de almacenamiento** debe ser escalable y lo suficientemente confiable como para almacenar datos masivos de cualquier tipo (datos estructurados, semiestructurados, no estructurados).
- Los dos tipos de **processing tools** tienen funciones separadas:
 - El primer tipo: migra datos al lago, incluida la definición de orígenes, la formulación de políticas de sincronización, el movimiento de datos y la compilación de catálogos.
 - El segundo tipo entonces utiliza esos datos, incluidos el análisis, la minería, y el uso de ellos. El lago de datos debe estar equipado con capacidades de amplio alcance, como la gestión integral del ciclo de vida de los datos y datos, el análisis de datos diversificado y la adquisición y liberación seguras de datos. Estas herramientas de gobernanza de datos ayudan a garantizar la calidad de los datos, que puede verse

comprometida por la falta de metadatos y convertir el lago de datos en un pantano de datos.

Ahora, con el big data y la IA, los datos en el lago son aún más valiosos y juegan nuevos roles. Representa más capacidades empresariales. Por ejemplo, el lago de datos puede centralizar la gestión de datos, ayudando a las empresas a construir modelos de operación más optimizados. También proporciona otras capacidades empresariales, como análisis de predicción y modelos de recomendación. Estos modelos pueden estimular un mayor crecimiento.

Al igual que cualquier otro almacén y lago, uno almacena bienes, o datos, de una fuente, mientras que el otro almacena agua, o datos, de muchas fuentes.

Tabla 2-4 Comparación entre lagos de datos y almacenes de datos

Dimensión	Lago de datos	Almacén de datos
Escenario de aplicación	Análisis exploratorio (aprendizaje automático, descubrimiento de datos, perfilado, predicción)	Análisis de datos (basado en datos estructurados históricos)
Costo	Bajo costo inicial, alto costo posterior	Alto costo inicial, bajo costo posterior
Calidad de los datos	Datos brutos masivos que deben limpiarse y normalizarse antes de su uso	Datos de alta calidad que pueden utilizarse como base de los hechos
Usuario objetivo	Científicos de datos y desarrolladores de datos	Analista comercial

Lakehouse

Aunque los escenarios de aplicación y las arquitecturas de un almacén de datos y un lago de datos son diferentes, pueden cooperar para resolver problemas. Un almacén de datos almacena datos estructurados y es ideal para un soporte rápido de BI y toma de decisiones, mientras que un lago de datos almacena datos en cualquier formato y puede generar mayor valor mediante la minería de datos. Por lo tanto, su convergencia puede traer más beneficios a las empresas en algunos escenarios.

Un lakehouse, la convergencia de un almacén de datos y un lago de datos, tiene como objetivo permitir la movilidad de datos y agilizar la construcción. La clave de la arquitectura del lakehouse es permitir el libre flujo de datos/metadatos entre el almacén de datos y el lago de datos. Los datos de valor explícito en el lago pueden fluir o incluso ser utilizados directamente por el almacén. Los datos de valor implícito en el almacén también pueden fluir al lago para el almacenamiento a largo plazo a un bajo costo y para la minería de datos futura.

Solución de datos inteligentes

DataArts es una plataforma de habilitación de datos que ayuda a grandes agencias gubernamentales y empresas a personalizar soluciones inteligentes de gestión de recursos de datos. Esta solución puede importar datos de todos los dominios en el lago de datos,

eliminando los silos de datos, liberando el valor de los datos y potenciando la transformación digital basada en datos.

DataArts Studio cuenta con el lago de datos inteligente FusionInsight como su núcleo. A su alrededor hay motores informáticos como la base de datos, el almacén de datos, el lago de datos y la plataforma de datos. Proporciona habilitación integral de datos, que abarca la recopilación de datos, agregación, cómputo, gestión de activos y apertura de datos.

Los motores de lagos, almacenes y bases de datos permiten la construcción ágil de lagos de datos, la migración rápida de bases de datos de GaussDB y el análisis en tiempo real del almacén de datos. Para obtener más información, visite:

- Base de datos
 - Las bases de datos relacionales incluyen: [Relational Database Service \(RDS\)](#), [GaussDB\(for MySQL\)](#), [GaussDB](#), [RDS for PostgreSQL](#) y [RDS for SQL Server](#).
 - Base de datos no relacional: [Document Database Service \(DDS\)](#), GaussDB NoSQL (including [Influx](#), [Redis](#), [Mongo](#), [Cassandra](#))
- Almacén de datos: [GaussDB\(DWS\)](#)
- Integración de lagos de datos y almacenes: [MapReduce Service \(MRS\)](#), [Data Lake Insight \(DLI\)](#) .
- Centro de gobernanza de datos: [DataArts Studio](#).

3 Gestión de clúster

3.1 ¿Qué hago si la creación de un clúster de GaussDB(DWS) ha fallado?

Solución de problemas

Compruebe que tiene suficiente cuota para crear el clúster.

Asistencia técnica

Si no se puede identificar el fallo, envíe un ticket de servicio para informar del problema: Inicie sesión en la consola de gestión y elija **Service Tickets > Create Service Ticket**.

3.2 ¿Cómo puedo borrar y recuperar el espacio de almacenamiento?

Después de eliminar los datos almacenados en los almacenes de datos de GaussDB(DWS), es posible que se generen datos sucios debido a que el espacio en disco no se libera. Esto da como resultado un desperdicio de espacio en disco y deteriora el rendimiento de la creación y restauración de instantáneas. A continuación se describe el impacto en el sistema y la operación posterior para borrar el espacio en disco:

Puntos que vale la pena mencionar durante la limpieza y recuperación de espacio de almacenamiento:

- Los datos innecesarios deben eliminarse para liberar el espacio de almacenamiento.
- Las operaciones de lectura y escritura frecuentes pueden afectar al uso adecuado de la base de datos. Por lo tanto, es una buena práctica limpiar y recuperar el espacio de almacenamiento cuando no está en horas pico.
- El tiempo de borrado de datos depende de los datos almacenados en la base de datos.

Realice los siguientes pasos para borrar y recuperar el espacio de almacenamiento:

1. Conéctese a la base de datos. Para obtener más información, consulte [Métodos de conexión a un clúster](#).

2. Ejecute el siguiente comando para borrar y recuperar el espacio de almacenamiento:

VACUUM FULL;

De forma predeterminada, se eliminan las tablas en las que el usuario actual tiene el permiso. Otras tablas son omitidas.

Se muestra la siguiente información una vez que se borra el espacio:

```
VACUUM
```

NOTA

- **VACUUM FULL** recupera todo el espacio de fila caducado, sin embargo, requiere un bloqueo exclusivo en cada tabla que se procesa y puede tardar mucho tiempo en completarse en tablas de base de datos grandes y distribuidas. Se recomienda hacer **VACUUM FULL** a las tablas especificadas. Si desea hacer **VACUUM FULL** a toda la base de datos, se recomienda hacerlo durante el mantenimiento de la base de datos.
- La información estadística se perderá si utiliza el parámetro **FULL**. Para recopilar las estadísticas, agregue la palabra clave **ANALYZE**, por ejemplo, **VACUUM FULL ANALYZE;**

Para obtener más información acerca de **VACUUM**, vea **VACUUM** en la *Referencia de sintaxis de SQL*.

3.3 ¿Puedo cambiar mis nodos de clúster a otra región después de la compra?

No. Los nodos de clúster no se pueden utilizar en todas las regiones.

En su lugar, cancele el orden en la región original, coloque un nuevo orden en la región deseada y cree un clúster.

3.4 ¿Por qué se redujo el almacenamiento usado después de la expansión horizontal?

Causas posibles

Si no ejecuta **VACUUM** para borrar y recuperar el espacio de almacenamiento antes de la expansión horizontal, es posible que los datos eliminados de GaussDB(DWS) no liberen el espacio ocupado en disco.

Durante la expansión horizontal, el sistema redistribuye los datos porque el volumen de datos de servicio en los nodos originales es significativamente mayor que el de los nodos recién añadidos. Cuando se inicia la redistribución, el sistema realiza automáticamente **VACUUM** para liberar el espacio de almacenamiento. Esto provoca una gran caída en la capacidad.

Procedimiento de manejo


Se recomienda borrar y recuperar periódicamente el espacio de almacenamiento ejecutando **VACUUM FULL** para evitar la expansión de los datos.

Si el espacio de almacenamiento utilizado sigue siendo grande después de ejecutar **VACUUM FULL**, analice si la variante de clúster existente cumple con los requisitos de servicio. Si no, ampliar el clúster.

3.5 ¿Cómo puedo ver las métricas de nodo (CPU, memoria y uso de disco)?

Puede ver la capacidad utilizada de la CPU, la memoria y los discos de un clúster en la consola de gestión de Cloud Eye. Realice los siguientes pasos para ver la información:

Paso 1 Inicie sesión en la consola de GaussDB(DWS) y haga clic en **Viewing Metric** junto a un clúster.

Paso 2 Haga clic en  para volver a la página **Cloud Service Monitoring** y cambie a la página **Data Warehouse Node** y haga clic en **View Metric** a la derecha del nodo correspondiente para ver su uso del disco.

----Fin

3.6 ¿GaussDB(DWS) soporta el nodo único para un entorno de aprendizaje?

Sí. En GaussDB(DWS), puede crear un clúster de almacén de datos híbrido en modo independiente. Si el nombre de la variante de nodo seleccionado contiene **h1** (por ejemplo, **dwsx2.h1.xlarge.2.c6**), el almacén de datos híbrido solo admite el despliegue independiente, que no proporciona capacidades de HA. El coste de almacenamiento se puede reducir a la mitad. Un almacén de datos independiente se puede restaurar mediante la reconstrucción automática de ECS, y su fiabilidad de datos está garantizada por el mecanismo multicopia de EVS. Es menos costoso que otras variantes y es una buena opción para servicios ligeros.

3.7 ¿GaussDB(DWS) es compatible con BMS?

Sí. Puede enviar un ticket de servicio para solicitar las variantes de BMS. GaussDB(DWS) utiliza ECS solo en Huawei Cloud por defecto.

3.8 ¿Cómo se calcula el espacio en disco o la capacidad de GaussDB(DWS)?

1. Capacidad total del disco de GaussDB (DWS): Para un clúster de tres nodos, si cada nodo es de 320 GB, la capacidad total es de 960 GB. Cuando se almacenan datos de 1 GB, GaussDB(DWS) almacena datos de 1 GB en dos nodos debido a la duplicación, un mecanismo de seguridad, ocupando así un total de 2 GB de espacio. Como resultado, se ocupa más de 2 GB de espacio si se calculan metadatos e índices. Por lo tanto, un clúster de tres nodos con una capacidad total de 960 GB puede almacenar datos de 480 GB. Este mecanismo garantiza la seguridad de los datos.

Al comprar nodos en la consola, se le factura por la capacidad disponible de un nodo. Por ejemplo, el espacio real de **dws.m3.xlarge** es de 320 GB y el espacio disponible mostrado es de 160 GB, el espacio que se le facturará.

2. Comprobar el uso del disco con el nodo único.

De manera similar, si la capacidad total es de 960 GB y hay tres nodos de datos, la capacidad de disco de cada nodo es de 320 GB.

Inicie sesión en la consola de Gauss (DWS) y elija **Monitoring > Node Monitoring > Overview** para ver el uso de discos y otros recursos en cada nodo.

📖 NOTA

- El espacio en disco que se muestra en la página **Node Management** es la capacidad total de todos los discos, es decir, los discos del sistema y los discos de datos, en el clúster del almacén de datos. El espacio de disco que se muestra en la página **Overview** es solo el espacio disponible para almacenar datos de tabla en el clúster. Además, las tablas en el clúster del almacén de datos tienen copias de respaldo, estas copias también ocupan el almacenamiento en disco.
- Si el clúster es de solo lectura y se genera una alarma por espacio en disco insuficiente, amplíe la capacidad del clúster siguiendo las instrucciones proporcionadas en [Expansión de un clúster](#).

3.9 ¿Cuáles son las bases de datos gaussdb y postgres de GaussDB(DWS)?

Las bases de datos **gaussdb** y **postgres** son bases de datos integradas de GaussDB(DWS). Puede crear esquemas y tablas en ellos. Sin embargo, se recomienda volver a crear una base de datos y crear esquemas y tablas en la nueva base de datos.


3.10 ¿Cómo configuro el número máximo de sesiones al agregar una regla de alarma en Cloud Eye?

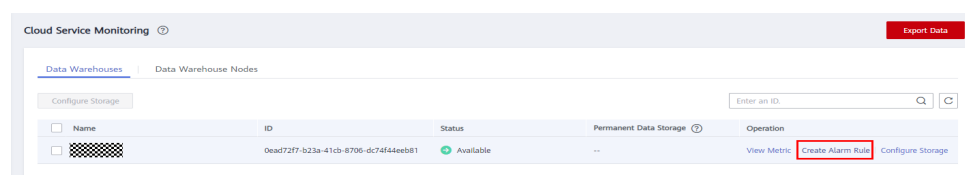
Después de conectarse a una base de datos, ejecute la siguiente instrucción SQL para comprobar el número máximo de sesiones simultáneas globalmente:

```
show max_active_statements;
```

Vaya a la consola de Cloud Eye y establezca el umbral entre el 70% y el 80% del valor obtenido. Por ejemplo, si el valor de **max_active_statements** es **80**, establezca el umbral en **56** (80 x 70%).

Procedimiento:

1. Vaya a la página **Clusters** en la consola de gestión de GaussDB(DWS).
2. Haga clic en **View Metric** en la columna **Operation** del clúster de destino para ir a la consola de Cloud Eye.
3. Haga clic en  en la esquina superior izquierda de la página mostrada y haga clic en **Create Alarm Rule** del clúster de destino.



4. Establezca **Method** en **Configure manually**, **Metric Name** en **Session Count**, **Alarm Policy** en **56**, y **Alarm Severity** en **Major**. A continuación, haga clic en **Create**.

The screenshot shows the configuration interface for an alarm policy in the GaussDB(DWS) console. Key elements include:

- Name:** alarm-hjvd
- Description:** (Empty text box, 0/256 characters)
- Enterprise Project:** default
- Resource Type:** Data Warehouse Service
- Dimension:** Data Warehouses
- Monitoring Scope:** Specific resources
- Monitored Object:** (Checkered icon)
- Method:** 'Configure manually' is selected and highlighted with a red box.
- Alarm Policy Table:**

Metric Name	Alarm Policy	Alarm Severity	Operation
Session Count	Raw d... 3 consecuti... >= 56 One day	Major	

3.11 ¿Cómo puedo determinar si un clúster utiliza la arquitectura x86 o Arm?

Procedimiento

- Paso 1** Inicie sesión en la consola de gestión de GaussDB(DWS).
- Paso 2** Elija **Clusters**. Todos los clústeres se mostrarán de forma predeterminada.
- Paso 3** En la lista de clústeres, haga clic en el nombre de un clúster para ir a la página **Cluster Information**. En el área **Basic Information**, compruebe las especificaciones de nodo del clúster.

The screenshot displays the 'Basic Information' section for a cluster. Key specifications include:

- Cluster ID:** dws2est_1234
- Cluster Version:** 8.1.1.300
- Node Flavor:** dws2.xlarge
- Current Specifications:** Cloud | 4 vCPUs | 32 GB Memory | 200 GB Ultra-high I/O
- Nodes:** 3
- Enterprise Project:** default
- Network:** Region: cn-north-7c, VPC: vpc-172, Subnet: cn-north-7c
- Storage/Backup Capacity:** Storage: Ultra-high I/O, Used/Allocated: 4.85/600 GB, Backup: Free: 0/600 GB

- Paso 4** Determine la arquitectura del clúster en función de las especificaciones del nodo. En la siguiente tabla se describen las especificaciones.

Tabla 3-1 Descripción de variante

Nodo	Núcleos de vCPU	Memoria (GB)	Arquitectura	Tipo
dws2.olap.4xlarge.m6	16	128	x86	ECS/storage-compute decoupling EVS
dws2.olap.8xlarge.m6	32	256	x86	ECS/storage-compute decoupling EVS
dws2.olap.16xlarge.m6	64	512	x86	ECS/storage-compute decoupling EVS
dws2.olap.4xlarge.kc1	16	64	Arm	ECS/storage-compute decoupling EVS
dws2.olap.4xlarge.km1	16	128	Arm	ECS/storage-compute decoupling EVS
dws2.olap.6xlarge.km1	24	192	Arm	ECS/storage-compute decoupling EVS
dws2.olap.8xlarge.km1	32	256	Arm	ECS/storage-compute decoupling EVS
dws2.olap.12xlarge.km1	48	384	Arm	ECS/storage-compute decoupling EVS
dws2.m6.4xlarge.8	16	128	x86	ECS/EVS
dws2.m6.8xlarge.8	32	256	x86	ECS/EVS
dws2.m6.16xlarge.8	64	512	x86	ECS/EVS
dws2.km1.4xlarge.8	16	128	Arm	ECS/EVS
dws2.km1.6xlarge.8	24	192	Arm	ECS/EVS
dws2.km1.8xlarge.8	32	256	Arm	ECS/EVS
dws2.km1.12xlarge.8	48	384	Arm	ECS/EVS
dws2.km1.xlarge	4	32	Arm	ECS/EVS
dws2.kc1.4xlarge	16	64	Arm	ECS/EVS
dws2.olap.4xlarge.i3	16	128	x86	ECS/Local passthrough
dws2.olap.8xlarge.i3	32	256	x86	ECS/Local passthrough
dws2.olap.16xlarge.i3	64	512	x86	ECS/Local passthrough
dws2.olap.4xlarge.ki1	16	64	Arm	ECS/Local passthrough

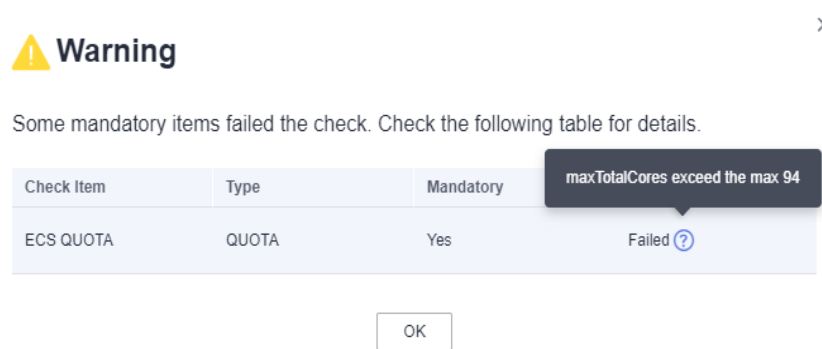
Nodo	Núcleos de vCPU	Memoria (GB)	Arquitectura	Tipo
dws2.olap.8xlarge.k1l	32	128	Arm	ECS/Local passthrough
dws2.olap.16xlarge.k1l	64	228	Arm	ECS/Local passthrough
dws2.physical.k1lne.4xlarge	128	512	Arm	BMS
dws2.physical.k1lne.4xlarge.a	128	512	Arm	BMS
dws2.physical.c6sd.6xlarge	104	768	x86	BMS
dws2.physical.c6sd.6xlarge.7	104	768	x86	BMS
dws2.physical.c6sd.6xlarge.7.cbg	104	768	x86	BMS
dws2.physical.c6sd.6xlarge.a.7	104	768	x86	BMS
dws2.physical.io6.3xlarge.e.4a	104	384	x86	BMS
dws2.physical.c6d.6xlarge.3a.cbg	88	768	x86	BMS
dws2.physical.c6sd.6xlarge.a.7.cbg	104	768	x86	BMS
dws2.physical.c6sd.6xlarge.1.cbg	104	768	x86	BMS

---Fin

3.12 ¿Qué debo hacer si falla la comprobación de expansión horizontal?

Síntoma

Después de hacer clic en **OK** durante la expansión o agregar nodos inactivos, se muestra un mensaje de advertencia y no puede ir al siguiente paso.



Análisis de las causas

Antes de enviar una tarea de expansión, el sistema comprueba elementos como las cuotas de recursos y los permisos de IAM. Los elementos anormales impedirán que se envíen tareas de expansión para evitar fallas de expansión.

Solución

- Si la comprobación de la cuota falla, compruebe si la cuota de recursos es suficiente. Si los nodos disponibles son insuficientes, haga clic en **Increase Quota** para enviar una orden de trabajo para aumentar la cuota de nodos.
- Permisos de IAM: Este problema se produce cuando un usuario solo tiene una cuenta de IAM. En este caso, los permisos como los permisos de VPC y EVC/BMS deben asignarse a la cuenta de IAM. O bien, el usuario puede usar la cuenta principal para realizar la expansión hacia fuera.
- Si no hay elementos anormales pero el cuadro de diálogo sigue existiendo, póngase en contacto con el soporte técnico.

3.13 ¿Cuándo debo agregar CN o ampliar un clúster?

Introducción a la concurrencia de CN

CN es la abreviatura de Coordinator Node. Un CN es un componente importante de GaussDB(DWS) y está estrechamente relacionado con los usuarios. Proporciona interfaces a aplicaciones externas, optimiza los planes de ejecución globales, distribuye los planes de ejecución a DataNodes y resume y procesa los resultados de la ejecución. Un CN es una interfaz para las aplicaciones externas. La capacidad de concurrencia de la CN determina la concurrencia de servicio.

La concurrencia de CN se determina mediante los siguientes parámetros:

- **max_connections**: especifica el número máximo de conexiones simultáneas a la base de datos. Este parámetro afecta a la capacidad de procesamiento simultáneo del clúster. El valor predeterminado depende de las especificaciones del clúster. Para obtener más información, consulte [Gestión de conexiones a bases de datos](#).
- **max_active_statements**: especifica el número máximo de trabajos simultáneos. Este parámetro se aplica a todos los trabajos de un CN. El valor predeterminado es **60**, lo que indica que se pueden ejecutar un máximo de 60 trabajos al mismo tiempo. Otros trabajos serán puestos en cola.

¿Agregar los CN o ampliar un clúster?

- **Conexiones insuficientes:** cuando se crea un clúster por primera vez, el número predeterminado de los CN en el clúster es 3, que puede cumplir con los requisitos básicos de conexión del cliente. Si el clúster tiene un gran número de solicitudes simultáneas y el número de conexiones a cada CN es grande, o el uso de CPU de un CN excede su capacidad, se recomienda agregar CN. Para más detalles, véase [CNs](#).
- **Capacidad y rendimiento de almacenamiento insuficientes:** si su empresa crece y tiene requisitos más altos en cuanto a capacidad y rendimiento de almacenamiento, o si la CPU del clúster es insuficiente, le aconsejamos que amplíe el clúster. Para obtener más información, consulte [Expansión de clúster](#).

Con la expansión de los nodos de clúster, se necesitan más CN para cumplir con los requisitos de distribución de GaussDB(DWS). En resumen, la adición de CN no requiere necesariamente expansión de clústeres. Sin embargo, después de la expansión de grupo, es posible que sea necesario agregar los CN.

3.14 ¿Cuáles son los escenarios de cambiar el tamaño de un clúster, cambiar la variante del nodo, ampliar y reducir?

Cambiar el tamaño de un clúster tiene un gran impacto en sus cargas de trabajo. Es similar a migrar un clúster antiguo a uno nuevo, con cambios en los nodos y las especificaciones. Se recomienda realizar operaciones ligeras, como expansión, reducción y cambio de variante. En la siguiente tabla se enumeran los escenarios de aplicación de las opciones de modificación del clúster.

Tabla 3-2 Opciones de modificación de clúster

Opción	Escenario de aplicación	Notas
Expansión horizontal	Si su empresa crece y tiene mayores requisitos de capacidad de almacenamiento y rendimiento, o si la CPU del clúster es insuficiente, se le aconseja que amplíe el clúster.	No se pueden agregar nodos a un almacén de datos híbrido (independiente).
Reducción horizontal	Durante las horas no pico cuando una gran cantidad de capacidad del clúster está inactiva, puede reducir el número de nodos para reducir los costos.	Un almacén de datos híbrido (modo de clúster) no se puede reducir un clúster independiente.

Opción	Escenario de aplicación	Notas
Cambio de la variante del nodo	Esta opción cambia las variantes de clúster (incluida la CPU, la memoria y otros) para cumplir con los requisitos de servicio. No cambia el número de nodos.	Actualmente, solo puede cambiar las variantes de los clústeres de almacén de datos en la nube y los clústeres de almacén de datos en streaming que solo utilizan recursos de ECS y EVS para cómputo y almacenamiento.
Cambio de todas las especificaciones	Puede cambiar el tamaño del clúster cuando: <ul style="list-style-type: none"> ● Los clústeres son clústeres de BMS o no admiten los cambios de variante. ● Desea cambiar la topología del clúster en lugar de expansión o reducción que simplemente agrega nodos o elimina nodos anillo por anillo. ● El clúster está envejecido y desea cambiarlo a un clúster nuevo sin migrar datos. 	Actualmente, solo se admiten clústeres de almacén de datos estándar.

3.15 ¿Cómo debo seleccionar entre un clúster de muchos nodos con variante pequeña y un clúster de tres nodos de variante grande con los mismos núcleos de CPU y memoria?

- De muchos nodos con la variante pequeña:
Si el volumen de datos es pequeño y necesita escalar nodos, pero tiene un presupuesto limitado, puede seleccionar un clúster de varios nodos con la variante pequeña.
Por ejemplo, un clúster con la variante pequeña (dwsx2.h.2xlarge.4.c6) con 8 núcleos y 32 GB de memoria puede proporcionar capacidades informáticas sólidas. El clúster tiene un gran número de nodos y puede procesar altas solicitudes simultáneas. En este caso, solo necesita asegurarse de que la velocidad de red entre nodos es normal para evitar la limitación del rendimiento del clúster.
- De tres nodos con la variante grande:
Si tiene que procesar una gran cantidad de datos, tiene un alto requisito de cómputo y tiene un alto presupuesto, puede seleccionar un clúster de tres nodos con la variante grande.

Por ejemplo, un clúster con la variante grande (dws2.m6.8xlarge.8) con 32 núcleos y 256 GB de memoria tiene una capacidad de procesamiento de CPU más rápida y una memoria más grande, y puede procesar datos más rápidamente. Sin embargo, el clúster tiene nodos limitados, lo que puede causar un bajo rendimiento en escenarios de alta simultaneidad.

3.16 ¿Cuáles son las diferencias entre las SSD en la nube y las SSD locales?

Las SSD en la nube son compatibles con expansión. Por lo tanto, se recomienda usarlas. Las diferencias entre las SSD en la nube y las SSD locales son las siguientes:

- **SSD en la nube**
 - Las SSD en la nube utilizan EVS como medio de almacenamiento y se pueden escalar a medida que crecen los datos. Esto es más flexible.
 - Las SSD en la nube no están vinculadas a la variante de ECS. Por lo tanto, puede ajustar las variantes de las SSD en la nube cuando sea necesario.
- **SSD locales**
 - Las SSD locales utilizan los discos locales de ECS como medio de almacenamiento. Tienen una capacidad fija y un rendimiento más alto, pero no se pueden escalar.
 - Si la capacidad es insuficiente, hay que agregar más nodos para aumentar la capacidad. No se pueden ajustar las variantes de las SSD locales.

3.17 ¿Cuáles son las diferencias entre el almacenamiento de datos en caliente y el almacenamiento de datos en frío?

La mayor diferencia entre el almacenamiento de datos calientes y el almacenamiento de datos fríos radica en los medios de almacenamiento.

- Los datos activos se consultan o actualizan con frecuencia y tienen altos requisitos de tiempo de respuesta de acceso. Se almacena en **DN data disks**.
- Los datos fríos no se actualizan y se consultan ocasionalmente, y no tienen altos requisitos de tiempo de respuesta de acceso. Se almacena en **OBS**.

Los diferentes medios de almacenamiento determinan el costo, el rendimiento y los escenarios de aplicación de los dos modos de almacenamiento, como se muestra en [Tabla 3-3](#).

Tabla 3-3 Diferencias entre almacenamiento de datos en caliente y en frío

Almacena miento	Lectura y escritura	Costo	Capacidad	Escenario de aplicación
Almacenamiento en caliente	Rápida	Alto	Fija y restringida	Este modo es aplicable a escenarios en los que el volumen de datos es limitado y necesita ser leído y actualizado con frecuencia.

Almacenamiento	Lectura y escritura	Costo	Capacidad	Escenario de aplicación
Almacenamiento en frío	Lento	Bajo	Grande e ilimitada	Este modo es aplicable a escenarios como el archivado de datos. Cuenta con costo bajo y capacidad grande.

4 Conexión de bases de datos

4.1 ¿Cómo se comunican las aplicaciones con GaussDB(DWS)?

Para que las aplicaciones se comuniquen con GaussDB(DWS), asegúrese de que las redes entre ellas estén conectadas. En la siguiente tabla se enumeran los escenarios de conexión más comunes.

Tabla 4-1 Comunicación entre aplicaciones y GaussDB(DWS)

Escenario		Descripción	Tipo de conexión compatible
Nube	La aplicación de servicio y GaussDB(DWS) están en la misma VPC en la misma región	Dos direcciones IP privadas en la misma VPC pueden comunicarse directamente entre sí.	<ul style="list-style-type: none"> ● gsq1 ● Data Studio ● JDBC/ODBC Para obtener más modos de conexión, consulte Métodos de conexión a un clúster .
	Aplicaciones de servicio y GaussDB(DWS) están en diferentes VPCs en la misma región	Después de crear una conexión de VPC peering entre dos VPC, las dos direcciones IP privadas pueden comunicarse directamente entre sí.	
	Las aplicaciones de servicio y GaussDB(DWS) están en diferentes regiones	Después de establecer una cloud connection (CC) entre dos regiones, las dos regiones se comunican entre sí con direcciones IP privadas.	

Escenario		Descripción	Tipo de conexión compatible
In situ y en la nube	Las aplicaciones de servicio se despliegan en centros de datos locales y necesitan comunicarse con GaussDB(DWS).	<ul style="list-style-type: none"> ● Utilice el public IP address or domain name de GaussDB(DWS) para la comunicación. ● Use Direct Connect (DC) para la comunicación. 	

La aplicación de servicio y GaussDB(DWS) están en la misma VPC en la misma región

Para garantizar una baja latencia de servicio, se recomienda que despliegue aplicaciones de servicio y GaussDB(DWS) en la misma región. Por ejemplo, si se despliega una aplicación de servicio en un ECS, se recomienda que despliegue el clúster de almacén de datos en la misma VPC que el ECS. De esta manera, la aplicación puede comunicarse directamente con GaussDB(DWS) con una dirección IP de intranet. En este caso, despliegue el clúster de almacén de datos en la misma región y VPC donde reside el ECS.

Por ejemplo, si el ECS se despliega en la **CN-Hong Kong** seleccione **CN-Hong Kong** para el clúster de GaussDB(DWS) y asegúrese de que el clúster de GaussDB(DWS) y el ECS estén ambos en la **VPC1**. La dirección IP privada del ECS es **192.168.120.1** y la dirección IP privada de GaussDB(DWS) es **192.168.120.2**. Por lo tanto, pueden comunicarse entre sí con direcciones IP privadas.

Los puntos clave en la comprobación de comunicación son la regla de salida de ECS y la regla de entrada GaussDB(DWS). El procedimiento de comprobación es el siguiente:

Paso 1 Comprobar las reglas salientes de ECS:

Asegúrese de que la regla de salida del grupo de seguridad ECS permita el acceso. Si no se permite el acceso, consulte la [Configuración de reglas del grupo de seguridad](#).

Action ?	Protocol & Port ?	Type	Destination ?
Allow	All	IPv4	0.0.0.0/0 ?

Paso 2 Comprobar las reglas entrantes de GaussDB(DWS):

Si no se configura ningún grupo de seguridad cuando se crea GaussDB(DWS), la regla de entrada predeterminada permite el acceso TCP desde todas las direcciones IPv4 y el puerto 8000. Para garantizar la seguridad, también puede permitir solo una dirección IP. Para obtener más información, consulte [¿Cómo configuro una lista blanca para proteger clústeres disponibles por una dirección IP pública?](#)

Allow	TCP : 8000	IPv4	0.0.0.0/0 ?
-------	------------	------	-------------

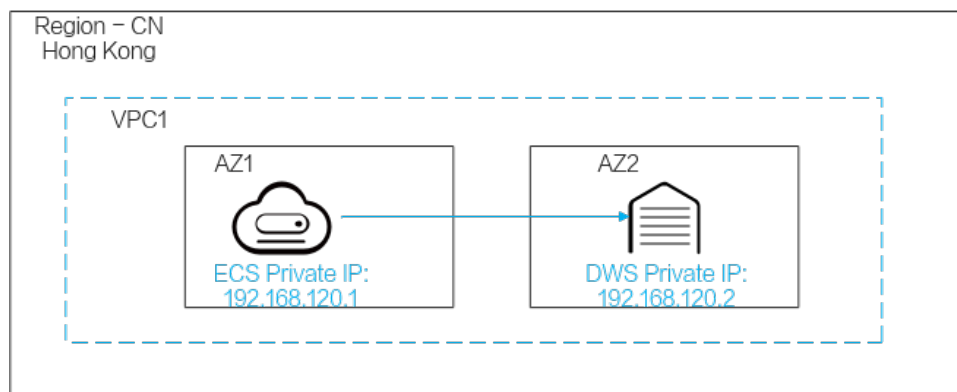
Paso 3 Inicie sesión en el ECS. Si se puede hacer ping la dirección IP interna de GaussDB(DWS), la conexión de red es normal. Si la dirección IP no se puede hacer ping, compruebe la configuración anterior. Si el ECS tiene un firewall, compruebe la configuración del firewall.

----Fin

Ejemplo de uso de gsql para la conexión:

```
gsql -d gaussdb -h 192.168.120.2 -p 8000 -U dbadmin -W password -r
```

Figura 4-1 Acceso por las direcciones IP privadas



Aplicaciones de servicio y GaussDB(DWS) están en diferentes VPCs en la misma región

Para garantizar una baja latencia de servicio, se recomienda que despliegue aplicaciones de servicio y GaussDB(DWS) en la misma región. Por ejemplo, si se despliegan las aplicaciones de servicio en un ECS, se recomienda que despliegue el clúster de almacén de datos en la misma VPC que el ECS. Si se selecciona una VPC diferente para el clúster del almacén de datos, el ECS no puede conectarse directamente a GaussDB(DWS).

Por ejemplo, tanto ECS como GaussDB(DWS) se despliegan en **CN-Hong Kong**, pero ECS está en VPC1 y GaussDB(DWS) está en VPC2. En este caso, necesita crear una **Interconexión de VPC** entre VPC1 y VPC2 para que ECS pueda acceder a GaussDB(DWS) utilizando la dirección IP privada de GaussDB(DWS).

Los puntos clave para comprobar la comunicación son las reglas salientes de ECS, las reglas entrantes de GaussDB(DWS) y las interconexiones de VPC. El procedimiento de comprobación es el siguiente:

Paso 1 Comprobar las reglas salientes de ECS:

Asegúrese de que la regla de salida del grupo de seguridad ECS permita el acceso. Si no se permite el acceso, consulte la **Configuración de reglas del grupo de seguridad**.

Action ?	Protocol & Port ?	Type	Destination ?
Allow	All	IPv4	0.0.0.0/0 ?

Paso 2 Comprobar las reglas entrantes de GaussDB(DWS):

Si no se configura ningún grupo de seguridad cuando se crea GaussDB(DWS), la regla de entrada predeterminada permite el acceso TCP desde todas las direcciones IPv4 y el puerto

8000. Para garantizar la seguridad, también puede permitir solo una dirección IP. Para obtener más información, consulte [¿Cómo configuro una lista blanca para proteger clústeres disponibles por una dirección IP pública?](#)

Allow	TCP : 8000	IPv4	0.0.0.0/0 ?
-------	------------	------	-------------

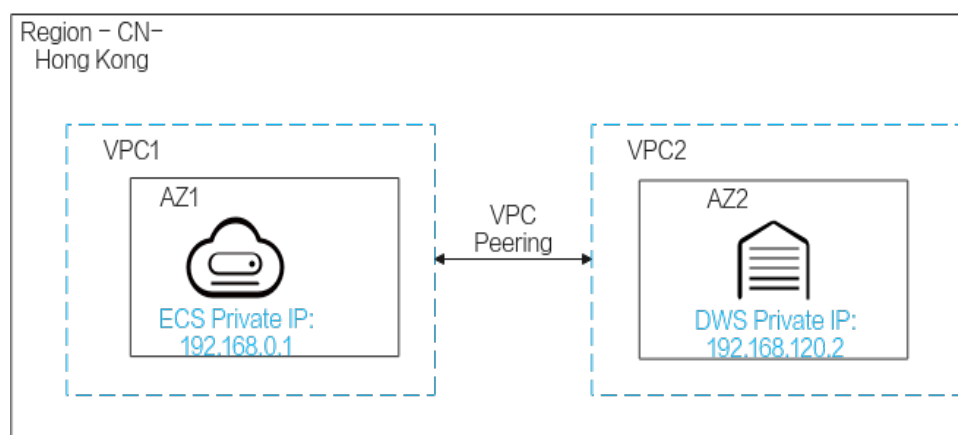
- Paso 3** Crear una **Interconexión de VPC** entre la VPC1 donde está ECS y la VPC2 donde está GaussDB(DWS).
- Paso 4** Inicie sesión en el ECS. Si se puede hacer ping la dirección IP interna de GaussDB(DWS), la conexión de red es normal. Si la dirección IP no se puede hacer ping, compruebe la configuración anterior. Si el ECS tiene un firewall, compruebe la configuración del firewall.

----Fin

Ejemplo de uso de gsql para la conexión:

```
gsql -d gaussdb -h 192.168.120.2 -p 8000 -U dbadmin -W password -r
```

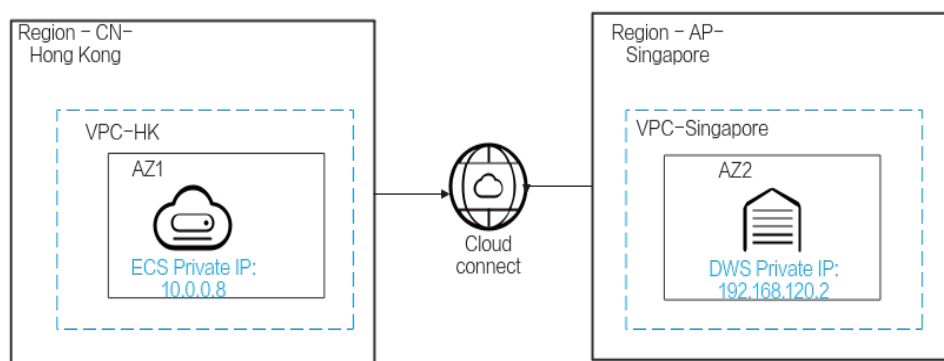
Figura 4-2 Acceso por la interconexión de VPC



Las aplicaciones de servicio y GaussDB(DWS) están en diferentes regiones

Si la aplicación de servicio y GaussDB(DWS) están en diferentes regiones, por ejemplo, ECS está en **CN-Hong Kong** y GaussDB(DWS) está en **AP-Singapore**, necesita establecer un **Cloud Connect** entre las dos regiones para la comunicación.

Figura 4-3 Acceso por la conexión en la nube



Las aplicaciones de servicio se despliegan en centros de datos locales y necesitan comunicarse con GaussDB(DWS).

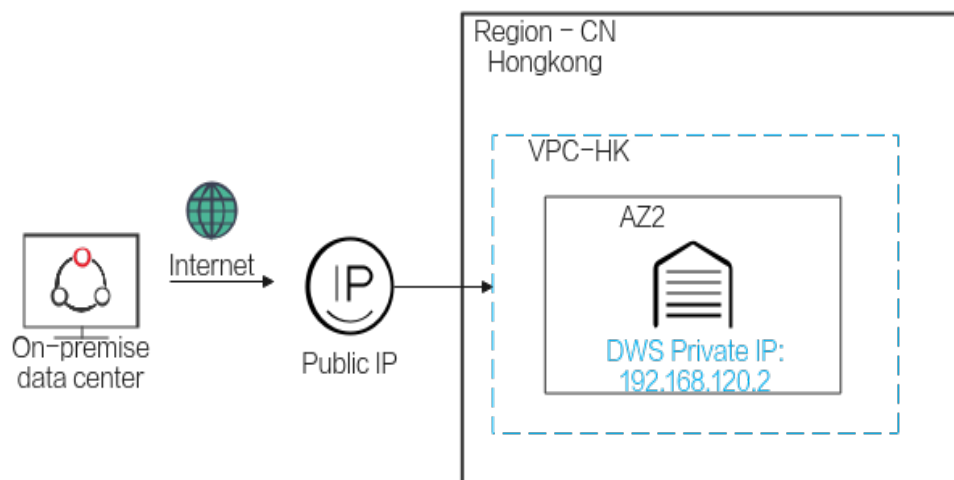
Si las aplicaciones de servicio no están en la nube sino en el centro de datos local, deben comunicarse con GaussDB(DWS) en la nube.

- **Escenario 1:** Las aplicaciones de servicio local se comunican con GaussDB(DWS) con direcciones IP públicas de GaussDB(DWS).

Ejemplo de uso de gsql para la conexión:

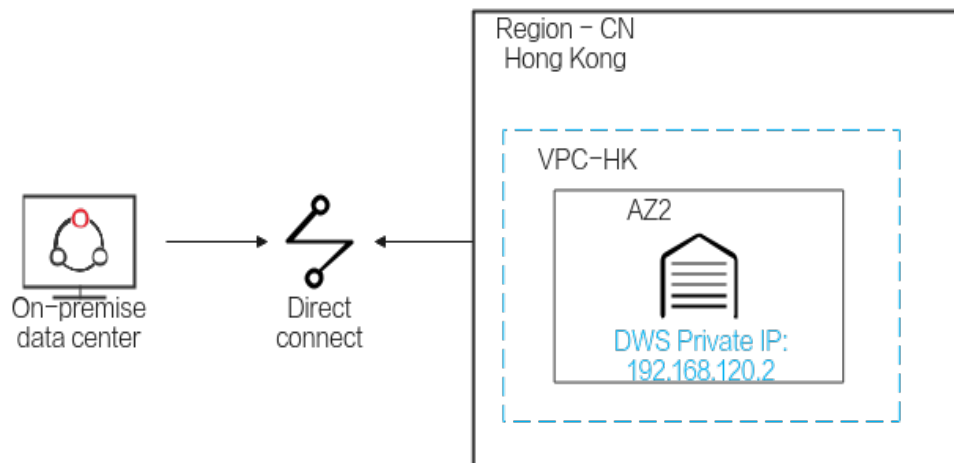
```
gsql -d gaussdb -h public_IP_address -p 8000 -U dbadmin -W password -r
```

Figura 4-4 Acceso por las direcciones IP públicas



- **Escenario 2:** Los servicios locales no pueden acceder a la red externa. En este caso, se requiere **Direct Connect** para la comunicación.

Figura 4-5 Acceso por la conexión directa



4.2 ¿GaussDB(DWS) es compatible con clientes de terceros y controladores JDBC y ODBC?

Sí, pero se recomiendan los clientes y controladores de GaussDB(DWS). A diferencia de los clientes y controladores de PostgreSQL de código abierto, los clientes y controladores de GaussDB(DWS) tienen dos ventajas clave:

- **Endurecimiento de la seguridad:** Los controladores de PostgreSQL solo admiten la autenticación de MD5, pero los controladores de GaussDB(DWS) admiten SHA256 y MD5.
- **Mejora del tipo de datos:** Los controladores de GaussDB(DWS) admiten nuevos tipos de datos `smalldatetime` y `tinyint`.

GaussDB(DWS) soporta clientes de PostgreSQL de código abierto y controladores JDBC y ODBC.

Las versiones de cliente y controlador compatibles son:

- PostgreSQL `psql` 9.2.4 o posterior
- PostgreSQL JDBC Driver 9.3-1103 o posterior
- PSQL ODBC 09.01.0200 o posterior

Para obtener más información sobre cómo usar JDBC/ODBC para conectarse a GaussDB(DWS), véase la [Guía: Desarrollo basado en JDBC o ODBC](#).

4.3 ¿Puedo conectarme a nodos de clúster de GaussDB(DWS) usando SSH?

No se admite el acceso directo. Las VM en la capa inferior de GaussDB(DWS) sirven como nodos de cálculo para el análisis de datos. Acceda a las bases de datos de clúster mediante la dirección de acceso a la red pública o privada en su lugar.

4.4 ¿Qué debo hacer si no puedo conectarme a un clúster de almacenamiento de datos?

Solución de problemas

Verificación:

- Si el estado del clúster es normal.
- Si el comando de conexión, nombre de usuario, contraseña, dirección IP y puerto son correctos.
- Si el tipo de sistema operativo y la versión del cliente son correctos.
- Si el cliente está instalado correctamente.

Si la conexión del clúster falló en la nube pública, verifique los siguientes elementos:

- Si los ECS están en la misma AZ, VPC, subred y grupo de seguridad que el clúster.
- Si las reglas entrantes y salientes del grupo de seguridad son correctas.

Si la conexión del clúster falla a través de Internet, confirme los elementos siguientes:

- Si su red está conectada a Internet.
- Si el firewall bloqueó el acceso.
- Si necesita acceder a Internet a través de un proxy.

Asistencia técnica

Si no se puede identificar el fallo, envíe un ticket de servicio para informar del problema:
Inicie sesión en la consola de gestión y elija **Service Tickets > Create Service Ticket**.

4.5 ¿Por qué no se me notificó un fallo al desvincular la EIP cuando GaussDB(DWS) está conectado por Internet?

Después de que la EIP esté libre, la red puede desconectarse. Sin embargo, la capa TCP no detecta una conexión física defectuosa a tiempo debido a la configuración Keepalive. Como resultado, los clientes gsql, ODBC y JDBC tampoco pueden identificar el error de red a tiempo.

La duración cuando la base de datos envía el mensaje de desconexión al cliente depende de la configuración de Keepalive. El algoritmo específico para calcular la duración es:

keepalive_time + keepalive_probes x keepalive_intvl

Los valores de keepalive afectan a la estabilidad de la comunicación de la red. Ajustarlos a la presión de servicio y a las condiciones de la red.

En Linux, ejecute el comando **sysctl** para modificar los siguientes parámetros:

- net.ipv4.tcp_keepalive_time
- net.ipv4.tcp_keealive_probes
- net.ipv4.tcp_keepalive_intvl

Por ejemplo, si desea cambiar el valor de **net.ipv4.tcp_keepalive_time**, ejecute el siguiente comando para cambiarlo a **120**.

sysctl net.ipv4.tcp_keepalive_time=120

En Windows, modifique la siguiente información de configuración en el Registro **HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters**:


- KeepAliveTime
- KeepAliveInterval
- TcpMaxDataRetransmissions (equivalent to **tcp_keepalive_probes**)

NOTA

Si no puede encontrar los parámetros anteriores en el registro **HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters**, agregue estos parámetros. Abra **Registry Editor**, haga clic con el botón derecho en el área en blanco de la derecha y seleccione **Create > DWORD (32-bit) Value** para agregar estos parámetros.

4.6 ¿Cómo configuro una lista blanca para proteger los clústeres disponibles por una dirección IP pública?

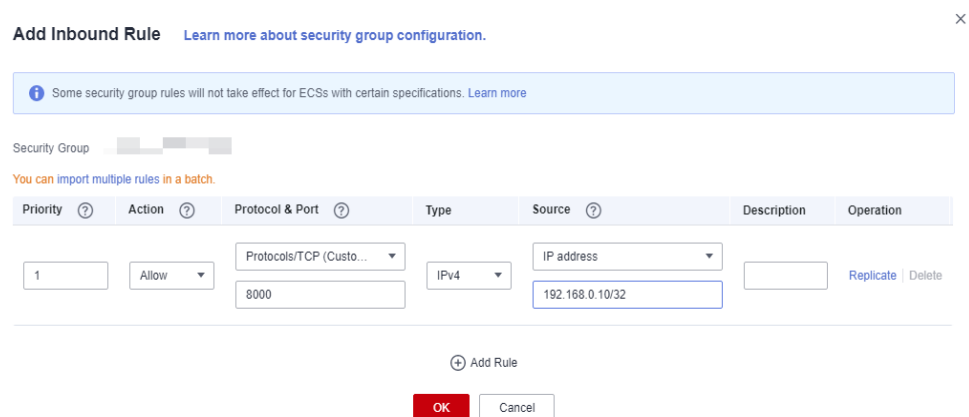
También puede iniciar sesión en la [consola de gestión de VPC](#) para crear manualmente un grupo de seguridad. A continuación, vuelva a la página para crear clústeres de almacén de

datos, haga clic en el botón  junto a la lista desplegable **Security Group** para actualizar la página y seleccione el nuevo grupo de seguridad.

Para permitir que el cliente de GaussDB(DWS) se conecte al clúster, debe agregar una regla de entrada al nuevo grupo de seguridad para conceder el permiso de acceso al puerto de la base de datos del clúster GaussDB(DWS).

- **Protocol: TCP**
- **Port: 8000** Utilice el conjunto de puertos de base de datos al crear el clúster GaussDB(DWS). Este puerto se utiliza para recibir conexiones de cliente a GaussDB(DWS).
- **Source:** Seleccione la **IP address** y utilice la dirección de IP del host del cliente, por ejemplo, **192.168.0.10/32**.

Figura 4-6 Adición de una regla de entrada



Add Inbound Rule [Learn more about security group configuration.](#)

Some security group rules will not take effect for ECSs with certain specifications. [Learn more](#)

Security Group: [redacted]

You can import multiple rules in a batch.

Priority	Action	Protocol & Port	Type	Source	Description	Operation
1	Allow	Protocols/TCP (Custom) 8000	IPv4	IP address 192.168.0.10/32		Replicate Delete

[+](#) Add Rule

Se agregará la lista blanca.

5 Importación y exportación de datos

5.1 ¿Cuáles son las diferencias entre los formatos de datos admitidos por las tablas extranjeras OBS y GDS?

Los formatos de archivo soportados por las tablas externas de OBS y de GDS son los siguientes:

OBS soporta formatos de archivo ORC, TEXT, JSON, CSV, CARBONDATA y PARQUET para la importación de datos y formatos de archivo ORC, CSV y TEXT para la exportación de datos. El formato predeterminado es TEXT.

GDS admite los siguientes formatos de archivo: TEXT, CSV y FIXED. El formato predeterminado es TEXT.

5.2 ¿Cómo puedo importar datos incrementales usando una tabla externa de OBS?

Cuando se utiliza una tabla externa de OBS para importar datos, **INSERT** importa los datos a una tabla física local. Cuando se actualizan los datos de OBS, no es necesario volver a ejecutar la sentencia **INSERT**. Puede utilizar la sentencia **MERGE INTO**.

5.3 ¿Cómo puedo importar datos a GaussDB(DWS)?

GaussDB(DWS) admite la importación eficiente de datos desde múltiples fuentes de datos. A continuación se enumeran los modos típicos de importación de datos. Para obtener más información, consulte la sección **Importación de datos** en la *Guía para desarrolladores de Data Warehouse Service (DWS)*.

- Importación de datos desde OBS
Sube datos a OBS y luego expórtalos a clústeres de GaussDB(DWS). Se admiten formatos de datos como CSV y TEXT.
- Inserción de datos con sentencias **INSERT**
Utilice la herramienta cliente gsql proporcionada por GaussDB(DWS) o el controlador JDBC/ODBC para escribir datos en GaussDB(DWS) desde aplicaciones de capa

superior. GaussDB(DWS) soporta operaciones CRUD a nivel de transacción de base de datos completas. Este es el método más simple y es aplicable a escenarios con pequeño volumen de datos y baja simultaneidad.

- Importación de datos de MRS con MRS como ETL.
- Importación de datos con el comando **COPY FROM STDIN**
Ejecute el comando **COPY FROM STDIN** para escribir datos en una tabla.
- Importación de datos desde un servidor remoto a GaussDB(DWS) mediante GDS
Utilice la función de importación de datos GDS proporcionada por GaussDB(DWS) para importar archivos de datos desde un sistema de archivos común (por ejemplo, un ECS).
- Migración de datos a GaussDB(DWS) mediante CDM

5.4 ¿Cuántos datos de servicio puede almacenar un almacén de datos?

Cada nodo de un clúster de almacén de datos tiene una capacidad de almacenamiento predeterminada de 1.49 TB, 2.98 TB, 4.47 TB, 160 GB, 1.68 TB o 13.41 TB. Un clúster puede alojar de 3 a 256 nodos y la capacidad total de almacenamiento del clúster se expande proporcionalmente a medida que crece la escala del clúster.

Para mejorar la fiabilidad, cada nodo tiene una copia, que ocupa la mitad del espacio de almacenamiento.

El sistema de GaussDB(DWS) realiza copias de seguridad de los datos y genera índices, archivos temporales de caché y registros de ejecución, que ocupan espacio de almacenamiento. Por lo tanto, el espacio de almacenamiento real de cada nodo es aproximadamente la mitad de la capacidad de almacenamiento total.

5.5 ¿Cómo uso \Copy para importar y exportar datos?

GaussDB(DWS) es un servicio totalmente gestionado en la nube. Los usuarios no pueden iniciar sesión en el fondo para importar o exportar datos mediante **COPY**, por lo que la sintaxis **COPY** está deshabilitada. Se recomienda almacenar archivos de datos en OBS y utilizar las tablas externas de OBS para importar datos. Si desea utilizar **COPY** para importar y exportar datos, realice las siguientes operaciones:

1. Coloque el archivo de datos en el cliente.
2. Utilice `gsql` para conectarse al clúster de destino.
3. Ejecute el siguiente comando para importar datos. Introduzca el nombre de directorio y el nombre de archivo del archivo de datos en el cliente y especifique la opción de importación en **with**. El comando es casi el mismo que el comando **COPY** común. Solo necesita agregar una barra invertida (`\`) antes del comando. Cuando los datos se importan correctamente, no se mostrará ninguna notificación.

```
\copy tb_name from '/directory_name/file_name' with(...);
```
4. Ejecute el siguiente comando para exportar datos a un archivo local. Conserve la configuración predeterminada de los parámetros.

```
\copy table_name to '/directory_name/file_name';
```
5. Especifique el parámetro **copy_option** para exportar datos a un archivo CSV.

```
\copy table_name to '/directory_name/file_name' CSV;
```
6. Utilice **with** para especificar parámetros, exportando datos como archivos CSV que utilizan barras verticales (`|`) como delimitadores.


```
\copy table_name to '/directory_name/file_name' with(format 'csv',delimiter
'|') ;
```

5.6 ¿Cómo puedo implementar la importación de tolerancia a fallos entre diferentes bibliotecas de codificación

Para importar datos de la base de datos A (UTF8) a la base de datos B (GBK), puede haber un error de falta de coincidencia de conjunto de caracteres que hace que la importación de datos falle.

Para importar una pequeña cantidad de datos, ejecute el comando `\COPY`. El procedimiento es el siguiente:

Paso 1 Crear las bases de datos A y B. El formato de codificación de la base de datos A es UTF8, y el de la base de datos B es GBK.

```
postgres=> CREATE DATABASE A ENCODING 'UTF8' template = template0;
postgres=> CREATE DATABASE B ENCODING 'GBK' template = template0;
```

Paso 2 Ver la lista de la base de datos. Puede ver las bases de datos creadas A y B.

```
postgres=> \l
```

List of databases					
Name	Owner	Encoding	Collate	Ctype	Access privileges
a	dbadmin	UTF8	C	C	
b	dbadmin	GBK	C	C	
gaussdb	Ruby	SQL_ASCII	C	C	
postgres	Ruby	SQL_ASCII	C	C	
template0	Ruby	SQL_ASCII	C	C	=c/Ruby + Ruby=CTc/Ruby
template1	Ruby	SQL_ASCII	C	C	=c/Ruby + Ruby=CTc/Ruby
xiaodi	dbadmin	UTF8	C	C	

(7 rows)

Paso 3 Cambie a la base de datos A e introduzca la contraseña de usuario. Cree una tabla denominada `test01` e inserte datos en la tabla.

```
postgres=> \c a
Password for user dbadmin:
SSL connection (protocol: TLSv1.3, cipher: TLS_AES_128_GCM_SHA256, bits: 128)
You are now connected to database "a" as user "dbadmin".

a=> CREATE TABLE test01
(
  c_customer_sk          integer,
  c_customer_id          char(5),
  c_first_name           char(6),
  c_last_name            char(8)
)
with (orientation = column,compression=middle)
distribute by hash (c_last_name);
CREATE TABLE
a=> INSERT INTO test01(c_customer_sk, c_customer_id, c_first_name) VALUES (3769,
'hello', 'Grace');
INSERT 0 1
a=> INSERT INTO test01 VALUES (456, 'good');
INSERT 0 1
```

Paso 4 Ejecute el comando `\COPY` para exportar datos de la biblioteca UTF8 en formato Unicode al archivo `test01.dat`.

```
\copy test01 to '/opt/test01.dat' with (ENCODING 'Unicode');
```

Paso 5 Cambie a la base de datos B y cree una tabla con el mismo nombre **test01**.

```
a=> \c b
Password for user dbadmin:
SSL connection (protocol: TLSv1.3, cipher: TLS_AES_128_GCM_SHA256, bits: 128)
You are now connected to database "b" as user "dbadmin".

b=> CREATE TABLE test01
(
  c_customer_sk          integer,
  c_customer_id         char(5),
  c_first_name          char(6),
  c_last_name           char(8)
)
with (orientation = column,compression=middle)
distribute by hash (c_last_name);
```

Paso 6 Ejecute el comando **\COPY** para importar el archivo **test01.dat** a la base de datos B.

```
\copy test01 from '/opt/test01.dat' with (ENCODING
'Unicode' ,COMPATIBLE_ILLEGAL_CHARS 'true');
```

 **NOTA**

- El parámetro de tolerancia a errores **COMPATIBLE_ILLEGAL_CHARS** especifica que se toleran caracteres no válidos durante la importación de datos. Los caracteres no válidos se convierten y, a continuación, se importan a la base de datos. No se muestra ningún mensaje de error. La importación no se interrumpe.
- El formato **BINARY** no es compatible. Cuando se importan datos de dicho formato, se producirá el error "cannot specify bulkload compatibility options in BINARY mode" (no se pueden especificar opciones de compatibilidad de carga masiva en modo **BINARY**).
- El parámetro solo es válido para la importación de datos mediante la opción **COPY FROM**.

Paso 7 Ver datos en la tabla **test01** de la base de datos B.

```
b=> select * from test01;
 c_customer_sk | c_customer_id | c_first_name | c_last_name
-----+-----+-----+-----
          3769 | hello        | Grace        |
          456  | good         |              |
(2 rows)
```

Paso 8 Después de realizar las operaciones anteriores, los datos se importan de la base de datos A (UTF8) a la base de datos B (GBK).

----Fin

5.7 ¿Puedo exportar/importar datos de GaussDB(DWS) desde/a OBS entre regiones?

No. No, GaussDB(DWS) no admite la importación o la exportación de datos de OBS entre regiones. El clúster de GaussDB(DWS) y el OBS deben estar en la misma región.

Por ejemplo, no se admite la exportación directa de datos desde un clúster de GaussDB(DWS) en Beijing 4 a OBS en Beijing 1. Sin embargo, primero puede exportar los datos al OBS en Beijing 4 y luego usar CDM para exportar los datos al OBS en Beijing 1.

5.8 ¿Cómo importo datos de GaussDB(DWS)/Oracle/MySQL/SQL Server a GaussDB(DWS) (Migración de toda la base de datos)?

Los datos heterogéneos se pueden importar a GaussDB(DWS) con CDM. Puede migrar una base de datos completa de Oracle, MySQL, SQL Server o GaussDB(DWS) a una base de datos de GaussDB(DWS). Para obtener más información, consulte [Creación de un trabajo de migración de base de datos completo](#).

También puede almacenar datos en OBS y luego volcar los datos en GaussDB(DWS). Para obtener más información, consulte [Acerca de la importación de datos paralelos desde OBS](#).

5.9 ¿Puedo importar datos por la red pública/externa con GDS?

No. El servidor de GDS y GaussDB(DWS) solo pueden comunicarse entre sí en la intranet. Cada DN en el clúster de GaussDB(DWS) se utiliza para conectarse al servidor de GDS en paralelo para importar una gran cantidad de datos. El servidor de GDS y el clúster deben estar en la misma red. Si GDS se despliega en un servidor sin conexión, el firewall debe estar habilitado y el clúster necesita una EIP. Sin embargo, un clúster solo puede vincularse a una EIP, y la importación de datos con múltiples DN no puede implementarse.

5.10 ¿Cuáles son los factores que afectan al rendimiento de importación de GaussDB(DWS)?

El rendimiento de importación de GaussDB(DWS) se ve afectado por los siguientes factores:

1. Especificaciones del clúster: E/S de disco, rendimiento de red, memoria y especificaciones de CPU
2. Planificación de servicios: tipo de campos de tabla, comprimir y almacén de filas o almacén de columnas
3. Almacenamiento de datos: clúster local, OBS
4. Modo de importación de datos

6 Cuenta, contraseña y permisos

6.1 ¿Cómo implementa GaussDB(DWS) el aislamiento de la carga de trabajo?

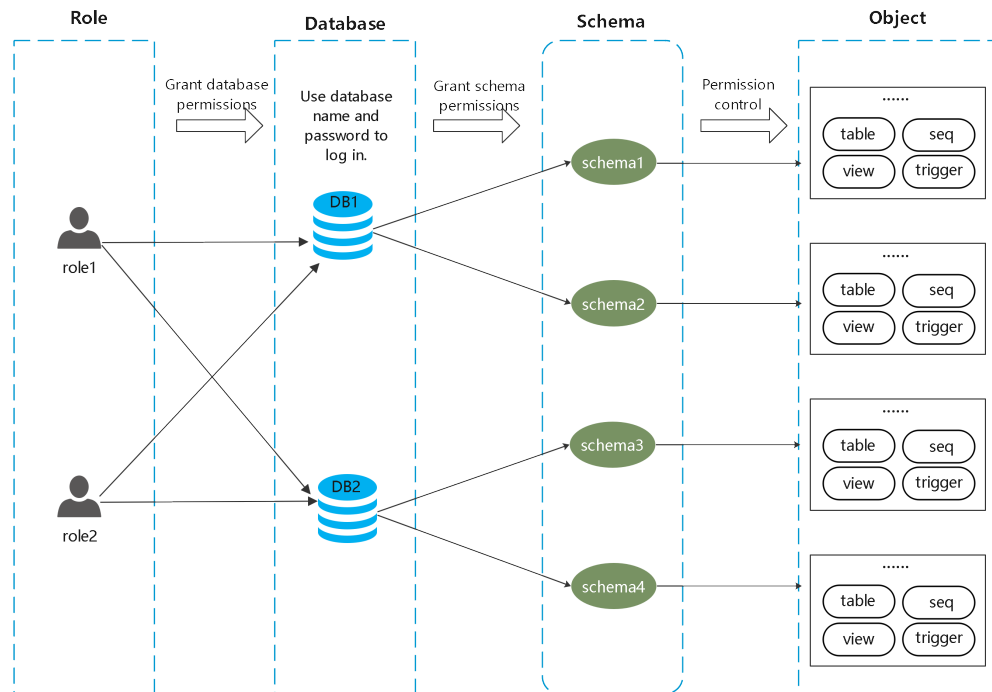
Aislamiento de la carga de trabajo

En GaussDB(DWS), puede aislar cargas de trabajo con configuraciones de bases de datos y esquemas. Sus diferencias son las siguientes:

- Las bases de datos no pueden comunicarse entre sí y compartir muy pocos recursos. Sus conexiones y permisos pueden ser aislados.
- Los esquemas comparten más recursos que las bases de datos. Los permisos de usuario en esquemas y objetos subordinados se pueden configurar de forma flexible mediante la sintaxis **GRANT** y **REVOKE**.

Se recomienda utilizar esquemas para aislar servicios para mayor comodidad y uso compartido de recursos. Se recomienda que los administradores del sistema creen esquemas y bases de datos y, a continuación, asignen los permisos necesarios a los usuarios.

Figura 6-1 Se utiliza para el control de permisos.



Base de datos

Una base de datos es una colección física de objetos de base de datos. Los recursos de las bases de datos diferentes están completamente aislados (excepto algunos objetos compartidos). Las bases de datos se utilizan para aislar cargas de trabajo. Los objetos de diferentes bases de datos no pueden tener acceso entre sí. Por ejemplo, no se puede tener acceso a los objetos de la base de datos B en la base de datos A. Por lo tanto, al iniciar sesión en un clúster, debe conectarse a la base de datos especificada.

Esquema

En una base de datos, los objetos de base de datos se dividen y se aíslan lógicamente en función de los esquemas.

Con la gestión de permisos, puede acceder y operar objetos en diferentes esquemas en la misma sesión. Los esquemas contienen objetos a los que las aplicaciones pueden acceder, como tablas, índices, datos de diversos tipos, funciones y operadores.

Los objetos de base de datos con el mismo nombre no pueden existir en el mismo esquema, pero los nombres de objetos en diferentes esquemas pueden ser los mismos.

```
gaussdb=> CREATE SCHEMA myschema;
CREATE SCHEMA
gaussdb=> CREATE SCHEMA myschema_1;
CREATE SCHEMA

gaussdb=> CREATE TABLE myschema.t1(a int, b int) DISTRIBUTE BY HASH(b);
CREATE TABLE
gaussdb=> CREATE TABLE myschema.t1(a int, b int) DISTRIBUTE BY HASH(b);
ERROR: relation "t1" already exists
gaussdb=> CREATE TABLE myschema_1.t1(a int, b int) DISTRIBUTE BY HASH(b);
CREATE TABLE
```

Los esquemas dividen lógicamente las cargas de trabajo. Estas cargas de trabajo son interdependientes con los esquemas. Por lo tanto, si un esquema contiene objetos, eliminarlo causará errores con la información de dependencia mostrada.

```
gaussdb=> DROP SCHEMA myschema_1;  
ERROR: cannot drop schema myschema_1 because other objects depend on it  
Detail: table myschema_1.t1 depends on schema myschema_1  
Hint: Use DROP ... CASCADE to drop the dependent objects too.
```

Cuando se elimina un esquema, se utiliza la opción **CASCADE** para eliminar los objetos que dependen del esquema.

```
gaussdb=> DROP SCHEMA myschema_1 CASCADE;  
NOTICE: drop cascades to table myschema_1.t1  
gaussdb=> DROP SCHEMA
```

Usuario/Rol

Los usuarios y los roles se utilizan para implementar el control de permisos en el servidor de base de datos (clúster). Son los propietarios y ejecutores de las cargas de trabajo de clúster y gestionan todos los permisos de objetos en clústeres. Un rol no está limitado a una base de datos específica. Sin embargo, cuando inicia sesión en el clúster, debe especificar explícitamente un nombre de usuario para garantizar la transparencia de la operación. Los permisos de un usuario para una base de datos se pueden especificar mediante la gestión de permisos.

Un usuario es el sujeto de permisos. La gestión de permisos es en realidad el proceso de decidir si un usuario puede realizar operaciones en objetos de base de datos.

Gestión de permisos

La gestión de permisos en GaussDB(DWS) se divide en tres categorías:

- Permiso del sistema

Los permisos del sistema también se denominan atributos de usuario, incluidos **SYSADMIN**, **CREATEDB**, **CREATEROLE**, **AUDITADMIN** y **LOGIN**.

Solo se pueden especificar mediante la sintaxis **CREATE ROLE** o **ALTER ROLE**. El permiso **SYSADMIN** se puede conceder y revocar usando **GRANT ALL PRIVILEGE** y **REVOKE ALL PRIVILEGE** respectivamente. Los permisos del sistema no pueden ser heredados por un usuario de un rol y no se pueden conceder mediante **PUBLIC**.

- Permisos

Otorgar los permisos de un rol o usuario a uno o más roles o usuarios. En este caso, cada rol o usuario puede considerarse como un conjunto de uno o más permisos de base de datos.

Si se especifica **WITH ADMIN OPTION**, el miembro puede a su vez conceder permisos en el rol a otros y revocar permisos en el rol también. Si se cambia o revoca un rol o usuario a quien se conceden ciertos permisos, los permisos heredados del rol o usuario también cambian.

Un administrador de base de datos puede concederles permisos y revocarlos de cualquier rol o usuario. Los roles que tienen permiso **CREATEROLE** pueden conceder o revocar la membresía en cualquier rol que no sea un administrador.

- Permiso de objeto

Los permisos de un objeto de base de datos (tabla, vista, columna, base de datos, función, esquema o espacio de tabla) se pueden conceder a un rol o usuario. El comando

GRANT se puede utilizar para conceder permisos a un usuario o rol. Estos permisos concedidos se agregan a los existentes.

Ejemplo de aislamiento de esquema

Ejemplo 1:

De forma predeterminada, el propietario de un esquema tiene todos los permisos sobre los objetos del esquema, incluido el permiso de eliminación. El propietario de una base de datos tiene todos los permisos sobre los objetos de la base de datos, incluido el permiso de eliminación. Por lo tanto, se recomienda controlar estrictamente la creación de bases de datos y esquemas. Cree bases de datos y esquemas como administrador y asigne permisos relacionados a los usuarios.

Paso 1 Asigne el permiso para crear esquemas en la base de datos **testdb** al usuario **user_1** como usuario **dbadmin**.

```
testdb=> GRANT CREATE ON DATABASE testdb to user_1;  
GRANT
```

Paso 2 Cambiar al usuario **user_1**.

```
testdb=> SET SESSION AUTHORIZATION user_1 PASSWORD '*****';  
SET
```

Cree un esquema denominado **myschema_2** en la base de datos **testdb** como **user_1**.

```
testdb=> CREATE SCHEMA myschema_2;  
CREATE SCHEMA
```

Paso 3 Cambiar al administrador **dbadmin**.

```
testdb=> RESET SESSION AUTHORIZATION;  
RESET
```

Cree **table t1** en el esquema **myschema_2** como administrador **dbadmin**.

```
testdb=> CREATE TABLE myschema_2.t1(a int, b int) DISTRIBUTE BY HASH(b);  
CREATE TABLE
```

Paso 4 Cambiar al usuario **user_1**.

```
testdb=> SET SESSION AUTHORIZATION user_1 PASSWORD '*****';  
SET
```

Eliminar la tabla **t1** creada por el administrador **dbadmin** en el esquema **myschema_2** como usuario **user_1**.

```
testdb=> drop table myschema_2.t1;  
DROP TABLE
```

---Fin

Ejemplo 2:

Debido al aislamiento lógico de los esquemas, los objetos de base de datos deben verificarse tanto en el nivel de esquema como en el nivel de objeto.

Paso 1 Conceda el permiso en la tabla **myschema.t1** a **user_1**.

```
gaussdb=> GRANT SELECT ON TABLE myschema.t1 TO user_1;  
GRANT
```

Paso 2 Cambiar al usuario **user_1**.

```
SET SESSION AUTHORIZATION user_1 PASSWORD '*****';  
SET
```

Consultar la tabla **myschema.t1**.

```
gaussdb=> SELECT * FROM myschema.t1;  
ERROR: permission denied for schema myschema  
LINE 1: SELECT * FROM myschema.t1;
```

Paso 3 Cambiar al administrador **dbadmin**.

```
gaussdb=> RESET SESSION AUTHORIZATION;  
RESET
```

Otorgar el permiso de la tabla **myschema.t1** al usuario **user_1**.

```
gaussdb=> GRANT USAGE ON SCHEMA myschema TO user_1;  
GRANT
```

Paso 4 Cambiar al usuario **user_1**.

```
gaussdb=> SET SESSION AUTHORIZATION user_1 PASSWORD '*****';  
SET
```

Consultar la tabla **myschema.t1**.

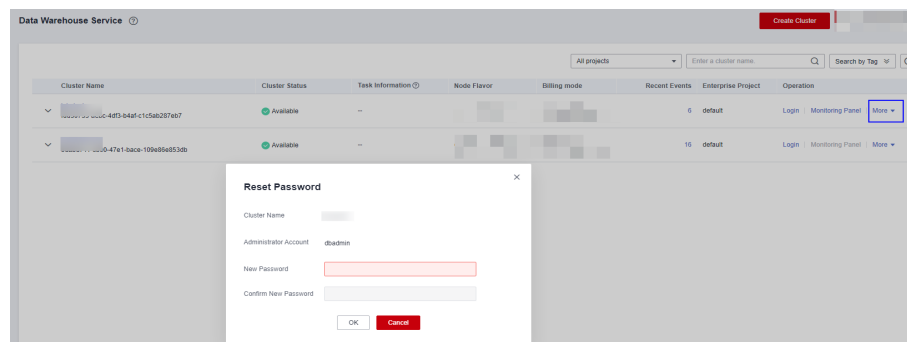
```
gaussdb=> SELECT * FROM myschema.t1;  
 a | b  
----+----  
(0 rows)
```

----Fin

6.2 ¿Cómo cambio la contraseña de una cuenta de base de datos cuando caduca la contraseña?

- Para cambiar la contraseña del administrador de la base de datos **dbadmin**, inicie sesión en la consola y elija **More > Reset Password** en la fila del clúster.

Figura 6-2 Restablecer la contraseña del usuario dbadmin



Por seguridad, los dos parámetros siguientes gestionan las contraseñas de la cuenta. Inicie sesión en la consola, haga clic en el nombre del clúster y cambie a la página de modificación de parámetros para modificar los parámetros.

- **fail_login_attempts**: número máximo de intentos de contraseña incorrectos consecutivos antes de bloquear la cuenta. Ejecute la siguiente instrucción como usuario **dbadmin** para desbloquear la cuenta:

```
ALTER USER user_name ACCOUNT UNLOCK;
```
- **password_effect_time**: período de validez de la contraseña de la cuenta, en días. El valor predeterminado es **90**.

- También puede conectarse a la base de datos y ejecutar el comando **ALTER USER** para cambiar el período de validez de contraseña de una cuenta de base de datos (usuario común y administrador dbadmin).

```
ALTER USER username PASSWORD EXPIRATION 90;
```

6.3 ¿Cómo puedo conceder permisos de tabla a un usuario?

Esta sección describe cómo conceder a los usuarios los permisos SELECT, INSERT, UPDATE o completos de tablas a los usuarios.

Sintaxis

```
GRANT { { SELECT | INSERT | UPDATE | DELETE | TRUNCATE | REFERENCES | TRIGGER |  
ANALYZE | ANALYSE } [, ...]  
| ALL [ PRIVILEGES ] }  
ON { [ TABLE ] table_name [, ...]  
| ALL TABLES IN SCHEMA schema_name [, ...] }  
TO { [ GROUP ] role_name | PUBLIC } [, ...]  
[ WITH GRANT OPTION ];
```

Escenario

Supongamos que hay usuarios **u1**, **u2**, **u3**, **u4**, y **u5** y cinco esquemas con el nombre de estos usuarios. Sus requisitos de permiso son los siguientes:

- El usuario **u2** es un usuario de sólo lectura y requiere el permiso SELECT para la tabla **u1.t1**.
- El usuario **u3** requiere el permiso SELECT para la tabla **u1.t1**.
- El usuario **u3** requiere el permiso UPDATE para la tabla **u1.t1**.
- El usuario **u5** requiere todos los permisos de la tabla **u1.t1**.

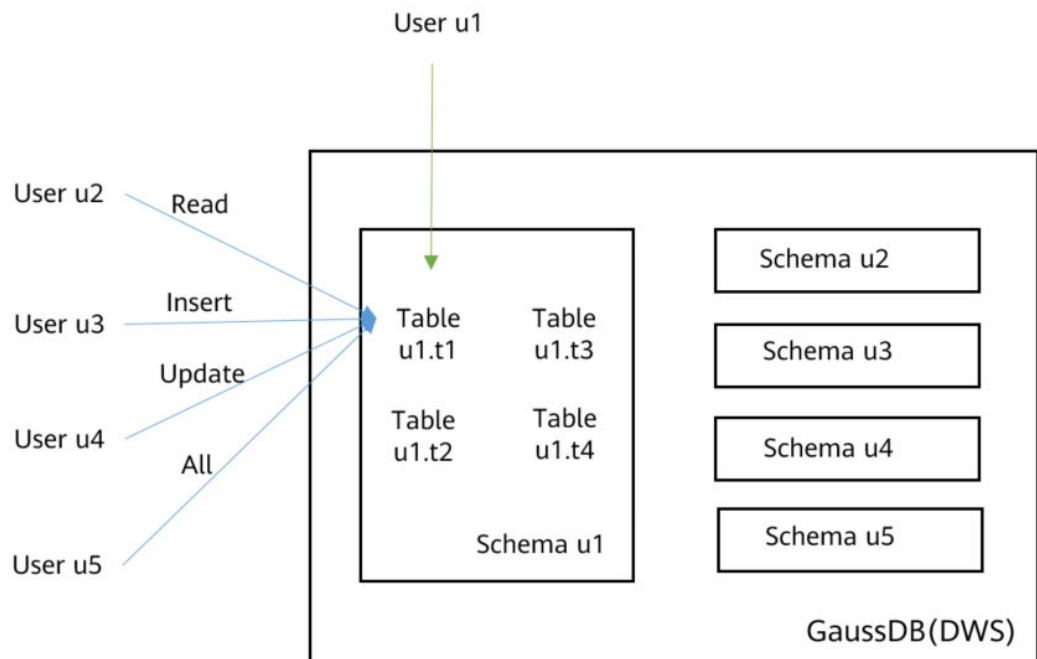


Tabla 6-1 Permisos de la tabla u1.t1

Usuario	Tipo	Sentencia de GRANT	Consultar	Ingresar	Actualizar	Eliminar
u1	Propietario	-	✓	✓	✓	✓
u2	Usuario de solo lectura	GRANT SELECT ON u1.t1 TO u2;	✓	x	x	x
u3	Usuario INSERT	GRANT INSERT ON u1.t1 TO u3;	x	✓	x	x
u4	Usuario UPDATE	GRANT SELECT,UPDATE ON u1.t1 TO u4; AVISO El permiso UPDATE se debe conceder junto con el permiso SELECT, o puede producirse una fuga de información.	✓	x	✓	x
u5	Usuarios con todos los permisos	GRANT ALL PRIVILEGES ON u1.t1 TO u5;	✓	✓	✓	✓

Procedimiento

Realice los siguientes pasos para conceder y verificar permisos:

- Paso 1** Conéctese a su base de datos como **dbadmin**. Ejecute las siguientes instrucciones para crear usuarios **u1** a **u5**. De forma predeterminada, se crearán cinco esquemas con el nombre de los usuarios.

```
CREATE USER u1 PASSWORD '{password}';
CREATE USER u2 PASSWORD '{password}';
CREATE USER u3 PASSWORD '{password}';
CREATE USER u4 PASSWORD '{password}';
CREATE USER u5 PASSWORD '{password}';
```

Paso 2 Crear tabla **u1.t1** en el esquema **u1**.

```
CREATE TABLE u1.t1 (c1 int, c2 int);
```

Paso 3 Insertar dos registros en la tabla.

```
INSERT INTO u1.t1 VALUES (1,2);  
INSERT INTO u1.t1 VALUES (1,2);
```

Paso 4 Conceder permisos de esquema a los usuarios.

```
GRANT USAGE ON SCHEMA u1 TO u2,u3,u4,u5;
```

Paso 5 Otorgar al usuario **u2** el permiso para consultar la tabla **u1.t1**.

```
GRANT SELECT ON u1.t1 TO u2;
```

Paso 6 Iniciar una nueva sesión y conéctese a la base de datos como usuario **u2**. Comprobar que el usuario **u2** puede consultar la tabla **u1.t1** pero no puede escribir en la tabla ni modificarla.

```
SELECT * FROM u1.t1;  
INSERT INTO u1.t1 VALUES (1,20);  
UPDATE u1.t1 SET c2 = 3 WHERE c1 =1;
```

```
gaussdb=> SELECT * FROM u1.t1;  
 c1 | c2  
----+----  
  1 |  2  
  1 |  2  
(2 rows)  
  
gaussdb=> INSERT INTO u1.t1 VALUES (1,20);  
ERROR: permission denied for relation t1  
gaussdb=> UPDATE u1.t1 SET c2 = 3 WHERE c1 =1;  
ERROR: permission denied for relation t1
```

Paso 7 En la sesión iniciada por el usuario **dbadmin**, conceda permisos a los usuarios **u3**, **u4** y **u5**.

```
GRANT INSERT ON u1.t1 TO u3; -- Allow u3 to insert data.  
GRANT SELECT,UPDATE ON u1.t1 TO u4; -- Allow u4 to modify the table.  
GRANT ALL PRIVILEGES ON u1.t1 TO u5; -- Allow u5 to query, insert, modify, and  
delete table data.
```

Paso 8 Inicie una nueva sesión y conéctese a la base de datos como usuario **u3**. Compruebe que el usuario **u3** puede consultar la tabla **u1.t1** pero no puede consultar o modificar la tabla.

```
SELECT * FROM u1.t1;  
INSERT INTO u1.t1 VALUES (1,20);  
UPDATE u1.t1 SET c2 = 3 WHERE c1 =1;
```

```
gaussdb=> SELECT * FROM u1.t1;  
ERROR: permission denied for relation t1  
gaussdb=> INSERT INTO u1.t1 VALUES (1,20);  
INSERT 0 1  
gaussdb=> UPDATE u1.t1 SET c2 = 3 WHERE c1 =1;  
ERROR: permission denied for relation t1
```

Paso 9 Inicie una nueva sesión y conéctese a la base de datos como usuario **u4**. Compruebe que el usuario **u4** puede modificar y consultar la tabla **u1.t1**, pero no puede insertar datos en la tabla.

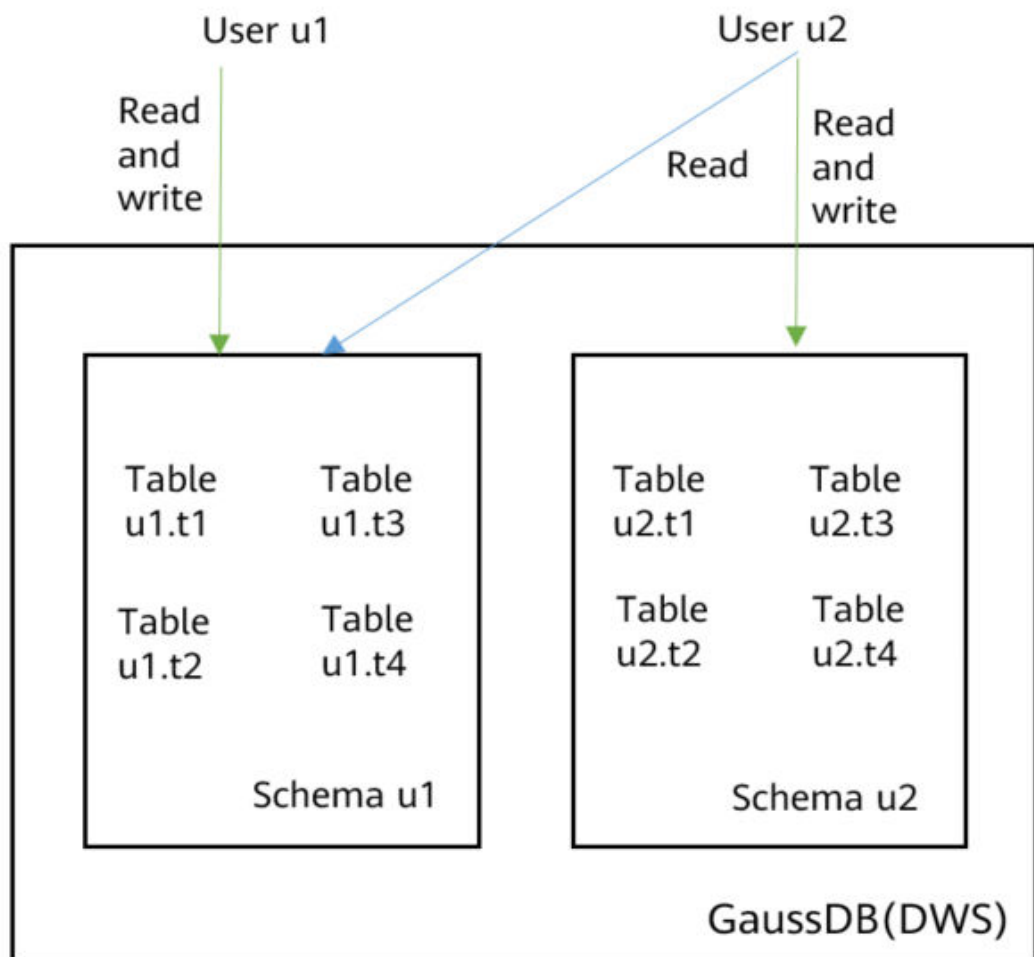
```
SELECT * FROM u1.t1;  
INSERT INTO u1.t1 VALUES (1,20);  
UPDATE u1.t1 SET c2 = 3 WHERE c1 =1;
```


6.4 ¿Cómo puedo conceder permisos de esquema a un usuario?

En esta sección se describe cómo conceder el permiso de consulta para un esquema como ejemplo. Para obtener más información, vea [¿Cómo puedo conceder permisos de tabla a un usuario?](#):

- Permiso para una tabla en un esquema
- Permiso para todas las tablas de un esquema
- Permiso para crear tablas en el esquema

Supongamos que hay usuarios **u1** y **u2**, y dos esquemas nombrados después de ellos. El usuario **u2** necesita acceder a las tablas en el esquema **u1**.



Paso 1 Conéctese a su base de datos como **dbadmin**. Ejecute las siguientes instrucciones para crear usuarios **u1** y **u2**. Se crearán dos esquemas y se les asignará el nombre de los usuarios de forma predeterminada.

```
CREATE USER u1 PASSWORD '{password}';  
CREATE USER u2 PASSWORD '{password}';
```

Paso 2 Cree tablas **u1.t1** y **u1.t2** en el esquema **u1**.

```
CREATE TABLE u1.t1 (c1 int, c2 int);  
CREATE TABLE u1.t2 (c1 int, c2 int);
```

Paso 3 Otorgue el permiso de acceso del esquema **u1** al usuario **u2**.

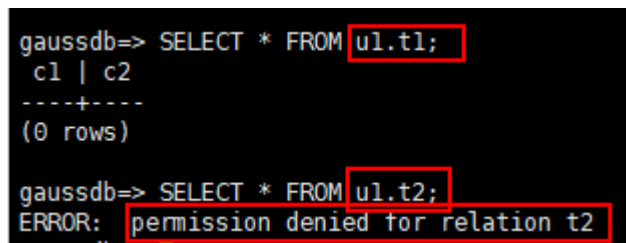
```
GRANT USAGE ON SCHEMA u1 TO u2;
```

Paso 4 Otorgue al usuario **u2** el permiso para consultar la tabla **u1.t1** en el esquema **u1**.

```
GRANT SELECT ON u1.t1 TO u2;
```

Paso 5 Inicie una nueva sesión y conectarse a la base de datos como usuario **u2**. Compruebe que el usuario **u2** puede consultar la tabla **u1.t1** pero no la tabla **u1.t2**.

```
SELECT * FROM u1.t1;  
SELECT * FROM u1.t2;
```



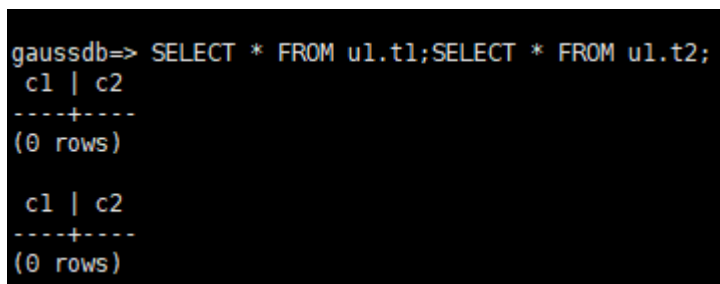
```
gaussdb=> SELECT * FROM u1.t1;  
c1 | c2  
----+----  
(0 rows)  
  
gaussdb=> SELECT * FROM u1.t2;  
ERROR: permission denied for relation t2
```

Paso 6 En la sesión iniciada por el usuario **dbadmin**, conceda al usuario **u2** el permiso para consultar todas las tablas del esquema **u1**.

```
GRANT SELECT ON ALL TABLES IN SCHEMA u1 TO u2;
```

Paso 7 En la sesión iniciada por el usuario **u2**, verifique que **u2** pueda consultar todas las tablas.

```
SELECT * FROM u1.t1;  
SELECT * FROM u1.t2;
```



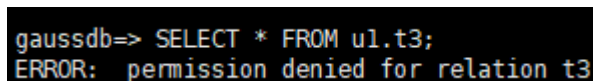
```
gaussdb=> SELECT * FROM u1.t1;SELECT * FROM u1.t2;  
c1 | c2  
----+----  
(0 rows)  
  
c1 | c2  
----+----  
(0 rows)
```

Paso 8 En la sesión iniciada por el usuario **dbadmin**, cree la tabla **u1.t3**.

```
CREATE TABLE u1.t3 (c1 int, c2 int);
```

Paso 9 En la sesión iniciada por el usuario **u2**, verifique que el usuario **u2** no tenga el permiso de consulta para **u1.t3**. Indica que el usuario **u2** tiene el permiso para acceder a todas las tablas existentes en el esquema **u1**, pero no a las tablas que se crearán en el futuro.

```
SELECT * FROM u1.t3;
```



```
gaussdb=> SELECT * FROM u1.t3;  
ERROR: permission denied for relation t3
```

Paso 10 En la sesión iniciada por el usuario **dbadmin**, conceda al usuario **u2** el permiso para consultar las tablas que se crearán en el esquema **u1**. Cree tabla **u1.t4**.

```
ALTER DEFAULT PRIVILEGES FOR ROLE u1 IN SCHEMA u1 GRANT SELECT ON TABLES TO u2;  
CREATE TABLE u1.t4 (c1 int, c2 int);
```

- Paso 11** En la sesión iniciada por el usuario **u2**, verifique que el usuario **u2** pueda acceder a la tabla **u1.t4**, pero no tenga el permiso para acceder a **u1.t3**. Para permitir que el usuario acceda a la tabla **u1.t3**, puede conceder permisos realizando [Paso 4](#).

```
SELECT * FROM u1.t4;
```

```
gaussdb=> SELECT * FROM u1.t4;
c1 | c2
----+----
(0 rows)
```

----Fin

6.5 ¿Cómo creo un usuario de solo lectura de la base de datos?

Escenario

En el desarrollo de servicios, los administradores de bases de datos usan esquemas para clasificar datos. Por ejemplo, en el sector financiero, los datos de pasivos pertenecen al esquema **s1** y los datos de activos pertenecen al esquema **s2**.

Ahora tiene que crear un usuario de solo lectura **user1** en la base de datos. El usuario puede acceder a todas las tablas (incluidas las nuevas tablas que se crearán en el futuro) del esquema **s1** para su lectura diaria, pero no puede insertar, modificar o eliminar datos.

Principios

DWS proporciona gestión de usuarios basada en roles. Es necesario crear un rol de solo lectura **role1** y conceder el rol a **user1**. Para obtener más información, consulte [Control de acceso basado en roles](#).

Procedimiento

- Paso 1** Conéctese a la base de datos de DWS como usuario **dbadmin**.

- Paso 2** Ejecute la siguiente instrucción de SQL para crear el rol **role1**:

```
CREATE ROLE role1 PASSWORD disable;
```

- Paso 3** Ejecute la siguiente instrucción de SQL para conceder permisos a **role1**:

```
The GRANT usage ON SCHEMA s1 TO role1; -- grants the access permission to schema s1.
GRANT select ON ALL TABLES IN SCHEMA s1 TO role1; -- grants the query permission on all tables in schema s1.
ALTER DEFAULT PRIVILEGES FOR USER tom IN SCHEMA s1 GRANT select ON TABLES TO role1; -- grants schema s1 the permission to create tables. tom is the owner of schema s1.
```

- Paso 4** Ejecute la siguiente instrucción de SQL para otorgar el rol **role1** al usuario **user1** real:

```
GRANT role1 TO user1;
```

- Paso 5** Lea todos los datos de la tabla en el esquema **s1** como el usuario de solo lectura **user1**.

----Fin

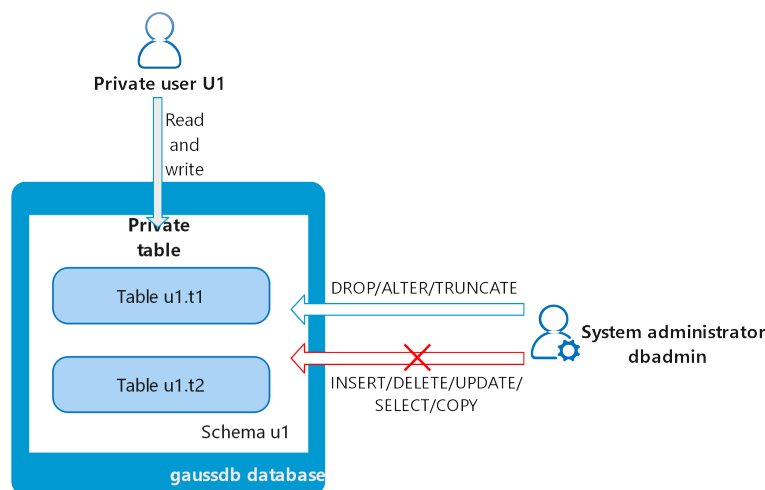
6.6 ¿Cómo puedo crear usuarios y tablas privadas de bases de datos?

Escenario

El administrador del sistema **dbadmin** tiene el permiso para acceder a las tablas creadas por usuarios comunes de forma predeterminada. Cuando **Separación de Permisos** está habilitada, el administrador **dbadmin** no tiene permiso para acceder a tablas de usuarios comunes o realizar operaciones de control (DROP, ALTER, y TRUNCATE).

Si es necesario crear un usuario privado y una tabla privada (tabla creada por el usuario privado) y solo el usuario privado puede acceder a la tabla privada y el administrador del sistema **dbadmin** y otros usuarios comunes no tienen el permiso para acceder a la tabla (INSERT, DELETE, UPDATE, SELECT y COPY). Sin embargo, el administrador del sistema **dbadmin** a veces necesita realizar las operaciones DROP, ALTER o TRUNCATE sin la autorización del usuario privado. En este caso, puede crear un usuario (usuario privado) con el atributo INDEPENDENT.

Figura 6-3 Usuarios privados



Principios

Esta función se implementa creando un usuario con el atributo INDEPENDENT.

INDEPENDENT | NOINDEPENDENT define los roles privados e independientes. Para un rol con el atributo **INDEPENDENT**, los derechos de los administradores para controlar y acceder a este rol están separados. Las normas específicas son las siguientes:

- Los administradores no tienen derechos para agregar, eliminar, consultar, modificar, copiar o autorizar los objetos de tabla correspondientes sin la autorización del rol INDEPENDENT.
- Los administradores no tienen derechos para modificar la relación de herencia del rol INDEPENDENT sin la autorización de este rol.

- Los administradores no tienen derechos para modificar el propietario de los objetos de tabla para el rol INDEPENDENT.
- Los administradores no tienen derechos para cambiar la contraseña de la base de datos del rol INDEPENDENT. El rol INDEPENDENT debe gestionar su propia contraseña, que no se puede restablecer si se pierde.
- El atributo **SYSADMIN** de un usuario no se puede cambiar por el atributo **INDEPENDENT**.

Procedimiento

Paso 1 Conéctese a la base de datos de DWS como usuario **dbadmin**.

Paso 2 Ejecute la siguiente instrucción SQL para crear el usuario privado **u1**:

```
CREATE USER u1 WITH INDEPENDENT IDENTIFIED BY 'password';
```

Paso 3 Cambie al usuario **u1**, cree la tabla **test** e inserte datos en la tabla.

```
CREATE TABLE test (id INT, name VARCHAR(20));  
INSERT INTO test VALUES (1, 'joe');  
INSERT INTO test VALUES (2, 'jim');
```

Paso 4 Cambie al usuario **dbadmin** y ejecute la siguiente instrucción de SQL para comprobar si el usuario **dbadmin** puede acceder a la tabla privada creada **test** por el usuario privado **u1**:

```
SELECT * FROM u1.test;
```

El resultado de la consulta indica que el usuario **dbadmin** no tiene el permiso de acceso. Esto significa que el usuario privado y la tabla privada se crean correctamente.

```
gaussdb=> SELECT * FROM u1.test;  
ERROR: SELECT permission denied to user "dbadmin" for relation "u1.test"
```

Paso 5 Ejecute la instrucción **DROP** como el usuario **dbadmin** para eliminar la tabla **test**.

```
DROP TABLE u1.test;
```

```
gaussdb=> drop table u1.test;  
DROP TABLE
```

----Fin

6.7 ¿Cómo revoco el permiso de CONNECT ON DATABASE de un usuario?

Escenario

En un servicio, el permiso del usuario **u1** para conectarse a una base de datos debe ser revocado. Una vez que el comando **REVOKE CONNECT ON DATABASE gaussdb FROM u1**; se ejecuta correctamente, el usuario **u1** todavía puede conectarse a la base de datos. Esto significa que la revocación no tiene efecto.

Análisis de las causas

Si ejecuta el comando **REVOKE CONNECT ON DATABASE gaussdb from u1** para revocar los permisos de **u1** de usuario, la revocación no surte efecto porque el permiso

CONNECT de la base de datos se concede a **PUBLIC**. Por lo tanto, debe especificar **PUBLIC**.

- GaussDB(DWS) proporciona un **PUBLIC** grupo definido implícitamente que contiene todos los roles. De forma predeterminada, todos los usuarios y roles nuevos tienen los permisos de **PUBLIC**. Para revocar los permisos de **PUBLIC** de un usuario o rol, o volver a concederles estos permisos, agregue la palabra clave **PUBLIC** en la instrucción **REVOKE** o **GRANT**.
- GaussDB(DWS) otorga los permisos para objetos de ciertos tipos al **PUBLIC**. De forma predeterminada, **PUBLIC** no concede permisos en tablas, columnas, secuencias, orígenes de datos externos, servidores externos, esquemas y tablas, pero **PUBLIC** concede los siguientes permisos:
 - **CONNECT** el permiso de una base de datos
 - **CREATE TEMP TABLE** el permiso de una base de datos
 - **EXECUTE** el permiso de una función
 - **USAGE** el permiso para idiomas y tipos de datos (incluidos dominios)
- Un propietario de objeto puede revocar los permisos predeterminados concedidos a **PUBLIC** y conceder permisos a otros usuarios según sea necesario.

Ejemplo de operaciones

Ejecute el siguiente comando para revocar el permiso del usuario **u1** para acceder al **gaussdb** de la base de datos:

Paso 1 Conéctese a la base de datos de GaussDB(DWS) **gaussdb**.

```
gsql -d gaussdb -p 8000 -h 192.168.x.xx -U dbadmin -W password -r  
gaussdb=>
```

Paso 2 Cree el usuario **u1**.

```
gaussdb=> CREATE USER u1 IDENTIFIED BY 'xxxxxxx';
```

Paso 3 Verifique que el usuario **u1** pueda acceder a GaussDB.

```
gsql -d gaussdb -p 8000 -h 192.168.x.xx -U u1 -W password -r  
gaussdb=>
```

Paso 4 Conéctese a la base de datos **gaussdb** como administrador **dbadmin** y ejecute el comando **REVOKE** para revocar el permiso de conexión en la base de datos del usuario **public**.

```
gsql -d gaussdb -h 192.168.x.xx -U dbadmin -p 8000 -r  
gaussdb=> REVOKE CONNECT ON DATABASE gaussdb FROM public;  
REVOKE
```

Paso 5 Verifique el resultado. Utilice **u1** para conectarse a la base de datos. Si se muestra la siguiente información, el permiso **connect on database** del usuario **u1** se ha revocado correctamente:

```
gsql -d gaussdb -p 8000 -h 192.168.x.xx -U u1 -W password -r  
gsql: FATAL: permission denied for database "gaussdb"  
DETAIL: User does not have CONNECT privilege.
```

----Fin

6.8 ¿Cómo puedo ver los permisos de tabla de un usuario?

Escenario 1: Ejecute el comando **information_schema.table_privileges** a **view the table permissions of a user**. Por ejemplo:

```
SELECT * FROM information_schema.table_privileges WHERE GRANTEE='user_name';
```

```
gaussdb-> SELECT * FROM information_schema.table_privileges WHERE GRANTEE='u2';
grantor | grantee | table_catalog | table_schema | table_name | privilege_type | is_grantable | with_hierarchy
-----|-----|-----|-----|-----|-----|-----|-----
u2      | u2      | gaussdb      | u2           | t2         | INSERT        | YES          | NO
u2      | u2      | gaussdb      | u2           | t2         | SELECT        | YES          | YES
u2      | u2      | gaussdb      | u2           | t2         | UPDATE        | YES          | NO
u2      | u2      | gaussdb      | u2           | t2         | DELETE        | YES          | NO
u2      | u2      | gaussdb      | u2           | t2         | TRUNCATE      | YES          | NO
u2      | u2      | gaussdb      | u2           | t2         | REFERENCES    | YES          | NO
u2      | u2      | gaussdb      | u2           | t2         | TRIGGER       | YES          | NO
u2      | u2      | gaussdb      | u2           | t2         | ANALYZE       | YES          | NO
u2      | u2      | gaussdb      | u2           | t2         | VACUUM        | YES          | NO
u2      | u2      | gaussdb      | u2           | t2         | ALTER         | YES          | NO
u2      | u2      | gaussdb      | u2           | t2         | DROP          | YES          | NO
u1      | u2      | gaussdb      | u1           | t1         | SELECT        | NO           | YES
(12 rows)
```

Tabla 6-2 Columnas de table_privileges

Columna	Tipo de datos	Descripción
grantor	sql_identifier	Concedidor de permisos
grantee	sql_identifier	Concesionario del permiso
table_catalog	sql_identifier	Base de datos donde se encuentra la tabla
table_schema	sql_identifier	Esquema donde se encuentra la tabla
table_name	sql_identifier	Nombre de la tabla
privilege_type	character_data	Tipo de permiso concedido. El valor puede ser SELECT , INSERT , UPDATE , DELETE , TRUNCATE , REFERENCES , ANALYZE , VACUUM , ALTER , DROP o TRIGGER .
is_grantable	yes_or_no	Indica si el permiso se puede conceder a otros usuarios. YES indica que el permiso se puede conceder a otros usuarios, y NO indica que el permiso no se puede conceder a otros usuarios.
with_hierarchy	yes_or_no	Indica si se permiten operaciones específicas heredadas en el nivel de tabla. Si la operación específica es SELECT , se muestra YES . De lo contrario, se muestra NO .

En la figura anterior, el usuario **u2** tiene todos los permisos de la tabla **t2** en el esquema **u2** y el permiso **SELECT** de la tabla **t1** en el esquema **u1**.

information_schema.table_privileges solo puede consultar los permisos concedidos directamente al usuario, la función **has_table_privilege()** puede consultar tanto los permisos concedidos directamente como los permisos indirectos (obtenidos por el rol de GRANT al usuario). Por ejemplo:

```
CREATE TABLE t1 (c1 int);
CREATE USER u1 password '*****';
CREATE USER u2 password '*****';
GRANT dbadmin to u2; //Indirectly grant permissions through roles.
GRANT SELECT on t1 to u1; // Directly grant the permission.

SET ROLE u1 password '*****';
SELECT * FROM public.t1; // Directly grant the permission to access the table.
c1
----
```

```
(0 rows)

SET ROLE u2 password '*****';
SELECT * FROM public.t1; // Indirectly grant the permission to access the table.
c1
----
(0 rows)

RESET role; //Switch back to dbadmin.
SELECT * FROM information_schema.table_privileges WHERE table_name = 't1'; // Can
only view direct grants.
 grantor | grantee | table_catalog | table_schema | table_name |
 privilege_type | is_grantable | with_hierarchy
-----+-----+-----+-----+-----+-----+-----+-----
+-----+-----+-----+-----+-----+-----+-----+-----
 dbadmin | u1 | gaussdb | public | t1 |
 SELECT | NO | YES
(1 rows)

SELECT has_table_privilege('u2', 'public.t1', 'select'); // Can view both direct
and indirect grants.
 has_table_privilege
-----
 t
(1 row)
```

Escenario 2: Para check whether a user has permissions on a table, realice los siguientes pasos:

6.9 ¿Quién es la usuaria Ruby?

Cuando ejecuta la instrucción **SELECT * FROM pg_user** para ver usuarios en el sistema actual, puede ver a la usuaria **Ruby** muchas veces, que tiene muchos permisos.

Usuaria **Ruby** es una cuenta oficial de O&M. Después de crear una base de datos de GaussDB(DWS), se genera la usuaria **Ruby** de forma predeterminada. No existe ningún riesgo de seguridad con respecto a la usuaria **Ruby**.

```
gaussdb> SELECT * FROM pg_user;
username | usesysid | usecreatedb | usesuper | usecatupd | userepl | passwd | valbegin | valuntil | respool | parent | spacelimit | useconfig | nodegroup | temppacelimit | spillspa
celimit
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----
 dbadmin | 18384 | f | f | f | f | ***** | | | default_pool | 0 | | | | |
 Ruby | 24584 | f | f | f | f | ***** | | | default_pool | 0 | | | | |
 user_1 | 24584 | f | f | f | f | ***** | | | default_pool | 0 | | | | |
 u1 | 24593 | f | f | f | f | ***** | | | default_pool | 0 | | | | |
 u2 | 24597 | f | f | f | f | ***** | | | default_pool | 0 | | | | |
(5 rows)
```

7 Uso de la base de datos

7.1 ¿Cómo cambio las columnas de distribución?

En una base de datos de almacén de datos, debe elegir cuidadosamente las columnas de distribución para las tablas grandes, ya que pueden afectar al rendimiento de la base de datos y las consultas. Si se utiliza una clave de distribución incorrecta, puede producirse un sesgo de datos después de importar los datos. Como resultado, el uso de algunos discos será mucho más alto que el de otros discos, e incluso el clúster puede llegar a ser de solo lectura. Si se utiliza la política de distribución hash y se produce un sesgo de datos, el rendimiento de E/S de algunos DN será deficiente, lo que afectará al rendimiento general de la consulta. La selección y el ajuste adecuados de las columnas de distribución son fundamentales para el rendimiento de las consultas de tabla.

Si se utiliza la política de distribución hash, debe comprobar las tablas para asegurarse de que sus datos estén distribuidos uniformemente en cada DN. Generalmente, más del 5% de diferencia entre la cantidad de datos en diferentes DN se considera como sesgo de datos. Si la diferencia es superior al 10%, tiene que elegir otra columna de distribución.

Para las tablas que no están distribuidas uniformemente, ajuste sus columnas de distribución para reducir el sesgo de datos y evitar problemas de rendimiento de la base de datos.

Elección de una columna de distribución apropiada

La columna de distribución de una tabla hash debe cumplir los siguientes requisitos, que se clasifican por prioridad en orden descendente:

- Los valores de la clave de distribución deben ser discretos para que los datos puedan distribuirse uniformemente en cada DN. Puede seleccionar la clave principal de la tabla como clave de distribución. Por ejemplo, para una tabla de información de persona, elija la columna número de tarjeta de identificación como clave de distribución.
- No seleccione la columna donde existe un filtro constante.
- Seleccione la condición de unión como columna de distribución, de modo que las tareas de unión se puedan empujar hacia abajo a los DN para ejecutar, reduciendo la cantidad de datos transferidos entre los DN.
- Se pueden seleccionar varias columnas de distribución para distribuir los datos de manera uniforme.

Procedimiento

Ejecute la sentencia **select version()**; para consultar la versión actual de la base de datos. El rendimiento requerido varía según la versión.

```
test_lhy=> select version();
                version
-----
PostgreSQL 9.2.4 (GaussDB 8.1.1 build 7ab61a49) compiled at 2021-06-26 12:05:53 commit 2518 last mr 3356 release
(1 row)
```

- **Para las versiones 8.0.x y anteriores, reconstruya una tabla.**

Paso 1 Utilice Data Studio o gsql en Linux para acceder a la base de datos.

Paso 2 Cree una tabla.

NOTA

En las instrucciones siguientes, **table1** es el nombre original de la tabla y **table1_new** es el nombre de la nueva tabla. **column1** y **column2** son los nombres de columna de distribución.

```
CREATE TABLE IF NOT EXISTS table1_new
 ( LIKE table1 INCLUDING ALL EXCLUDING DISTRIBUTION)
DISTRIBUTE BY
HASH (column1, column2);
```

Paso 3 Migre datos a la nueva tabla.

```
START TRANSACTION;
LOCK TABLE table1 IN ACCESS EXCLUSIVE MODE;
INSERT INTO table1_new SELECT * FROM table1;
COMMIT;
```

Paso 4 Compruebe que se han migrado los datos de la tabla. Elimine la tabla original.

```
SELECT COUNT(*) FROM table1_new;
DROP TABLE table1;
```

Paso 5 Reemplace la tabla original.

```
ALTER TABLE table1_new RENAME TO table1;
```

----Fin

7.2 ¿Cómo puedo ver y establecer la codificación de caracteres de la base de datos?

Consulta de la codificación de caracteres de la base de datos

Utilice el parámetro **server_encoding** para comprobar la codificación del juego de caracteres de la base de datos actual. Por ejemplo, la codificación de caracteres de la base de datos **music** es UTF8.

```
music=> show server_encoding;
server_encoding
-----
UTF8
(1 row)
```

Configuración de la codificación de caracteres de base de datos

NOTA

GaussDB(DWS) no admite la modificación del formato de codificación de caracteres de una base de datos creada.

Si necesita especificar el formato de codificación de caracteres de una base de datos, use **template0** y la sintaxis **CREATE DATABASE** para crear una base de datos. Para que su base de datos sea compatible con la mayoría de los caracteres, se recomienda utilizar la codificación de UTF8 al crear una base de datos.

Sintaxis de CREATE DATABASE

```
CREATE DATABASE database_name
[ [ WITH ] { [ OWNER [=] user_name ] |
[ TEMPLATE [=] template ] |
[ ENCODING [=] encoding ] |
[ LC_COLLATE [=] lc_collate ] |
[ LC_CTYPE [=] lc_ctype ] |
[ DBCOMPATIBILITY [=] compatibility_type ] |
[ CONNECTION LIMIT [=] connlimit ]} [...] ];
```

- **TEMPLATE [=] template**

Indica el nombre de la plantilla, es decir, el nombre de la plantilla que se utilizará para crear la base de datos. GaussDB(DWS) crea una base de datos haciendo frente a una plantilla de base de datos. GaussDB(DWS) tiene dos bases de datos de plantillas iniciales **template0** y **template1** y una base de datos de usuario por defecto **gaussdb**.

Rango de valores: un nombre de base de datos existente. Si no se especifica, el sistema copia **template1** de forma predeterminada. Su valor no puede ser **gaussdb**.

AVISO

Actualmente, las plantillas de base de datos no pueden contener secuencias. Si existen secuencias en la biblioteca de plantillas, se producirá un error en la creación de la base de datos.

- **ENCODING [=] encoding**

Codificación de caracteres utilizada por la base de datos. El valor puede ser una cadena de caracteres (por ejemplo, **SQL_ASCII**) o un número entero.

Si no se especifica este parámetro, se utiliza la codificación de la base de datos de plantillas de forma predeterminada. La codificación de las bases de datos de plantillas **template0** y **template1** depende del sistema operativo de forma predeterminada. La codificación de caracteres de **template1** no se puede cambiar. Para cambiar la codificación, utilice **template0** para crear una base de datos.

Rango de valores: **GBK**, **UTF8** y **Latin1**

AVISO

La codificación del juego de caracteres de la nueva base de datos debe ser compatible con la configuración local (**LC_COLLATE** y **LC_CTYPE**).

Ejemplos

Cree la base de datos **music** con UTF8 (el tipo de codificación local es también UTF8).

```
CREATE DATABASE music ENCODING 'UTF8' template = template0;
```

7.3 ¿Qué hago si el tipo de fecha se convierte automáticamente en el tipo de marca de tiempo durante la creación de la tabla?

Al crear una base de datos, puede establecer el parámetro **DBCOMPATIBILITY** en el tipo de base de datos compatible. El valor de **DBCOMPATIBILITY** puede ser **ORA**, **TD** y **MySQL**, indicando las bases de datos Oracle, Teradata y MySQL, respectivamente. Si este parámetro no se especifica durante la creación de la base de datos, se utiliza el valor predeterminado **ORA**. En el modo de compatibilidad **ORA**, el tipo de fecha se convierte automáticamente en marca de tiempo (0).

Para evitar esta conversión, establezca la base de datos en el modo de compatibilidad **MySQL**, que es el único modo que admite el tipo de fecha. No se puede cambiar el modo de compatibilidad de una base de datos existente. Solo se puede especificar durante la creación de la base de datos. GaussDB(DWS) soporta el modo de compatibilidad de **MySQL** en la versión 8.1.1 y posterior de clúster. Para configurar este modo, ejecute los siguientes comandos:

```
gaussdb=> CREATE DATABASE mydatabase DBCOMPATIBILITY='mysql';
CREATE DATABASE
gaussdb=> \c mydatabase
Non-SSL connection (SSL connection is recommended when requiring high-security)
You are now connected to database "mydatabase" as user "dbadmin".
mydatabase=> create table t1(c1 int, c2 date);
NOTICE: The 'DISTRIBUTE BY' clause is not specified. Using round-robin as the
distribution mode by default.
HINT: Please use 'DISTRIBUTE BY' clause to specify suitable data distribution
column.
CREATE TABLE
```

Si el problema no se puede resolver cambiando la compatibilidad, puede intentar cambiar el tipo de columna. Por ejemplo, inserte datos del tipo de fecha como cadena en una tabla. Por ejemplo:

```
gaussdb=> CREATE TABLE mytable (a date,b int);
CREATE TABLE
gaussdb=> INSERT INTO mytable VALUES(date '12-08-2023',01);
INSERT 0 1
gaussdb=> SELECT * FROM mytable;
      a      | b
-----+-----
2023-12-08 00:00:00 | 1
(1 row)
gaussdb=> ALTER TABLE mytable MODIFY a VARCHAR(20);
ALTER TABLE
gaussdb=> INSERT INTO mytable VALUES('2023-12-10',02);
INSERT 0 1
gaussdb=> SELECT * FROM mytable;
      a      | b
-----+-----
2023-12-08 00:00:00 | 1
2023-12-10          | 2
(2 rows)
```


7.4 ¿Necesito ejecutar **VACUUM FULL** y **ANALYZE** en tablas comunes periódicamente?

Sí.

Para las tablas que se agregan, eliminan o modifican con frecuencia, debe realizar periódicamente **VACUUM FULL** y **ANALYZE** para recuperar el espacio en disco ocupado por los datos actualizados o eliminados, evitando el deterioro del rendimiento causado por la acumulación de datos y estadísticas inexactas.

- Por lo general, se recomienda realizar **ANALYZE** después de un gran número de operaciones de **adición o modificación** se realizan en una tabla.
- Después de eliminar una tabla, se recomienda ejecutar **VACUUM** en lugar de **VACUUM FULL**. Sin embargo, puede ejecutar **VACUUM FULL** en algunos casos particulares, como cuando desea reducir físicamente una tabla para reducir el espacio ocupado en disco después de eliminar la mayoría de las filas de la tabla. Para obtener más información sobre las diferencias entre **VACUUM** y **VACUUM FULL**, consulte [VACUUM y VACUUM FULL](#).

Sintaxis

Realice **ANALYZE** en una tabla.

```
ANALYZE table_name;
```

Realice **ANALYZE** en todas las tablas (tablas no extranjeras) de la base de datos.

```
ANALYZE;
```

Realice **VACUUM** en una tabla.

```
VACUUM table_name;
```

Realice **VACUUM FULL** en una tabla.

```
VACUUM FULL table_name;
```

Para más detalles, véase [VACUUM](#) y [ANALYZE | ANALYSE](#).

NOTA

- Si el uso de espacio físico no disminuye después de ejecutar el comando **VACUUM FULL** compruebe si había otras transacciones activas. (iniciado antes de eliminar las transacciones de datos y no finalizado antes de ejecutar **VACUUM FULL**). En caso afirmativo, vuelva a ejecutar este comando cuando las transacciones hayan finalizado.
- En la versión 8.1.3 o posterior, se puede invocar **VACUUM/VACUUM FULL** en el plano de gestión. Para obtener más información, consulte [O&M inteligente](#).

VACUUM y VACUUM FULL

En GaussDB(DWS), la operación **VACUUM** es como una aspiradora utilizada para absorber el polvo. Aquí, "polvo" significa los datos antiguos. Si los datos no se borran de manera oportuna, el espacio de la base de datos se hinchará, causando un deterioro del rendimiento o incluso un fallo del sistema.

Finalidades del **VACUUM**:

- Resolver hinchazón de espacio: Borrar tuplas obsoletas e índices correspondientes, que incluyen la tupla (e índice) de una transacción de **DELETE** comprometida, la versión antigua (e índice) de una transacción de **UPDATE** insertada (e índice) de una transacción de **INSERT** removida, la nueva versión (e índice) de una transacción de **UPDATE** y la tupla (e índice) de una transacción de **COPY**.
- **VACUUM FREEZE**: Evita la avería del sistema causada por la envoltura de ID de transacción. Convierte los ID de transacción más pequeños que OldestXmin para congelar xids, actualizar relfrozenxids en una tabla y actualizar relfrozenxids y truncar obstrucciones en una base de datos.
- Actualizar estadísticas: **VACUUM ANALYZE** actualiza las estadísticas, lo que permite al optimizador seleccionar una mejor manera de ejecutar sentencias SQL.

La instrucción de **VACUUM** incluye **VACUUM** y **VACUUM FULL**. Actualmente, el **VACUUM** solo puede funcionar en tablas de almacén de filas. **VACUUM FULL** se puede utilizar para liberar espacio de las tablas de almacén de columnas. Para más detalles, consulte la siguiente tabla.

Tabla 7-1 **VACUUM** y **VACUUM FULL**

Elemento	VACUUM	VACUUM FULL
Despejando o el espacio	Si el registro eliminado está al final de una tabla, el espacio ocupado por el registro eliminado se libera físicamente y se devuelve al sistema operativo. Si los datos no están al final de una tabla, el espacio ocupado por las tuplas muertas en la tabla o índice se establece para que esté disponible para su reutilización.	A pesar de la posición de los datos eliminados, el espacio ocupado por los datos se libera físicamente y se devuelve al sistema operativo. Cuando se insertan datos, se asigna una nueva página de disco.
Tipo de bloqueo	Bloqueo compartido. La operación VACUUM se puede realizar en paralelo con otras operaciones.	Bloqueo exclusivo. Todas las operaciones basadas en la tabla se suspenden durante la ejecución.
Espacio físico	No lanzado	Lanzado
ID de transacción	No reclamado	Reciclado
Gastos generales de ejecución	La sobrecarga es baja y la operación se puede ejecutar periódicamente.	La sobrecarga es alta. Se recomienda realizarlo cuando el espacio de la página de disco ocupado por la base de datos está cerca del umbral y las operaciones de datos son pocas.
Efecto	Mejora la eficiencia de las operaciones sobre la tabla.	Mejora en gran medida la eficiencia de las operaciones sobre la tabla.

7.5 ¿Necesito ajustar una clave de distribución después de establecer una clave principal?

No, solo necesita ajustar la clave principal. De forma predeterminada, la primera columna de la clave principal se selecciona como clave de distribución. Si ambos están definidos, la clave principal debe contener la clave de distribución.

7.6 ¿Es GaussDB(DWS) compatible con los procedimientos almacenados de PostgreSQL?

Sí.

GaussDB(DWS) es compatible con los procedimientos almacenados de PostgreSQL. Para obtener más información, consulte [Procedimientos almacenados](#).

7.7 ¿Qué son las tablas particionadas, las particiones y las claves de partición?

Tabla particionada: La partición se refiere a dividir lo que lógicamente es una tabla grande en piezas físicas más pequeñas basadas en esquemas específicos. La tabla basada en la lógica se llama una tabla particionada, y una pieza física se llama una partición. Los datos se almacenan en estas piezas físicas más pequeñas, a saber, particiones, en lugar de la tabla particionada lógica más grande.

Partición: En el sistema distribuido GaussDB(DWS), la partición de datos es para dividir horizontalmente los datos de la tabla dentro de un nodo basado en una política especificada. La tabla se divide en particiones que no se superponen dentro de un rango específico.

Clave de partición: Una clave de partición es un conjunto ordenado de una o más columnas de tabla. Los valores de las claves de partición de tabla se utilizan para determinar la partición de datos a la que pertenece una fila.

7.8 ¿Cómo puedo exportar la estructura de la tabla?

Se recomienda utilizar el cliente gráfico de Data Studio para exportar datos de tabla. Puede exportar datos desde:

- Una tabla específica
- Todas las tablas de un esquema
- Todas las tablas de una base de datos

Para obtener más información, consulte [Exportación de datos de tabla](#).

Alternativamente, utilice `gs_dump` y `gs_dumpall` para exportar datos. Puedes hacer lo siguiente:

- Exportar una única base de datos:

- Exportación a nivel de base de datos
- Exportación a nivel de modo
- Exportación a nivel de tabla
- Exportar todas las bases de datos:
 - Exportación a nivel de base de datos
 - Exportación de objetos globales de cada base de datos

Para obtener más información, consulte [gs_dump](#) y [gs_dumpall](#).

7.9 ¿Cómo puedo eliminar los datos de la tabla de manera eficiente?

Sí. **TRUNCATE** es más eficiente que **DELETE** para eliminar datos masivos.

Para obtener más información, consulte [TRUNCATE](#).

Function

TRUNCATE quickly removes all rows from a database table.

It has the same effect as the unconditional **DELETE**, but **TRUNCATE** is faster, especially for large tables, because it does not scan tables.

Functions

- **TRUNCATE TABLE** works like a **DELETE** statement with no **WHERE** clause, that is, emptying a table.
- **TRUNCATE TABLE** uses less system and transaction log resources.
 - **DELETE** deletes a row each time, and records each deletion in the transaction log.
 - **TRUNCATE TABLE** deletes all rows in a table by releasing the data page, and only records each releasing of the data page in the transaction log.
- **TRUNCATE**, **DELETE**, and **DROP** are different in that:
 - **TRUNCATE TABLE** deletes content, releases space, but does not delete definitions.
 - **DELETE TABLE** deletes content, but does not delete definitions or release space.
 - **DROP TABLE** deletes content and definitions, and releases space.

Examples

```
--Create a table.CREATE TABLE tpcds.reason_t1 AS TABLE tpcds.reason;
--Truncate the table.TRUNCATE TABLE tpcds.reason_t1;
--Drop the table.DROP TABLE tpcds.reason_t1;
--Create a partitioned table.
CREATE TABLE tpcds.reason_p
(
  r_reason_sk integer,
  r_reason_id character(16),
  r_reason_desc character(100)
)PARTITION BY RANGE (r_reason_sk)
```


- Escenario 1

Si se incluye la clave principal o la restricción única durante la creación de la tabla, se selecciona la distribución hash. La columna de distribución es la columna correspondiente a la clave principal o restricción única.

```
CREATE TABLE warehouse1
(
  W_WAREHOUSE_SK          INTEGER          PRIMARY KEY,
  W_WAREHOUSE_ID         CHAR(16)          NOT NULL,
  W_WAREHOUSE_NAME       VARCHAR(20)
);
NOTICE: CREATE TABLE / PRIMARY KEY will create implicit index
"warehouse1_pkey" for table "warehouse1"
CREATE TABLE

SELECT getdistributekey('warehouse1');
getdistributekey
-----
w_warehouse_sk
(1 row)
```

- Escenario 2

Si la clave principal o la restricción única no se incluyen durante la creación de la tabla pero hay columnas cuyos tipos de datos se pueden utilizar como columnas de distribución, se selecciona la distribución hash. La columna de distribución es la primera columna cuyo tipo de datos se puede utilizar como columna de distribución.

```
CREATE TABLE warehouse2
(
  W_WAREHOUSE_SK          INTEGER          ,
  W_WAREHOUSE_ID         CHAR(16)          NOT NULL,
  W_WAREHOUSE_NAME       VARCHAR(20)
);
NOTICE: The 'DISTRIBUTE BY' clause is not specified. Using 'w_warehouse_sk'
as the distribution column by default.
HINT: Please use 'DISTRIBUTE BY' clause to specify suitable data
distribution column.
CREATE TABLE

SELECT getdistributekey('warehouse2');
getdistributekey
-----
w_warehouse_sk
(1 row)
```

- Escenario 3

Si la clave principal o la restricción única no se incluyen durante la creación de la tabla y no existe ninguna columna cuyo tipo de datos se pueda utilizar como columna de distribución, se selecciona la distribución round-robin.

```
CREATE TABLE warehouse3
(
  W_WAREHOUSE_ID         CHAR(16)          NOT NULL,
  W_WAREHOUSE_NAME       VARCHAR(20)
);
NOTICE: The 'DISTRIBUTE BY' clause is not specified. Using 'w_warehouse_id'
as the distribution column by default.
HINT: Please use 'DISTRIBUTE BY' clause to specify suitable data
distribution column.
CREATE TABLE

SELECT getdistributekey('warehouse3');
getdistributekey
-----
w_warehouse_id
(1 row)
```

7.12 ¿Cómo reemplazo el resultado nulo por 0?

Cuando se ejecuta OUTER JOIN (LEFT JOIN, RIGHT JOIN y FULL JOIN), el error de coincidencia en la unión externa genera un gran número de valores NULL. Puede reemplazar estos valores nulos por 0.

Puede usar la función **COALESCE** para hacerlo. Esta función devuelve el primer valor de parámetro no nulo en la lista de parámetros. Por ejemplo:

```
SELECT coalesce(NULL, 'hello');
 coalesce
-----
hello
(1 row)
```

Usa la combinación izquierda para unirte a las tablas **course1** y **course2**.

```
SELECT * FROM course1;
 stu_id | stu_name | cour_name
-----+-----+-----
 20110103 | ALLEN   | Math
 20110102 | JACK    | Programming Design
 20110101 | MAX     | Science
(3 rows)

SELECT * FROM course2;
 cour_id | cour_name | teacher_name
-----+-----+-----
 1002 | Programming Design | Mark
 1001 | Science | Anne
(2 rows)

SELECT course1.stu_name, course2.cour_id, course2.cour_name, course2.teacher_name
FROM course1 LEFT JOIN course2 ON course1.cour_name = course2.cour_name ORDER BY
1;
 stu_name | cour_id | cour_name | teacher_name
-----+-----+-----+-----
ALLEN    |         |           |
JACK     | 1002   | Programming Design | Mark
MAX      | 1001   | Science | Anne
(3 rows)
```

Utilice la función **COALESCE** para reemplazar valores nulos en el resultado de la consulta con 0 u otros valores distintos de cero:

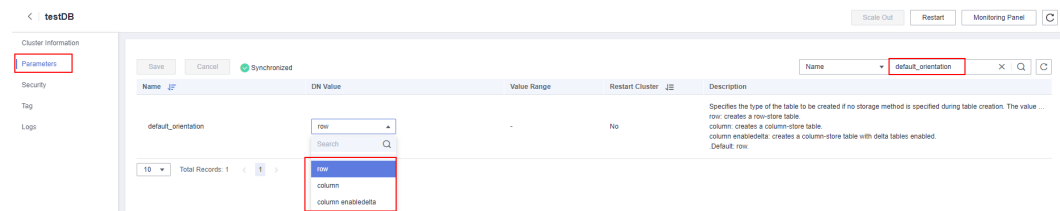
```
SELECT course1.stu_name,
 coalesce(course2.cour_id, 0) AS cour_id,
 coalesce(course2.cour_name, 'NA') AS cour_name,
 coalesce(course2.teacher_name, 'NA') AS teacher_name
FROM course1
LEFT JOIN course2 ON course1.cour_name = course2.cour_name
ORDER BY 1;
 stu_name | cour_id | cour_name | teacher_name
-----+-----+-----+-----
ALLEN    | 0 | NA | NA
JACK     | 1002 | Programming Design | Mark
MAX      | 1001 | Science | Anne
(3 rows)
```

7.13 ¿Cómo puedo comprobar si una tabla está almacenada en filas o en columnas?

El modo de almacenamiento de una tabla se controla mediante el parámetro **ORIENTATION** en la instrucción de creación de tabla. **row** indica el almacenamiento de filas y **column** indica el almacenamiento de columnas.

En 8.1.2 y versiones anteriores, si no se especifica **ORIENTATION**, el almacenamiento de filas se utiliza de forma predeterminada.

En versiones posteriores a la 8.1.3, el parámetro **default_orientation** se puede utilizar para controlar el modo de almacenamiento. Si el parámetro **ORIENTATION** no se especifica durante la creación de la tabla, el modo de almacenamiento de tabla se basa en el valor de **default_orientation**. **row** indica una tabla de almacén de filas, **column** indica una tabla de almacén de columnas y **column enabledelta** indica que se crea una tabla de almacén de columnas con delta activado. El parámetro GUC se puede configurar en la consola GaussDB(DWS), como se muestra en la siguiente figura.



Puede utilizar la función de definición de tabla **PG_GET_TABLEDEF** para comprobar si la tabla creada es almacén de filas o almacén de columnas.

Por ejemplo, **orientation=column** indica una tabla de almacenamiento de columnas.

Actualmente, no puede ejecutar la instrucción **ALTER TABLE** para modificar el parámetro **ORIENTATION**.

```
SELECT * FROM PG_GET_TABLEDEF('customer_t1');

pg_get_tabledef
-----
-
SET search_path = tpchobs;
+
CREATE TABLE customer_t1
(
    c_customer_sk integer,
+
    c_customer_id character(5),
+
    c_first_name character(6),
+
    c_last_name character(8)
+
)
+
WITH (orientation=column, compression=middle, colversion=2.0, enable_delta=false)
+
DISTRIBUTE BY HASH(c_last_name)
+
TO GROUP group_version1;
(1 row)
```


7.14 ¿Cómo puedo consultar la información sobre las tablas de almacén de columnas de GaussDB(DWS)?

Las siguientes instrucciones de SQL se utilizan para consultar información común acerca de las tablas de almacenamiento de columnas:

Cree una tabla **my_table** particionada de almacén de columnas.

```
CREATE TABLE my_table
(
  product_id INT,
  product_name VARCHAR2(40),
  product_quantity INT
)
WITH (ORIENTATION = COLUMN)
PARTITION BY range(product_quantity)
(
  partition my_table_p1 values less than(600),
  partition my_table_p2 values less than(800),
  partition my_table_p3 values less than(950),
  partition my_table_p4 values less than(1000));
INSERT INTO my_table VALUES(1011, 'tents', 720);
INSERT INTO my_table VALUES(1012, 'hammock', 890);
INSERT INTO my_table VALUES(1013, 'compass', 210);
INSERT INTO my_table VALUES(1014, 'telescope', 490);
INSERT INTO my_table VALUES(1015, 'flashlight', 990);
INSERT INTO my_table VALUES(1016, 'ropes', 890);
```

Ejecute el siguiente comando para ver la tabla particionada de almacén de columnas creada:

```
SELECT * FROM my_table;
product_id | product_name | product_quantity
-----+-----+-----
1013 | compass | 210
1014 | telescope | 490
1011 | tents | 720
1015 | flashlight | 990
1012 | hammock | 890
1016 | ropes | 890
(6 rows)
```

Consulta del contorno de una partición

```
SELECT relname, partstrategy, boundaries FROM pg_partition where parentid=(select
parentid from pg_partition where relname='my_table');
relname | partstrategy | boundaries
-----+-----+-----
my_table | r |
my_table_p1 | r | {600}
my_table_p2 | r | {800}
my_table_p3 | r | {950}
my_table_p4 | r | {1000}
(5 rows)
```

Consulta del número de columnas en una tabla de almacén de columnas

```
SELECT count(*) FROM ALL_TAB_COLUMNS where table_name='my_table';
count
-----
3
(1 row)
```

Consulta de distribución de datos en DN

```
SELECT table_skewness('my_table');
       table_skewness
-----
("dn_6007_6008      ",3,50.000%)
("dn_6009_6010      ",2,33.333%)
("dn_6003_6004      ",1,16.667%)
("dn_6001_6002      ",0,0.000%)
("dn_6005_6006      ",0,0.000%)
("dn_6011_6012      ",0,0.000%)
(6 rows)
```

Consulta de los nombres de las tablas de Cudesc y de Delta en la partición P1 en un DN

```
EXECUTE DIRECT ON (dn_6003_6004) 'select a.relname from pg_class a, pg_partition
b where (a.oid=b.reldeltarelid or a.oid=b.relcudescrelid) and
b.relname='my_table_p1''';
       relname
-----
pg_delta_part_60317
pg_cudesc_part_60317
(2 rows)
```

7.15 ¿Por qué a veces los índices de consulta de GaussDB(DWS) se vuelven inválidos?

La creación de índices para tablas puede mejorar el rendimiento de las consultas de base de datos. Sin embargo, a veces los índices no se pueden utilizar en un plan de consulta. Esta sección describe varias razones comunes y métodos de optimización.

Motivo 1: Los conjuntos de resultados devueltos son grandes.

A continuación, se utiliza Seq Scan y Index Scan en una tabla de almacén de filas como ejemplo:

- Seq Scan: busca registros de tabla en secuencia. Todos los registros se recuperan durante cada escaneo. Este es el método de escaneo de tabla más simple y básico, y su costo es alto.
- Análisis de índice: busque primero el índice, encuentre la ubicación de destino (puntero) en el índice y, a continuación, recupera datos en la página de destino.

La exploración de índice es más rápida que la exploración de secuencia en la mayoría de los casos. Sin embargo, si los conjuntos de resultados obtenidos representan una gran proporción (más del 70%) de todos los datos, Index Scan necesita analizar los índices antes de leer los datos de la tabla. Esto hace que el escaneo de la tabla sea más lento.

Razón 2: ANALYZE no se realiza de manera oportuna.

ANALYZE se utiliza para actualizar las estadísticas de la tabla. Si **ANALYZE** no se ejecuta en una tabla o una gran cantidad de datos se añade o se elimina de una tabla después de ejecutar **ANALYZE** las estadísticas pueden ser inexactas, lo que puede hacer que una consulta omita el índice.

Método de optimización: ejecute la instrucción **ANALYZE** en la tabla para actualizar las estadísticas.

Motivo 3: Las condiciones de filtrado contienen funciones o conversión de tipo de datos implícito

Si la conversión de cálculo, función o tipo de datos implícito está contenida en los criterios de filtro, los índices pueden fallar al seleccionarse.

Por ejemplo, cuando se crea una tabla, los índices se crean en las columnas **a**, **b** y **c**.

```
create table test(a int, b text, c date);
```

- Realice el cálculo en las columnas indexadas.

El siguiente resultado del comando indica que tanto **where a = 101** como **where a = 102 - 1** usan el índice en la columna a, pero **where a + 1 = 102** no usa el índice.

```
explain verbose select * from test where a = 101;
QUERY PLAN
-----
id | operation | E-rows | E-distinct |
E-memory | E-width | E-costs
-----+-----+-----
1 | -> Streaming (type: GATHER) | 1 | |
| | 44 | 16.27
2 | -> Index Scan using index_a on public.test | 1 | |
1MB | | 44 | 8.27

Predicate Information (identified by plan id)
-----
2 --Index Scan using index_a on public.test
Index Cond: (test.a = 101)

Targetlist Information (identified by plan id)
-----
1 --Streaming (type: GATHER)
Output: a, b, c
Node/s: dn_6005_6006
2 --Index Scan using index_a on public.test
Output: a, b, c
Distribute Key: a

===== Query Summary =====
-----
System available mem: 3358720KB
Query Max mem: 3358720KB
Query estimated mem: 1024KB
(24 rows)
explain verbose select * from test where a = 102 - 1;
QUERY PLAN
-----
id | operation | E-rows | E-distinct |
E-memory | E-width | E-costs
-----+-----+-----
1 | -> Streaming (type: GATHER) | 1 | |
| | 44 | 16.27
2 | -> Index Scan using index_a on public.test | 1 | |
1MB | | 44 | 8.27

Predicate Information (identified by plan id)
-----
2 --Index Scan using index_a on public.test
Index Cond: (test.a = 101)

Targetlist Information (identified by plan id)
-----
1 --Streaming (type: GATHER)
```

```

Output: a, b, c
Node/s: dn_6005_6006
2 --Index Scan using index_a on public.test
Output: a, b, c
Distribute Key: a

===== Query Summary =====
-----
System available mem: 3358720KB
Query Max mem: 3358720KB
Query estimated mem: 1024KB
(24 rows)
explain verbose select * from test where a + 1 = 102;
                                QUERY PLAN
-----
id | operation | E-rows | E-distinct | E-memory | E-
width | E-costs
-----+-----+-----+-----+-----+-----
1 | -> Streaming (type: GATHER) | 1 | | |
44 | 22.21
2 | -> Seq Scan on public.test | 1 | | 1MB |
44 | 14.21

Predicate Information (identified by plan id)
-----
2 --Seq Scan on public.test
Filter: ((test.a + 1) = 102)

Targetlist Information (identified by plan id)
-----
1 --Streaming (type: GATHER)
Output: a, b, c
Node/s: All datanodes
2 --Seq Scan on public.test
Output: a, b, c
Distribute Key: a

===== Query Summary =====
-----
System available mem: 3358720KB
Query Max mem: 3358720KB
Query estimated mem: 1024KB
(24 rows)

```

Método de optimización: use constantes en lugar de expresiones, o ponga el cálculo constante a la derecha del signo igual (=).

- Utilice funciones en columnas indexadas.

De acuerdo con el siguiente resultado de ejecución, si se usa una función en una columna indexada, el índice no se puede seleccionar.

```

explain verbose select * from test where to_char(c, 'yyyymmdd') =
to_char(CURRENT_DATE, 'yyyymmdd');
                                QUERY PLAN
-----
id | operation | E-rows | E-distinct | E-memory | E-
width | E-costs
-----+-----+-----+-----+-----+-----
1 | -> Streaming (type: GATHER) | 1 | | |
44 | 22.28
2 | -> Seq Scan on public.test | 1 | | 1MB |
44 | 14.28

Predicate Information
(identified by plan id)

```

```

-----
      2 --Seq Scan on public.test
          Filter: (to_char(test.c, 'yyyymmdd'::text) =
to_char(('2022-11-30'::pg_catalog.date)::timestamp with time zone,
'yyyymmdd'::text))

Targetlist Information (identified by plan id)
-----
      1 --Streaming (type: GATHER)
          Output: a, b, c
          Node/s: All datanodes
      2 --Seq Scan on public.test
          Output: a, b, c
          Distribute Key: a

===== Query Summary =====
-----
System available mem: 3358720KB
Query Max mem: 3358720KB
Query estimated mem: 1024KB
(24 rows)
explain verbose select * from test where c = current_date;
                                          QUERY PLAN
-----
 id | operation | E-rows | E-distinct |
E-memory | E-width | E-costs
-----+-----+-----+-----+-----+-----
  1 | -> Streaming (type: GATHER) | 1 | |
| | 44 | 16.27
  2 | -> Index Scan using index_c on public.test | 1 | |
1MB | 44 | 8.27

Predicate Information (identified by plan id)
-----
      2 --Index Scan using index_c on public.test
          Index Cond: (test.c = '2022-11-30'::pg_catalog.date)

Targetlist Information (identified by plan id)
-----
      1 --Streaming (type: GATHER)
          Output: a, b, c
          Node/s: All datanodes
      2 --Index Scan using index_c on public.test
          Output: a, b, c
          Distribute Key: a

===== Query Summary =====
-----
System available mem: 3358720KB
Query Max mem: 3358720KB
Query estimated mem: 1024KB
(24 rows)

```

Método de optimización: no utilice funciones innecesarias en columnas indexadas.

- Conversión implícita de tipos de datos.

Este escenario es común. Por ejemplo, el tipo de la columna **b** es Text y la condición de filtrado es **where b = 2**. Durante la generación del plan, el tipo Text se convierte implícitamente al tipo Bigint y la condición de filtrado real cambia a **where b::bigint = 2**. Como resultado, el índice de la columna **b** no es válido.

```

explain verbose select * from test where b = 2;
                                          QUERY PLAN
-----
 id | operation | E-rows | E-distinct | E-memory | E-

```

```
width | E-costs
-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+
1 | -> Streaming (type: GATHER) | 1 | | |
44 | 22.21
2 | -> Seq Scan on public.test | 1 | | 1MB |
44 | 14.21

Predicate Information (identified by plan id)
-----
2 --Seq Scan on public.test
  Filter: ((test.b)::bigint = 2)

Targetlist Information (identified by plan id)
-----
1 --Streaming (type: GATHER)
  Output: a, b, c
  Node/s: All datanodes
2 --Seq Scan on public.test
  Output: a, b, c
  Distribute Key: a

==== Query Summary ====
-----
System available mem: 3358720KB
Query Max mem: 3358720KB
Query estimated mem: 1024KB
(24 rows)
explain verbose select * from test where b = '2';
                                                    QUERY PLAN
-----
id | operation | E-rows | E-distinct |
E-memory | E-width | E-costs
-----+-----+-----+-----+-----+
1 | -> Streaming (type: GATHER) | 1 | | |
| | 44 | 16.27
2 | -> Index Scan using index_b on public.test | 1 | | |
1MB | | 44 | 8.27

Predicate Information (identified by plan id)
-----
2 --Index Scan using index_b on public.test
  Index Cond: (test.b = '2'::text)

Targetlist Information (identified by plan id)
-----
1 --Streaming (type: GATHER)
  Output: a, b, c
  Node/s: All datanodes
2 --Index Scan using index_b on public.test
  Output: a, b, c
  Distribute Key: a

==== Query Summary ====
-----
System available mem: 3358720KB
Query Max mem: 3358720KB
Query estimated mem: 1024KB
(24 rows)
```

Método de optimización: utilice constantes del mismo tipo que la columna indexada para evitar la conversión de tipos implícitos.

Escenario 4: Hashjoin se reemplaza con Nestloop + Indexscan.

Cuando se unen dos tablas, el número de filas en el conjunto de resultados filtrado por la condición WHERE en una tabla es pequeño, por lo que el número de filas en el conjunto de resultados final también es pequeño. En este caso, el efecto de nestloop+indexscan es mejor que el de hashjoin. El mejor plan de ejecución es el siguiente:

Puede ver que el índice Cond: (t1.b = t2.b) en la capa 5 ha empujado la condición de unión hacia abajo hasta el escaneo de la tabla base.

```
explain verbose select t1.a,t1.b from t1,t2 where t1.b=t2.b and t2.a=4;
 id | operation | E-rows | E-distinct | E-
memory | E-width | E-costs
-----+-----+-----+-----+-----+-----+-----
1 | -> Streaming (type: GATHER) | 26 | |
| | | 8 | 17.97
2 | -> Nested Loop (3,5) | 26 | |
1MB | | 8 | 11.97
3 | -> Streaming(type: BROADCAST) | 2 | |
2MB | | 4 | 2.78
4 | -> Seq Scan on public.t2 | 1 | |
1MB | | 4 | 2.62
5 | -> Index Scan using t1_b_idx on public.t1 | 26 | |
1MB | | 8 | 9.05
(5 rows)

Predicate Information (identified by plan id)
-----
4 --Seq Scan on public.t2
Filter: (t2.a = 4)
5 --Index Scan using t1_b_idx on public.t1
Index Cond: (t1.b = t2.b)
(4 rows)

Targetlist Information (identified by plan id)
-----
1 --Streaming (type: GATHER)
Output: t1.a, t1.b
Node/s: All datanodes
2 --Nested Loop (3,5)
Output: t1.a, t1.b
3 --Streaming(type: BROADCAST)
Output: t2.b
Spawn on: datanode2
Consumer Nodes: All datanodes
4 --Seq Scan on public.t2
Output: t2.b
Distribute Key: t2.a
5 --Index Scan using t1_b_idx on public.t1
Output: t1.a, t1.b
Distribute Key: t1.a
(15 rows)

===== Query Summary =====
-----
System available mem: 9262694KB
Query Max mem: 9471590KB
Query estimated mem: 5144KB
(3 rows)
```

Si el optimizador no selecciona dicho plan de ejecución, puede optimizarlo de la siguiente manera:

```
set enable_index_nestloop = on;
set enable_hashjoin = off;
set enable_seqscan = off;
```

Razón 5: El método de escaneo está especificado incorrectamente por las sugerencias.

Las sugerencias del plan de GaussDB(DWS) pueden especificar tres métodos de análisis: tablescan, indexscan y indexonlyscan.

- Exploración de tablas: exploración de tablas completas, como Seq Scan de tablas de almacenamiento de filas y CStore Scan de tablas de almacenamiento de columnas.
- Análisis de índices: analiza los índices y, a continuación, obtiene los registros de tabla basados en los índices.
- Análisis de solo índice: analiza índices, que cubren todos los resultados requeridos. En comparación con el análisis de índice, el análisis de solo índice cubre todas las columnas consultadas. De esta manera, solo se recuperan índices y no es necesario recuperar registros de datos.

En los escenarios de análisis de solo índice, el análisis de índice especificado por una sugerencia no será válido.

```
explain verbose select/*+ indexscan(test)*/ b from test where b = '1';
WARNING:  unused hint: IndexScan(test)
                                QUERY PLAN
-----
id | operation | E-rows | E-distinct
| E-memory | E-width | E-costs
-----+-----+-----+-----
1 | -> Streaming (type: GATHER) | 1 |
| | 32 | 16.27
2 | -> Index Only Scan using index_b on public.test | 1 |
| 1MB | 32 | 8.27

Predicate Information (identified by plan id)
-----
2 --Index Only Scan using index_b on public.test
  Index Cond: (test.b = '1'::text)

Targetlist Information (identified by plan id)
-----
1 --Streaming (type: GATHER)
  Output: b
  Node/s: All datanodes
2 --Index Only Scan using index_b on public.test
  Output: b
  Distribute Key: a

===== Query Summary =====
-----
System available mem: 3358720KB
Query Max mem: 3358720KB
Query estimated mem: 1024KB
(24 rows)
explain verbose select/*+ indexonlyscan(test)*/ b from test where b = '1';
                                QUERY PLAN
-----
id | operation | E-rows | E-distinct
| E-memory | E-width | E-costs
-----+-----+-----+-----
1 | -> Streaming (type: GATHER) | 1 |
| | 32 | 16.27
2 | -> Index Only Scan using index_b on public.test | 1 |
| 1MB | 32 | 8.27
```



```

Predicate Information (identified by plan id)
-----
2 --Index Only Scan using index_b on public.test
   Index Cond: (test.b = '1'::text)

Targetlist Information (identified by plan id)
-----
1 --Streaming (type: GATHER)
   Output: b
   Node/s: All datanodes
2 --Index Only Scan using index_b on public.test
   Output: b
   Distribute Key: a

===== Query Summary =====
-----
System available mem: 3358720KB
Query Max mem: 3358720KB
Query estimated mem: 1024KB
(24 rows)

```

Método de optimización: Especifique correctamente Análisis de índice y Análisis solo de índice.

Razón 6: Uso incorrecto del índice GIN en la recuperación de texto completo

Para acelerar la búsqueda de texto, puede crear un índice GIN para la búsqueda de texto completo.

```
CREATE INDEX idxb ON test using gin(to_tsvector('english',b));
```

Al crear el índice de GIN, debe usar la versión de 2 argumentos de `to_tsvector`. Solo cuando la consulta también utiliza la versión de 2 argumentos y los argumentos son los mismos que en el índice de Gin, se puede invocar al índice de GIN.

NOTA

La función `to_tsvector()` acepta uno o dos aumentos. Si se utiliza la versión de un aumento del índice, el sistema utilizará la configuración especificada por `default_text_search_config` de forma predeterminada. Para crear un índice, se debe usar la versión de dos aumentos o el contenido del índice puede ser inconsistente.

```

explain verbose select * from test where to_tsvector(b) @@ to_tsquery('cat')
order by 1;

```

QUERY PLAN						
id	operation	E-rows	E-distinct	E-memory	E-	E-
width	E-costs					
1	-> Streaming (type: GATHER)	2				
44	22.23					
2	-> Sort	2		16MB		
44	14.23					
3	-> Seq Scan on public.test	1		1MB		
44	14.21					

```

Predicate Information (identified by plan id)
-----
3 --Seq Scan on public.test
   Filter: (to_tsvector(test.b) @@ '''cat''':tsquery)

Targetlist Information (identified by plan id)
-----
1 --Streaming (type: GATHER)
   Output: a, b, c

```

```

Merge Sort Key: test.a
Node/s: All datanodes
2 --Sort
  Output: a, b, c
  Sort Key: test.a
3 --Seq Scan on public.test
  Output: a, b, c
  Distribute Key: a

===== Query Summary =====
-----
System available mem: 3358720KB
Query Max mem: 3358720KB
Query estimated mem: 1024KB
(29 rows)
explain verbose select * from test where to_tsvector('english',b) @@
to_tsquery('cat') order by 1;

                                QUERY PLAN
-----
id | operation | E-rows | E-distinct | E-memory
| E-width | E-costs |-----+-----+-----
+-----+-----+-----+-----+-----
1 | -> Streaming (type: GATHER) | 2 | |
| 44 | 20.03 |
2 | -> Sort | 2 | | 16MB
| 44 | 12.03 |
3 | -> Bitmap Heap Scan on public.test | 1 | | 1MB
| 44 | 12.02 |
4 | -> Bitmap Index Scan | 1 | | 1MB
| 0 | 8.00 |

Predicate Information (identified by plan id)
-----
3 --Bitmap Heap Scan on public.test
  Recheck Cond: (to_tsvector('english'::regconfig, test.b) @@
  'cat'::tsquery)
4 --Bitmap Index Scan
  Index Cond: (to_tsvector('english'::regconfig, test.b) @@
  'cat'::tsquery)

Targetlist Information (identified by plan id)
-----
1 --Streaming (type: GATHER)
  Output: a, b, c
  Merge Sort Key: test.a
  Node/s: All datanodes
2 --Sort
  Output: a, b, c
  Sort Key: test.a
3 --Bitmap Heap Scan on public.test
  Output: a, b, c
  Distribute Key: a

===== Query Summary =====
-----
System available mem: 3358720KB
Query Max mem: 3358720KB
Query estimated mem: 2048KB
(32 rows)

```

Método de optimización: utilice la versión de 2 argumentos de to_tsvector para la consulta y asegúrese de que los valores de los argumentos son los mismos que los del índice.

7.16 ¿Cómo uso una función definida por el usuario para reescribir la función CRC32()?

Actualmente, GaussDB(DWS) no tiene una función integrada de **CRC32()**. En su lugar, puede usar la función definida por el usuario de GaussDB(DWS) para reescribir la función **CRC32()**.

- **CRC32(expr)**
- Descripción: Calcula la redundancia cíclica. El parámetro de entrada **expr** es una cadena. Si el parámetro es NULL, se devuelve NULL. De lo contrario, se devuelve un valor sin signo de 32 bits después del cálculo de redundancia.

Ejemplo de reescritura de la función **CRC32()** usando la sentencia de función definida por el usuario de GaussDB(DWS):

```
CREATE OR REPLACE FUNCTION crc32(text_string text) RETURNS bigint AS $$
DECLARE
    val bigint;
    i int;
    j int;
    byte_length int;
    binary_string bytea;
BEGIN
    IF text_string is null THEN
        RETURN null;
    ELSIF text_string = '' THEN
        RETURN 0;
    END IF;

    i = 0;
    val = 4294967295;
    byte_length = bit_length(text_string) / 8;
    binary_string = decode(replace(text_string, E'\\', E'\\\\\\'), 'escape');
    LOOP
        val = (val # get_byte(binary_string, i))::bigint;
        i = i + 1;
        j = 0;
        LOOP
            val = ((val >> 1) # (3988292384 * (val & 1)))::bigint;
            j = j + 1;
            IF j >= 8 THEN
                EXIT;
            END IF;
        END LOOP;
        IF i >= byte_length THEN
            EXIT;
        END IF;
    END LOOP;
    RETURN (val # 4294967295);
END
$$ IMMUTABLE LANGUAGE plpgsql;
```

Verifique el resultado de reescritura.

```
select crc32(null), crc32(''), crc32('1');
   crc32 |  crc32 |   crc32
-----+-----+-----
         |      0 | 2212294583
(1 row)
```

Para obtener detalles sobre cómo usar funciones definidas por el usuario, consulte la sección [CREATE FUNCTION](#).

7.17 ¿Cuáles son los esquemas que comienzan con pg_toast_temp* o pg_temp*?

Cuando consulta la lista de esquemas, el resultado de la consulta puede contener esquemas que comiencen por **pg_temp*** o **pg_toast_temp*** como se muestra en la siguiente figura.

```
SELECT * FROM pg_namespace;
```

namespace	nspowner	nspacl	nspacl	perspace	usedspace
pg_toast	10	0		-1	0
cstore	10	0		-1	0
pg_logical_cluster	10	0		-1	0
sys	10	0		-1	0
dbes_cm	10	0	{Ruby=UC/Ruby,Ruby=LP/Ruby,=U/Ruby}	-1	24576
dbes_lab	10	0	{Ruby=UC/Ruby,Ruby=LP/Ruby,=U/Ruby}	-1	0
pg_catalog	10	0	{Ruby=UC/Ruby,Ruby=LP/Ruby,=U/Ruby}	-1	19088896
public	10	0	{Ruby=UC/Ruby,Ruby=LP/Ruby,=U/Ruby}	-1	822392
information_schema	10	0	{Ruby=UC/Ruby,Ruby=LP/Ruby,=U/Ruby}	-1	32796
utl_file	10	0	{Ruby=UC/Ruby,Ruby=LP/Ruby,=U/Ruby}	-1	0
dbes_output	10	0	{Ruby=UC/Ruby,Ruby=LP/Ruby,=U/Ruby}	-1	0
dbes_random	10	0	{Ruby=UC/Ruby,Ruby=LP/Ruby,=U/Ruby}	-1	0
utl_row	10	0	{Ruby=UC/Ruby,Ruby=LP/Ruby,=U/Ruby}	-1	0
dbes_spl	10	0	{Ruby=UC/Ruby,Ruby=LP/Ruby,=U/Ruby}	-1	0
dbes_lab	10	0	{Ruby=UC/Ruby,Ruby=LP/Ruby,=U/Ruby}	-1	0
scheduler	10	0		-1	8328
ui	24954	0		-1	0
ud	24953	0		-1	0
us	24952	0		-1	0
ud	24956	0		-1	0
s1	16833	0	{dbadmin=UC/dbadmin,dbadmin=LP/dbadmin,rs1_select=U/dbadmin,rs1_update=U/dbadmin,rs2_select=U/dbadmin,rs2_update=U/dbadmin}	-1	0
s2	16833	0	{dbadmin=UC/dbadmin,dbadmin=LP/dbadmin,rs1_select=U/dbadmin,rs1_update=U/dbadmin,rs2_select=U/dbadmin,rs2_update=U/dbadmin}	-1	0
pg_temp_cn_5003_4_1_201471110284072	10	0		-1	0
pg_toast_temp_cn_5003_4_1_201471110284272	10	0		-1	0
vz*	10	0		-1	0

Estos esquemas se crean cuando se crean tablas temporales. Cada sesión tiene un esquema independiente que comienza por **pg_temp** para asegurarse de que las tablas temporales solo son visibles para la sesión actual. Por lo tanto, no se recomienda eliminar manualmente los esquemas que comienzan con **pg_temp** o **pg_toast_temp** durante las operaciones rutinarias.

Las tablas temporales solo son visibles en la sesión actual y se eliminan automáticamente después de que finalice la sesión. También se eliminan los esquemas correspondientes.

7.18 Soluciones a los resultados incoherentes de las consultas a GaussDB(DWS)

En GaussDB(DWS), a veces una consulta de SQL puede obtener resultados diferentes. Es muy probable que este problema sea causado por una sintaxis o un uso inadecuados. Para evitar este problema, utilice la sintaxis correctamente. Los siguientes son algunos ejemplos de incoherencia en los resultados de la consulta junto con las soluciones.

Los resultados de la función de ventana están incompletos

Escenario:

En la función de ventana **row_number()**, la columna **c** de la tabla **t3** se consulta después de ordenar. Los dos resultados de la consulta son diferentes.

```
select * from t3 order by 1,2,3;
 a | b | c
---+---+---
 1 | 2 | 1
 1 | 2 | 2
 1 | 2 | 3
(3 rows)

select c,rn from (select c,row_number() over(order by a,b) as rn from t3) where
rn = 1;
 c | rn
---+---
 1 | 1
(1 row)

select c,rn from (select c,row_number() over(order by a,b) as rn from t3) where
```

```
rn = 1;
c | rn
---+---
3 | 1
(1 row)
```

Análisis:

Como se muestra anteriormente, ejecutar **select c,rn from (select c,row_number() over(order by a,b) as rn from t3) where rn = 1;** dos veces, los resultados son diferentes. Esto se debe a que existen valores duplicados **1** y **2** en las columnas de ordenación **a** y **b** de la función ventana, mientras que sus valores en la columna **c** son diferentes. Como resultado, cuando el primer registro se obtiene basándose en el resultado de la clasificación en las columnas **a** y **b** los datos obtenidos en la columna **c** son aleatorios, como resultado, los conjuntos de resultados son incoherentes.

Solución:

Los valores de la columna **c** deben agregarse a la ordenación.

```
select c,rn from (select c,row_number() over(order by a,b,c) as rn from t3) where
rn = 1;
c | rn
---+---
1 | 1
(1 row)

select c,rn from (select c,row_number() over(order by a,b,c) as rn from t3) where
rn = 1;
c | rn
---+---
1 | 1
(1 row)
```

Uso de la ordenación en subvistas/subconsultas

Escenario

Después de crear la tabla **test** y la vista **v**, los resultados de la consulta son incoherentes cuando se utiliza la ordenación para consultar las **test** de tabla en una subconsulta.

```
CREATE TABLE test(a serial ,b int);
INSERT INTO test(b) VALUES(1);
INSERT INTO test(b) SELECT b FROM test;
...
INSERT INTO test(b) SELECT b FROM test;
CREATE VIEW v as SELECT * FROM test ORDER BY a;
```

Problema de SQL:

```
select * from v limit 1;
a | b
---+---
3 | 1
(1 row)

select * from (select *from test order by a) limit 10;
a | b
---+---
14 | 1
(1 row)

select * from test order by a limit 10;
a | b
---+---
1 | 1
(1 row)
```

Análisis:

ORDER BY no es válido para subconsultas y subconsultas.

Solución:

No se recomienda utilizar **ORDER BY** en subvistas y subconsultas. Para asegurarse de que los resultados están en orden, utilice **ORDER BY** en la consulta más externa.

LIMIT en subconsultas

Escenario: Cuando se utiliza **LIMIT** en una subconsulta, los dos resultados de la consulta son incoherentes.

```
select * from (select a from test limit 1 ) order by 1;
a
---
5
(1 row)

select * from (select a from test limit 1 ) order by 1;
a
---
1
(1 row)
```

Análisis:

El **LIMIT** en la subconsulta provoca que se obtengan resultados aleatorios.

Solución:

Para garantizar la estabilidad del resultado final de la consulta, no utilice **LIMIT** en subconsultas.

Using String_agg

Escenario: cuando se utiliza **string_agg** para consultar la tabla **employee**, los resultados de la consulta son inconsistentes.

```
select * from employee;
empno | ename | job | mgr | hiredate | sal | comm | deptno
-----+-----+-----+-----+-----+-----+-----+-----
 7654 | MARTIN | SALEMAN | 7698 | 2022-11-08 00:00:00 | 12000 | 1400 | 30
 7566 | JONES | MANAGER | 7839 | 2022-11-08 00:00:00 | 32000 | 0 | 20
 7499 | ALLEN | SALEMAN | 7698 | 2022-11-08 00:00:00 | 16000 | 300 | 30
(3 rows)

select count(*) from (select deptno, string_agg(ename, ',') from employee group
by deptno) t1, (select deptno, string_agg(ename, ',') from employee group by
deptno) t2 where t1.string_agg = t2.string_agg;
count
-----
      2
(1 row)

select count(*) from (select deptno, string_agg(ename, ',') from employee group
by deptno) t1, (select deptno, string_agg(ename, ',') from employee group by
deptno) t2 where t1.string_agg = t2.string_agg;
count
-----
      1
(1 row)
```

Análisis:

La función **string_agg** se utiliza para concatenar datos de un grupo en una fila. Sin embargo, si utiliza **string_agg(ename, ',')** debe especificarse el orden de los resultados concatenados. Por ejemplo, en la instrucción anterior **select deptno, string_agg(ename, ',') from employee group by deptno;**

Puede generar cualquiera de las siguientes opciones:

```
30 | ALLEN,MARTIN
```

O:

```
30 | MARTIN,ALLEN
```

En el escenario anterior, el resultado de la subconsulta **t1** puede ser diferente de la subconsulta **t2** cuando deptno es de **30**.

Solución:

Agregue **ORDER BY** a **String_agg** para asegurarse de que los datos están concatenados en secuencia.

```
select count(*) from (select deptno, string_agg(ename, ',' order by ename desc)
from employee group by deptno) t1 ,(select deptno, string_agg(ename, ',' order by
ename desc) from employee group by deptno) t2 where t1.string_agg = t2.string_agg;
```

Modo de compatibilidad de bases de datos

Scenario: Los resultados de la consulta de cadenas vacías en la base de datos son inconsistentes.

database1 (compatible con TD):

```
td=# select '' is null;
 isnull
-----
 f
(1 row)
```

database 2 (compatible con ORA):

```
ora=# select '' is null;
 isnull
-----
 t
(1 row)
```

Análisis:

Los resultados de la consulta de cadenas vacías son diferentes porque la sintaxis de la cadena vacía es diferente de la de la cadena nula en diferentes compatibilidades de base de datos.

Actualmente, GaussDB(DWS) soporta tres tipos de compatibilidad de bases de datos: Oracle, TD y MySQL. La sintaxis y el comportamiento varían dependiendo de la compatibilidad. Para obtener detalles sobre las diferencias de compatibilidad, consulte [Diferencias de compatibilidad de sintaxis entre Oracle, Teradata y MySQL](#).

Las bases de datos en diferentes modos de compatibilidad tienen diferentes problemas de compatibilidad. Puede ejecutar **select datname, datcompatibility from pg_database;** para comprobar la compatibilidad de la base de datos.

Solución:

El problema se resuelve cuando los modos de compatibilidad de las bases de datos en los dos entornos se establecen en el mismo. El atributo **DBCMPATIBILITY** de una base de datos

no es compatible con **ALTER**. Solo se puede especificar el mismo atributo **DBCOMPATIBILITY** al crear una base de datos.

El elemento de configuración **behavior_compat_options** para comportamientos de compatibilidad de bases de datos se configura de manera inconsistente.

Escenario: Los resultados del cálculo de la función **add_months** son inconsistentes.

database1:

```
select add_months('2018-02-28',3) from dual;
add_months
-----
2018-05-28 00:00:00
(1 row)
```

database2:

```
select add_months('2018-02-28',3) from dual;
add_months
-----
2018-05-31 00:00:00
(1 row)
```

Análisis:

Algunos comportamientos varían según el elemento **behavior_compat_options** de configuración de compatibilidad de bases de datos. Para obtener más información sobre las opciones de parámetros, consulte [behavior_compat_options](#).

El **end_month_calculate** de **behavior_compat_options** controla la lógica de cálculo de la función **add_months**. Si se especifica este parámetro, y el **Day** de **param1** indica el último día de un mes inferior a **result** el **Day** en el resultado del cálculo será igual al de **result**.

Solución:

El parámetro **behavior_compat_options** debe configurarse de forma coherente. Este parámetro es del tipo **USERSET** y se puede establecer en el nivel de sesión o modificar en el nivel de clúster.

Los atributos de la función definida por el usuario no están configurados correctamente.

Escenario: Cuando se invoca la función personalizada **get_count()** los resultados son inconsistentes.

```
CREATE FUNCTION get_count() returns int
SHIPPABLE
as $$
declare
    result int;
begin
result = (select count(*) from test); --test table is a hash table.
    return result;
end;
$$
language plpgsql;
```

Invocar a esta función.

```
SELECT get_count();
get_count
-----
```



```
      2106  
(1 row)  
  
SELECT get_count() FROM t_src;  
get_count  
-----  
      1032  
(1 row)
```

Análisis:

Esta función especifica el atributo **SHIPPABLE**. Cuando se genera un plan, la función lo empuja hacia abajo a los DN para su ejecución. La tabla de prueba definida en la función es una tabla de hash. Por lo tanto, cada DN tiene solo parte de los datos en la tabla, el resultado devuelto por **select count(*) from test**; no es el resultado de los datos completos en la tabla de prueba. El resultado esperado cambia después de agregar **from**.

Solución:

Utilice cualquiera de los siguientes métodos (se recomienda el primer método):

1. Cambie la función para que no empujes hacia abajo: **ALTER FUNCTION get_count() not shippable;**
2. Cambie la tabla utilizada en la función a una tabla de replicación. De esta manera, los datos completos de la tabla se almacenan en cada DN. Incluso si el plan se empuja hacia abajo a los DN para su ejecución, el conjunto de resultados será el esperado.

Uso de la tabla sin registro

Escenario:

Una vez que se utiliza una tabla sin registro y se reinicia el clúster, el conjunto de resultados de consulta asociado es anormal y faltan algunos datos en la tabla sin registro.

Análisis:

Si **max_query_retry_times** se establece en **0** y la palabra clave **UNLOGGED** se especifica durante la creación de la tabla, la tabla creada será una tabla sin registro. Los datos escritos en tablas no registradas no se escriben en el log de escritura anticipada, lo que los hace considerablemente más rápidos que las tablas ordinarias. Sin embargo, una tabla no registrada se trunca automáticamente después de un bloqueo o un apagado sucio, incurriendo en riesgos de pérdida de datos. El contenido de una tabla no registrada tampoco se replica en los servidores en espera. Los índices creados en una tabla no registrada tampoco se registran automáticamente. Si el clúster se reinicia inesperadamente (reinicio del proceso, fallo del nodo o reinicio del clúster), algunos datos de la memoria no se vacían en los discos de manera oportuna y se pierden algunos datos, lo que provoca que el conjunto de resultados sea anormal.

Solución:

No se puede garantizar la seguridad de las tablas no registradas si el clúster falla. En la mayoría de los casos, las tablas no registradas solo se utilizan como tablas temporales. Si un clúster es defectuoso, debe reconstruir la tabla no registrada o hacer una copia de respaldo de los datos e importarlos a la base de datos de nuevo para asegurarse de que los datos son normales.

7.19 ¿En qué catálogos de sistema no se puede realizar la operación de VACUUM FULL?

VACUUM FULL se puede realizar en todos los catálogos de sistemas de GaussDB(DWS). Sin embargo, durante el proceso, se impondrán bloqueos de nivel 8 en los catálogos del sistema, y se bloquearán los servicios que involucran estos catálogos del sistema.

Las sugerencias se basan en las versiones de la base de datos:

8.1.3 y las versiones posteriores

- Para los clústeres de la versión 8.1.3 o posterior, el **AUTO VACUUM** está habilitado de forma predeterminada (controlado por el parámetro **autovacuum**). Después de establecer el parámetro, el sistema realiza automáticamente **VACUUM FULL** en todos los catálogos del sistema y tablas de almacén de filas.
 - Si el valor de **autovacuum_max_workers** es **0**, ni en los catálogos del sistema ni en las tablas ordinarias se realizará automáticamente **VACUUM FULL**.
 - Si **autovacuum** se establece en **off**, **VACUUM FULL** se ejecutará automáticamente en las tablas ordinarias, pero no en catálogos del sistema.
- Esto solo se aplica a las tablas de almacén de filas. Para activar automáticamente **VACUUM** para tablas de almacén de columnas, debe configurar tareas de programación inteligentes en la consola de gestión. Para obtener más detalles, consulte el [plan de O&M](#).

8.1.1 y versiones anteriores

1. La reforma de **VACUUM FULL** en los siguientes catálogos del sistema afecta a todos los servicios. Realice esta operación en una ventana de tiempo inactivo o cuando se detienen los servicios.
 - **pg_statistic** (Información estadística. Se recomienda no borrarla porque afecta al rendimiento de la consulta de servicio)
 - **pg_attribute**
 - **pgxc_class**
 - **pg_type**
 - **pg_depend**
 - **pg_class**
 - **pg_index**
 - **pg_proc**
 - **pg_partition**
 - **pg_object**
 - **pg_shdepend**
2. Los siguientes catálogos del sistema afectan a la supervisión de recursos y a las interfaces de consulta de tamaño de tabla, pero no a otros servicios.
 - **gs_wlm_user_resource_history**
 - **gs_wlm_session_info**

- gs_wlm_instance_history
 - gs_respool_resource_history
 - pg_relfilenode_size
3. Otros catálogos del sistema no ocupan espacio y no necesitan ser borrados.
 4. Durante la rutina de O&M, se le aconseja monitorear los tamaños de estos catálogos del sistema, y recopilar estadísticas cada semana. Si el espacio debe recuperarse, borre el espacio en función de los tamaños de las tablas del sistema.

La declaración es la siguiente:

```
SELECT c.oid,c.relname, c.relkind, pg_relation_size(c.oid) AS size FROM
pg_class c WHERE c.relkind IN ('r') AND c.oid <16385 ORDER BY size DESC;
```

7.20 ¿En qué escenarios estaría una declaración "idle in transaction"?

Cuando se consulta información de SQL de usuario en la vista **PGXC_STAT_ACTIVITY**, la columna **state** del resultado de la consulta a veces muestra **idle in transaction**. **idle in transaction** indica que el backend está en una transacción, pero no se está ejecutando ninguna instrucción. Este estado indica que se ha ejecutado una instrucción. Por lo tanto, el valor de **query_id** es 0, pero la transacción no se ha confirmado ni se ha revertido. Las sentencias en este estado se han ejecutado y no ocupan recursos de CPU y E/S, pero ocupan recursos de conexión tales como conexiones y conexiones simultáneas.

Si una declaración se encuentra en el estado **idle in transaction**, rectificar el fallo haciendo referencia a los siguientes escenarios y soluciones comunes:

Escenario 1: Se ha iniciado una transacción pero no se ha cometido, y la declaración se encuentra en el estado "inactivo en la transacción"

BEGIN/START TRANSACTION se ejecuta manualmente para iniciar una transacción. Después de ejecutar las sentencias, no se ejecuta **COMMIT/ROLLBACK**. Ver el **PGXC_STAT_ACTIVITY**:

```
SELECT state, query, query_id FROM pgxc_stat_activity;
```

El resultado muestra que la instrucción está en el estado **idle in transaction**.

state	query	query_id
active		0
idle		0
idle		0
active	WLM fetch collect info from data nodes	73464968921613282
active	WLM calculate space info process	0
active	WLM monitor update and verify local info	73464968921613276
active	WLM arbiter sync info by CCN and CNS	0
idle in transaction	select count(1) from t group by a order by 1 desc limit 1;	0
idle		0
active	select state,query,query_id from pgxc_stat_activity;	73464968921613283
active		0
idle		0
idle		0
active	WLM fetch collect info from data nodes	145522562959541153
active	WLM calculate space info process	0
active	WLM monitor update and verify local info	145522562959541123
active	WLM arbiter sync info by CCN and CNS	0
active	SELECT * FROM pg_stat_activity	73464968921613283
idle		0

(19 rows)

Solution: Ejecute manualmente COMMIT/ROLLBACK en la transacción iniciada.

Escenario 2: Después de ejecutar una sentencia de DDL en un procedimiento almacenado, otros nodos del procedimiento almacenado están en el estado "inactivo en transacción"

Crear un procedimiento almacenado:

```
CREATE OR REPLACE FUNCTION public.test_sleep()
RETURNS void
LANGUAGE plpgsql
AS $$
BEGIN
    truncate t1;
    truncate t2;
    EXECUTE IMMEDIATE 'select pg_sleep(6)';
    RETURN;
END$$;
```

Ver la vista PGXC_STAT_ACTIVITY:

```
SELECT coorname,pid,query_id,state,query,username FROM pgxc_stat_activity WHERE
username='jack';
```

El resultado muestra que **truncate t2** está en el estado **idle in transaction** y **coorname** es **coordinator2**. Esto indica que la instrucción se ha ejecutado en **cn2** y que el procedimiento almacenado está ejecutando la siguiente instrucción.

coorname	pid	query_id	state	query	username
coordinator1	139767124588288	73464968921614213	active	select test_sleep();	jack
coordinator2	140055318353664	0	idle in transaction	truncate t2	jack

(2 rows)

Solution: Este problema es causado por la ejecución lenta del procedimiento almacenado. Espere hasta que se complete la ejecución del procedimiento almacenado. También puede optimizar las instrucciones que se ejecutan lentamente en el procedimiento almacenado.

Escenario 3: Un gran número de declaraciones SAVEPOINT/RELEASE están en el estado "inactivo en la transacción" (Versiones del grupo anteriores a 8.1.0)

Ver la vista PGXC_STAT_ACTIVITY:

```
SELECT coorname,pid,query_id,state,query,username FROM pgxc_stat_activity WHERE
username='jack';
```

El resultado muestra que la instrucción SAVEPOINT/RELEASE está en el estado **idle in transaction**.

coorname	pid	query_id	state	query	username
coordinator1	140127877723904	77687093572141691	active	select test_sleep1();	jack
coordinator2	139773127153408	0	idle in transaction	release s1	jack
coordinator3	140193352906496	0	idle in transaction	release s1	jack

(3 rows)

Solución:

Las declaraciones **SAVEPOINT** y **RELEASE** son generadas automáticamente por el sistema cuando se ejecuta un procedimiento almacenado con **EXCEPTION**. En las versiones posteriores a 8.1.0, **SAVEPOINT** no se entrega a los CN. Los procedimientos almacenados en GaussDB(DWS) con **EXCEPTION** se implementan según subtransacciones, el mapeo es como sigue:

```
begin
  (Savepoint s1)
  DDL/DML
exception
  (Rollback to s1)
  (Release s1)
...
end
```

Si hay **EXCEPTION** en un procedimiento almacenado cuando se inicia, se iniciará una subtransacción. Si hay una excepción durante la ejecución, la transacción actual se revierte y se gestiona la excepción; si no hay excepción, se confirma la subtransacción.

Este problema puede ocurrir cuando hay muchos procedimientos almacenados y los procedimientos almacenados están anidados. Similar al escenario 2, solo tiene que esperar después de que se ejecute todo el procedimiento almacenado. Si hay un gran número de mensajes de **RELEASES**, el procedimiento almacenado desencadena varias excepciones. En este caso, debe analizar si la lógica del procedimiento almacenado es adecuada.

7.21 ¿Cómo implementa GaussDB(DWS) las conversiones de fila a columna y de columna a fila?

Esta sección describe cómo usar instrucciones de SQL para convertir filas en columnas y convertir columnas en filas en GaussDB(DWS).

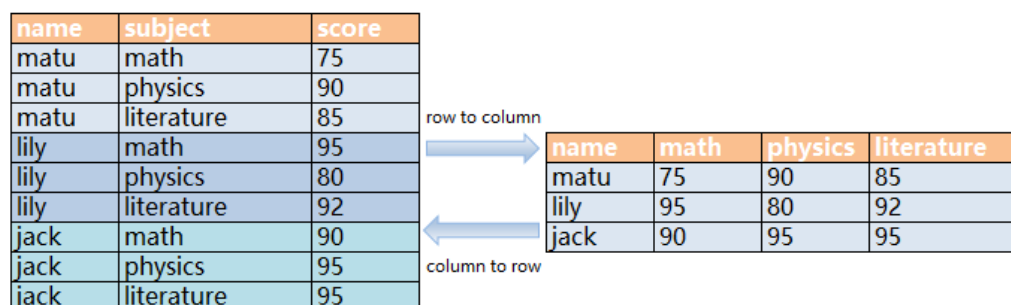
Escenario

Utilice una tabla de puntuación de un estudiante como ejemplo:

Los maestros registran la puntuación de cada asignatura de cada estudiante en una tabla, pero a los estudiantes solo les importan sus propias puntuaciones. Un estudiante necesita usar la conversión fila a columna para ver sus puntuaciones de todas las asignaturas. Si el profesor de una asignatura quiere ver las llagas de todos los estudiantes de esa asignatura, el profesor debe usar la conversión de columna a fila.

La siguiente figura muestra la conversión de fila a columna y de columna a fila.

Figura 7-1 Diagrama



- Conversión de filas a columnas
Convierta varias filas de datos en una fila o convierta una columna de datos en varias columnas.

- Conversión de columna a fila

Convierta una fila de datos en varias filas o convierta varias columnas de datos en una sola columna.

Ejemplo

- Cree un **students_info** de tabla de almacén de filas e inserte datos en la tabla.

```
CREATE TABLE students_info(name varchar(20),subject varchar(100),score
bigint) distribute by hash(name);
INSERT INTO students_info VALUES('lily','math',95);
INSERT INTO students_info VALUES('lily','physics',80);
INSERT INTO students_info VALUES('lily','literature',92);
INSERT INTO students_info VALUES('matu','math',75);
INSERT INTO students_info VALUES('matu','physics',90);
INSERT INTO students_info VALUES('matu','literature',85);
INSERT INTO students_info VALUES('jack','math',90);
INSERT INTO students_info VALUES('jack','physics',95);
INSERT INTO students_info VALUES('jack','literature',95);
```

Ver información sobre la tabla **students_info**.

```
SELECT * FROM students_info;
name | subject | score
-----+-----+-----
matu | math | 75
matu | physics | 90
matu | literature | 85
lily | math | 95
lily | physics | 80
lily | literature | 92
jack | math | 90
jack | physics | 95
jack | literature | 95
```

- Crear una tabla de almacenamiento de columnas **students_info1** e inserte datos en la tabla.

```
CREATE TABLE students_info1(name varchar(20), math bigint, physics bigint,
literature bigint) with (orientation = column) distribute by hash(name);
INSERT INTO students_info1 VALUES('lily',95,80,92);
INSERT INTO students_info1 VALUES('matu',75,90,85);
INSERT INTO students_info1 VALUES('jack',90,95,95);
```

Ver la información sobre la tabla **students_info1**.

```
SELECT * FROM students_info1;
name | math | physics | literature
-----+-----+-----+-----
matu | 75 | 90 | 85
lily | 95 | 80 | 92
jack | 90 | 95 | 95
(3 rows)
```

Conversión estática de filas a columnas

La conversión estática de fila a columna requiere que especifique manualmente los nombres de columna con los valores indicados. Si no se da ningún valor a una columna, se asigna el valor predeterminado **0** a la columna.

```
SELECT name,
sum(case when subject='math' then score else 0 end) as math,
sum(case when subject='physics' then score else 0 end) as physics,
sum(case when subject='literature' then score else 0 end) as literature FROM
students_info GROUP BY name;
name | math | physics | literature
-----+-----+-----+-----
matu | 75 | 90 | 85
lily | 95 | 80 | 92
```


Reconstruir la base de datos:

```
CALL build_view();
```

Consultar la vista:

```
SELECT * FROM get_score;
name | literature | physics | math
-----+-----+-----+-----
matu |          85 |         90 |    75
lily |          92 |         80 |    95
jack |          95 |         95 |    90
(3 rows)
```

Conversión de columna a fila

Use **UNION ALL** para fusionar materias (matemáticas, física y literatura) en una columna. A continuación se presenta un ejemplo:

```
SELECT * FROM
(
SELECT name, 'math' AS subject, math AS score FROM students_info1
union all
SELECT name, 'physics' AS subject, physics AS score FROM students_info1
union all
SELECT name, 'literature' AS subject, literature AS score FROM students_info1
)
order by name;
name | subject | score
-----+-----+-----
jack | math | 90
jack | physics | 95
jack | literature | 95
lily | math | 95
lily | physics | 80
lily | literature | 92
matu | math | 75
matu | physics | 90
matu | literature | 85
(9 rows)
```

7.22 ¿Cuáles son las diferencias entre las restricciones únicas y los índices únicos?

- Los conceptos de una restricción única y un índice único son diferentes.
Una restricción única especifica que los valores de una columna o un grupo de columnas son todos únicos. Si no se especifica **DISTRIBUTE BY REPLICATION**, la tabla de columnas que contiene solo valores únicos debe contener columnas de distribución.
Se utiliza un índice único para garantizar la unicidad de un valor de campo o la combinación de valores de varios campos. **CREATE UNIQUE UNINDEX** crea un índice único.
- Las funciones de una restricción única y un índice único son diferentes.
Las restricciones se utilizan para garantizar la integridad de los datos y los índices se utilizan para facilitar la consulta.
- Los usos de una restricción única y un índice único son diferentes.
 - a. Tanto las restricciones únicas como los índices únicos se pueden utilizar para garantizar la unicidad de los valores de columna que pueden ser NULL.

- b. Cuando se crea una restricción única, se crea automáticamente un índice único con el mismo nombre. El índice no se puede eliminar por separado. Cuando se suprime la restricción, el índice se suprime automáticamente. Una restricción única utiliza un índice único para garantizar la unicidad de los datos. Las tablas de almacén de filas de GaussDB(DWS) admiten restricciones únicas, pero las tablas de almacén de columnas no lo hacen.
- c. Un índice único creado es independiente y se puede eliminar por separado. Actualmente en GaussDB(DWS), los índices únicos solo se pueden crear usando B-Tree.
- d. Si desea tener una restricción única y un índice único en una columna y se pueden eliminar por separado, puede crear un índice único y, a continuación, una restricción única con el mismo nombre.
- e. Si un campo de una tabla se va a utilizar como clave externa de otra tabla, el campo debe tener una restricción única (o es una clave primaria). Si el campo solo tiene un índice único, se notifica un error.

Ejemplo: Crear un índice compuesto para dos columnas, que no es necesario que sea un índice único.

```
CREATE TABLE t (n1 number,n2 number);  
CREATE INDEX t_idx ON t(n1,n2);
```

Puede utilizar el **t_idx** de índice creado en el ejemplo anterior para crear un **t_uk** de restricción único, que solo es único en la columna **n1**. Una restricción única es más estricta que un índice único.

```
ALTER TABLE t ADD CONSTRAINT t_uk UNIQUE (n1) USING INDEX t_idx;
```

8 Rendimiento de bases de datos

8.1 ¿Por qué la ejecución de SQL es lenta después del uso largo de GaussDB(DWS)?

Después de utilizar una base de datos durante un período de tiempo, los datos de la tabla aumentan a medida que crecen los servicios, o los datos de la tabla se agregan, eliminan o modifican con frecuencia. Como resultado, se producen tablas de hinchazón y estadísticas inexactas, lo que deteriora el rendimiento de la base de datos.

Se le aconseja realizar periódicamente **VACUUM FULL** y **ANALYZE** en tablas que se agregan, eliminan o modifican con frecuencia. Realice las siguientes operaciones:

- Paso 1** De forma predeterminada, se recopilan 100 registros de estadísticas de cada 30,000. Cuando se trata de una gran cantidad de datos, la ejecución de SQL es inestable, lo que puede deberse a un plan de ejecución modificado. En este caso, es necesario ajustar la frecuencia de muestreo para las estadísticas. Puede ejecutar **set default_statistics_target** para aumentar la frecuencia de muestreo, lo que ayuda al optimizador a generar el plan óptimo.

```
gaussdb=> set default_statistics_target=-2;  
SET
```

- Paso 2** Realice **ANALYZE** de nuevo. Para obtener más información, consulte [ANALYZE | ANALYZE](#).

```
gaussdb=> ANALYZE customer_t1;  
ANALYZE
```

----Fin

NOTA

Para comprobar si los fragmentos de disco afectan al rendimiento de la base de datos, utilice la siguiente función:

```
select * from pgxc_get_stat_dirty_tables(30,100000);
```

8.2 ¿Por qué GaussDB(DWS) funciona peor que una base de datos de un solo servidor en los escenarios extremos?

Debido a la limitación de la arquitectura MPP de GaussDB(DWS), algunos métodos y funciones de PostgreSQL no se pueden enviar a los DN para su ejecución. Como resultado, se producen cuellos de botella de rendimiento en los CN.

Explicación:

- Una operación puede ejecutarse concurrentemente sólo cuando es lógicamente una operación concurrente. Por ejemplo, SUM realizada en todos los DN simultáneamente debe centralizar el resumen final en un CN. En este caso, la mayor parte del trabajo de resumen se ha completado en DN, por lo que el trabajo en el CN es relativamente ligero.
- En algunos escenarios, la operación debe ejecutarse centralmente en un nodo. Por ejemplo, la asignación de un nombre global único a un ID de transacción se implementa usando el sistema GTM. Por lo tanto, el GTM es también un componente único a nivel mundial (activo/en espera). Todas las tareas únicas a nivel mundial se implementan a través del GTM en GaussDB(DWS), pero el código de software está optimizado para reducir este tipo de tareas. Por lo tanto, el GTM no tiene muchos cuellos de botella. En algunos escenarios, GTM-Free y GTM-Lite pueden ser implementados.
- Para garantizar un excelente rendimiento, los servicios deben modificarse ligeramente para su adaptación durante la migración desde el modo de desarrollo de aplicaciones de la base de datos tradicional de un solo nodo al de la base de datos paralela, especialmente para el anidamiento de procedimientos almacenados tradicionales de Oracle.

Soluciones:

- Si se produce un problema de este tipo, consulte "Optimización del rendimiento de consultas" en la [Guía para desarrolladores de Data Warehouse Service \(DWS\)](#).
- Alternativamente, póngase en contacto con el soporte técnico para modificar y optimizar los servicios.

8.3 ¿Cómo puedo ver registros de ejecución de SQL en un cierto período cuando las solicitudes de lectura y escritura están bloqueadas?

Puede utilizar la característica de SQL superior para ver las sentencias de SQL ejecutadas en un período especificado. Se pueden ver las instrucciones de SQL del CN actual o de todos los CN.

TopSQL le permite ver las sentencias de SQL históricas y en tiempo real.

- Para obtener detalles sobre la consulta de sentencia SQL en tiempo real, consulte la sección [TopSQL en tiempo real](#).
- Para obtener detalles sobre cómo consultar sentencias de SQL históricas, consulte la sección [TopSQL histórico](#).

8.4 ¿Qué hago si mi clúster no está disponible debido a la insuficiencia de espacio?

Puede utilizar una instantánea para restaurar el clúster en uno nuevo que tenga mayor espacio de almacenamiento y, a continuación, eliminar el clúster antiguo para evitar el desperdicio de recursos. Puede aprender el almacenamiento en clúster si selecciona **Almacenamiento disponible**. Para restaurar un clúster nuevo, consulte [Restaurar una instantánea en un clúster nuevo](#). Para eliminar un clúster, consulte [Eliminación de un clúster](#).

NOTA

Este método solo es compatible con el almacén de datos estándar.

8.5 ¿Qué es el derrame del operador en GaussDB(DWS)?

Durante la ejecución de la consulta, si la memoria del clúster es insuficiente, la base de datos elegirá almacenar los resultados temporales en el disco. Cuando el almacenamiento en disco utilizado por los resultados temporales excede un cierto valor, los usuarios recibirán una alarma: "data spilling exceeds the threshol" (el derrame de datos excede el umbral). ¿Qué significa derrame?

Derrame del operador a los discos

Cualquier ordenador consume espacio de memoria. Si se consume demasiada memoria, la memoria para ejecutar otros trabajos será insuficiente, lo que hará que se ejecute un trabajo inestable. Por lo tanto, debe limitar el uso de memoria de las instrucciones de consulta para garantizar la estabilidad de ejecución del trabajo.

Si un trabajo que se ejecuta requiere 500 MB de memoria pero solo se le asignan 300 MB de memoria, los datos que no se utilizan temporalmente deben escribirse en los discos, y solo los datos que se están utilizando se conservan en la memoria. Esto se llama derrame de datos. También se llama **operator spilling**. El gran tamaño de los datos derramados al disco puede causar un uso del disco del 100%. Si sucede, la base de datos pasará a ser de solo lectura y el rendimiento de la consulta se verá muy afectado. Por lo tanto, GaussDB(DWS) impone un límite al derrame del operador. Si el derrame del operador excede el límite, se notifica un error y la consulta sale.

¿Qué operadores pueden derramar?

Los operadores que pueden derramarse al disco son **Hash(VecHashJoin)**, **Agg(VecAgg)**, **Sort(VecSort)**, **Material(VecMaterial)**, **SetOp(VecSetOp)** y **WindowAgg(VecWindowAgg)**. Pueden estar vectorizados o no vectorizados.

¿Qué parámetros se pueden utilizar para controlar el derrame del operador?

- **work_mem**: establece el umbral de derrame. El uso del disco que exceda este parámetro causará que el operador se derrame. Este parámetro solo tiene efecto cuando la memoria no es autoadaptable. (**enable_dynamic_workload=off**). Garantiza el rendimiento simultáneo y el rendimiento de un único trabajo de consulta. Por lo tanto, es necesario optimizar el parámetro basado en la salida de **Explain Performance**.

- **temp_file_limit**: limita el tamaño de los archivos derramados en los discos. Se recomienda establecer este parámetro en función de los requisitos del sitio para evitar que los archivos derramados consuman el espacio en disco. Los archivos derramados que excedan el valor de este parámetro provocarán un error.

¿Cómo sé si un estado de cuenta se derrama en discos?

- Compruebe los archivos de derrames. Los archivos de derrame se almacenan en el directorio **base/pgsql_tmp** del directorio de instancia. Los archivos de derrames se denominan *pgsql_tmp\$queryid_\$pid*. Puede determinar qué instrucción de SQL se derrama en el disco basándose en el **queryid**.
- Verifique la vista de espera (**pgxc_thread_wait_status**). Si se muestra **write file** en la vista de espera, hay resultados temporales derramados a los discos.
- Verifique el plan de ejecución (**EXPLAIN PERFORMANCE**). Palabras clave como **spill**, **written disk** y **temp file num** indican que hay un operador que se derrama.
- Compruebe si la columna **spill_info** en **TopSQL en tiempo real** o **Historical TopSQL** contiene la información derrame. Si esta columna no está vacía, los datos se han derramado a los discos de los DN. (Requisito: Se ha habilitado la función topsql.)

¿Cómo evito el derrame?

Cuando los operadores se derraman en los discos, los datos de cálculo del operador se escriben en los discos. En comparación con el acceso a la memoria, las operaciones del disco son lentas, lo que provoca un deterioro del rendimiento y el deterioro del tiempo de respuesta a las consultas. Por lo tanto, trate de evitar que el operador se derrame durante la ejecución de la consulta. Se recomienda utilizar los siguientes métodos:

- Reducir el conjunto de resultados intermedios: Si el conjunto de resultados intermedios es demasiado grande, puede agregar criterios de filtro para reducir el tamaño del conjunto de resultados intermedios.
- Evitar el sesgo de datos: Si hay un sesgo de datos grave, se producirá un derrame en un DN que tiene una gran cantidad de datos.
- Realizar **ANALYZE** de manera oportuna: cuando las estadísticas son inexactas, se puede estimar que el número de filas es menor. Como resultado, el plan no es óptimo y los datos se derraman a los discos.
- Realizar optimización de un solo punto: Realizar optimización en sentencias SQL individuales.
- Si la memoria no es autoadaptable, y el conjunto de resultados intermedio no se puede reducir, aumente el valor de **work_mem** como sea apropiado.
- Si la memoria es autoadaptable, aumente la memoria disponible de la base de datos tanto como sea posible para reducir la probabilidad de derramamiento de datos.

8.6 Gestión de recursos de CPU de GaussDB(DWS)

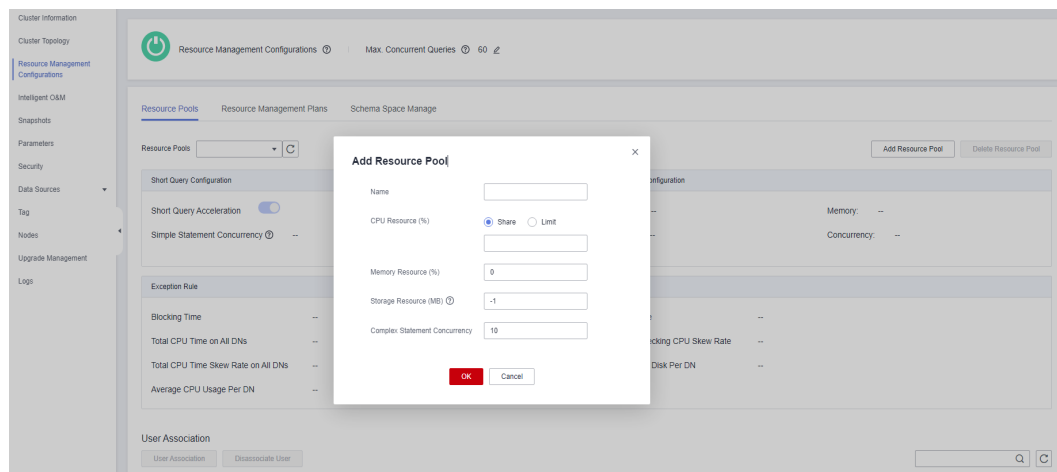
Descripción de la gestión de recursos de CPU

En diferentes escenarios de servicio, los recursos del sistema (CPU, memoria, E/S y recursos de almacenamiento) de la base de datos se asignan correctamente a las consultas para garantizar el rendimiento de las consultas y la estabilidad del servicio.

GaussDB(DWS) proporciona la función de gestión de recursos. Puede poner recursos en diferentes grupos de recursos, que están aislados entre sí. A continuación, puede asociar usuarios de base de datos a estos grupos de recursos. Cuando un usuario inicia una consulta de SQL, la consulta se transferirá al grupo de recursos asociado con el usuario. Puede especificar el número de consultas que se pueden ejecutar simultáneamente en un grupo de recursos, el límite superior de memoria utilizado para una sola consulta y los recursos de memoria y CPU que puede utilizar un grupo de recursos. De esta manera, puede limitar y aislar los recursos ocupados por diferentes cargas de trabajo.

GaussDB(DWS) utiliza cgroups para gestionar y controlar recursos de CPU, involucrando los subsistemas de CPU, de cpuacct y de cpuset. El recurso compartido de CPU se implementa basándose en el subsistema de CPU cpu.shares. Las ventajas de compartir la CPU son las siguientes: El control de la CPU no se activa cuando la CPU del SO no está completamente ocupada. El límite de CPU se implementa basándose en cpuset, que es un subsistema de CPU usado para monitorear el uso de recursos de CPU.

Al agregar un grupo de recursos en la consola de gestión de GaussDB(DWS), debe elegir entre **Share** y **Limit**.



Compartir CPU

CPU Share: Porcentaje de tiempo de CPU que pueden utilizar los usuarios asociados al grupo de recursos actual para ejecutar trabajos.

La acción tiene dos significados:

- **Compartir:** La CPU es compartida por todos los Cgroups. Los recursos de CPU inactivos pueden ser utilizados por otros Cgroups.
- **Cuota:** Cuando la CPU está completamente cargada durante las horas pico, los Cgroups prefieren los recursos de CPU en función de sus cuotas.

El uso compartido de CPU se implementa basado en cpu.shares y solo tiene efecto cuando la CPU está completamente cargada. Cuando la CPU está inactiva, no hay garantía de que un Cgroup se adelantará a los recursos de CPU apropiados a su cuota. Todavía puede haber contención de recursos cuando la CPU está inactiva. Las tareas en un Cgroup pueden usar recursos de CPU sin restricciones. Aunque el uso promedio de CPU puede no ser alto, la contención de recursos de CPU todavía puede ocurrir en un momento específico.

Por ejemplo, 10 trabajos se están ejecutando en 10 CPU, y un trabajo se está ejecutando en cada CPU. En este caso, cualquier solicitud de trabajo para recursos de CPU se responderá

instantáneamente, y no hay ninguna contención. Si se ejecutan 20 trabajos en 10 CPU, es posible que el uso de CPU aún no sea alto porque los trabajos no siempre ocupan la CPU y pueden esperar recursos de E/S y de red. Los recursos de CPU parecen estar inactivos. Sin embargo, si 2 o más trabajos solicitan una CPU al mismo tiempo, se produce una contención de recursos de CPU, lo que afecta al rendimiento del trabajo.

Límite de CPU

CPU Limit: especifica el porcentaje del número máximo de núcleos de CPU que puede utilizar un usuario de base de datos en el grupo de recursos.

El límite tiene dos significados:

- **Dedicado:** Los recursos de CPU están dedicados a un Cgroup. Incluso la parte inactiva no puede ser utilizada por otros Cgroups.
- **Cuota:** Solo se pueden usar los recursos de CPU en la cuota asignada. Los recursos de CPU inactivos de otros Cgroups no se pueden anular.

El límite de CPU se implementa basado en `cpuset.cpu`. Puede establecer una cuota adecuada para implementar el aislamiento absoluto de los recursos de CPU entre Cgroups. De esta manera, las tareas de los diferentes Cgroups no se afectarán entre sí. Sin embargo, el aislamiento absoluto de la CPU hará que los recursos inactivos de la CPU en un Cgroup sean desperdiciados. Por lo tanto, el límite no puede ser demasiado grande. Un límite mayor puede no traer un mejor rendimiento.

Por ejemplo, en un caso, 10 trabajos se ejecutan en 10 CPU y el uso promedio de CPU es de aproximadamente el 5%. En otro caso, 10 trabajos se ejecutan en 5 CPUs y el uso promedio de CPU es de aproximadamente el 10%. De acuerdo con el análisis anterior, aunque el uso de CPU es bajo cuando se ejecutan 10 trabajos en cinco CPU. Sin embargo, todavía existe contención de recursos de CPU. Por lo tanto, el rendimiento de ejecutar 10 trabajos en 10 CPU es mejor que el de ejecutar 10 trabajos en 5 CPUs. Pero más CPU no siempre es mejor. Si diez trabajos se ejecutan en 20 CPUs, en cualquier momento, al menos 10 CPUs están inactivas. Por lo tanto, teóricamente, ejecutar 10 trabajos en 20 CPUs no tiene mejor rendimiento que ejecutar 10 CPUs. Para un Cgroup con una concurrencia de N , si el número de CPU asignadas es menor que N , el rendimiento del trabajo es mejor con más CPU. Sin embargo, si el número de CPU asignadas es mayor que N , el rendimiento del trabajo no se mejorará con más CPU.

Escenarios de aplicación de la gestión de recursos de CPU

El límite de CPU y la cuota de CPU tienen sus propias ventajas y desventajas. El uso compartido de la CPU puede utilizar completamente los recursos de la CPU. Sin embargo, los recursos de diferentes Cgroups no están completamente aislados, lo que puede afectar al rendimiento de la consulta. El límite de CPU puede implementar el aislamiento absoluto de los recursos de CPU. Sin embargo, los recursos de CPU inactivos se desperdiciarán. En comparación con el límite de CPU, el uso compartido de CPU tiene un mayor uso de CPU y un rendimiento general de trabajo. En comparación con el uso compartido de CPU, el límite de CPU tiene un aislamiento completo de la CPU, lo que puede satisfacer mejor los requisitos de los usuarios sensibles al rendimiento.

Si se produce contención de CPU cuando se ejecutan varios tipos de trabajos en el sistema de base de datos, puede seleccionar diferentes modos de control de recursos de CPU en función de diferentes escenarios.

- **Escenario 1:** Utilice completamente los recursos de la CPU. Concéntrese en el rendimiento general de la CPU en lugar del rendimiento de un solo tipo de trabajos.

Sugerencia: No se recomienda aislar CPUs entre usuarios. No importa qué tipo de control de CPU se implemente, el uso general de CPU se ve afectado.

- Escenario 2: Se permite un cierto grado de contención de recursos de CPU y pérdida de rendimiento. Cuando la CPU está inactiva, los recursos de la CPU se utilizan completamente. Cuando la CPU está completamente cargada, cada tipo de servicio necesita usar la CPU proporcionalmente.

Sugerencia: Puede utilizar el uso compartido de CPU para mejorar el uso general de la CPU mientras implementa el aislamiento y el control de la CPU cuando las CPU están completamente cargadas.

- Escenario 3: Algunos trabajos son sensibles al rendimiento y se permite el desperdicio de recursos de CPU.

Sugerencia: Puede usar el límite de CPU para implementar el aislamiento absoluto de CPU entre diferentes tipos de trabajos.

8.7 ¿Por qué las tareas ejecutadas por un usuario ordinario son más lentas que las ejecutadas por el usuario dbadmin?

La velocidad de ejecución de un usuario ordinario es más lenta que la del usuario dbadmin en los siguientes escenarios:

Escenario 1: Los usuarios ordinarios están sujetos a la gestión de recursos.

Usuarios ordinarios en cola: **waiting in queue/waiting in global queue/waiting in ccn queue**

1. Los usuarios ordinarios serán **waiting in queue/waiting in global queue** cuando el número de sentencias activas exceda el valor de **max_active_statements**. Mientras que los administradores no necesitan cola.

Puede aumentar el valor de este parámetro o borrar algunas instrucciones para evitar la cola.

Cambie el valor de **max_active_statements** en la consola de gestión.

- a. Inicie sesión en la consola de gestión de GaussDB(DWS).
 - b. En el árbol de navegación de la izquierda, haga clic en **Clusters**.
 - c. En la lista de clústeres, busque el clúster objetivo y haga clic en el nombre del clúster. Se muestra **Basic Information**.
 - d. Vaya a la página **Parameter Modifications** del clúster, busque el parámetro **max_active_statements**, cambie su valor y haga clic en **Save**.
2. Se necesita mucho tiempo para que los usuarios ordinarios esperen en la cola ccn. Cuando se habilita la gestión dinámica de recursos (**enable_dynamic_workload** se establece en **on**), si la simultaneidad es alta y la memoria disponible es pequeña, los usuarios comunes pueden entrar en este estado al ejecutar instrucciones. Los administradores no están controlados. Puede detener algunas instrucciones o aumentar el valor del parámetro de memoria. Si el uso de memoria de cada DN no es alto, puede deshabilitar el parámetro de gestión dinámica de recursos **enable_dynamic_workload** definiéndolo en **off**.

Escenario 2: La condición OR en el plan de ejecución comprueba las sentencias ejecutadas por los usuarios comunes una por una. Esto consume mucho tiempo.

Las condiciones **OR** de los planes de ejecución contienen comprobaciones relacionadas con los permisos. Este escenario suele ocurrir cuando se utiliza la vista del sistema. Por ejemplo, en la siguiente instrucción de SQL:

```
SELECT distinct(dtp.table_name),
ta.table_catalog,
ta.table_schema,
ta.table_name,
ta.table_type
from information_schema.tables ta left outer join DBA_TAB_PARTITIONS dtp
on (dtp.schema = ta.table_schema and dtp.table_name = ta.table_name)
where ta.table_schema = 'public';
```

Parte del plan de ejecución es el siguiente:

En la vista de sistema, se utiliza la condición **OR** para la comprobación de permisos.

```
pg_has_role(c.reowner, 'USAGE'::text) OR has_table_privilege(c.oid, 'SELECT,
INSERT, UPDATE, DELETE, TRUNCATE, REFERENCES, TRIGGER'::text) OR
has_any_column_privilege(c.oid, 'SELECT, INSERT, UPDATE, REFERENCES'::text)
```

true siempre se devuelve para **pg_has_role** del uso **dbadmin**. Por lo tanto, no es necesario comprobar las condiciones después del **OR**.

Mientras que las condiciones **OR** de un usuario ordinario necesitan ser comprobadas una por una. Si hay un gran número de tablas en la base de datos, el tiempo de ejecución del usuario ordinario es más largo que el del usuario **dbadmin**.

En este escenario, si el número de conjuntos de resultados de salida es pequeño, puede establecer **set enable_hashjoin** y **enable_seqscan** en **off** para usar el plan **index+nestloop**.

Escenario 3: Los grupos de recursos asignados a los usuarios y administradores ordinarios son diferentes.

Ejecute el siguiente comando para comprobar si los grupos de recursos correspondientes a un usuario ordinario son los mismos que los del usuario administrador. Si son diferentes, compruebe si los recursos del tenant asignados a los dos usuarios son diferentes.

```
select * from pg_user;
```

8.8 ¿Cuáles son los factores relacionados con el rendimiento de la consulta de tabla única en GaussDB(DWS)?

GaussDB(DWS) utiliza la arquitectura de nada compartido, y los datos se almacenan de manera distribuida. Por lo tanto, la clave de distribución, el volumen de datos y el número de particiones afectan al rendimiento general de la consulta de una sola tabla.

1. Diseño de clave de distribución

Por defecto, GaussDB(DWS) toma la primera columna de la clave primaria como clave de distribución. Si ambos están definidos, la clave principal debe contener la clave de distribución. Las claves de distribución determinan la distribución de datos entre particiones. Si las claves de distribución están bien distribuidas entre particiones, se puede mejorar el rendimiento de la consulta.

Si la clave de distribución está seleccionada incorrectamente, puede producirse un sesgo de datos después de importar los datos. El uso de algunos discos puede ser mucho más alto que el de otros discos, y el clúster puede convertirse en solo lectura en algunos casos extremos. La selección adecuada de las claves de distribución es fundamental para el rendimiento de las consultas de tabla. Además, las claves de distribución adecuadas permiten crear y mantener índices de datos más rápidamente.

2. Volumen de datos almacenado en una sola tabla

Cuanto mayor sea la cantidad de datos almacenados en una sola tabla, menor será el rendimiento de la consulta. Si una tabla contiene una gran cantidad de datos, debe almacenar los datos en particiones. Para convertir una tabla ordinaria en una tabla particionada, debe crear una tabla particionada e importar datos a ella desde la tabla ordinaria. Cuando diseñe tablas, planifique si desea utilizar tablas particionadas en función de los requisitos de servicio.

Para particionar una tabla, cumpla con los siguientes principios:

- Utilice campos con rangos obvios para particionar, por ejemplo, fecha o región.
- El nombre de la partición debe reflejar las características de los datos de la partición. Por ejemplo, su formato puede ser la palabra clave + características de rango.
- Establezca el límite superior de una partición en **MAXVALUE** para evitar el desbordamiento de datos.

3. Número de particiones

Las tablas e índices se pueden dividir en unidades más pequeñas y fáciles de gestionar. Esto reduce significativamente el espacio de búsqueda y mejora el rendimiento del acceso.

El número de particiones afecta al rendimiento de la consulta. Si el número de particiones es demasiado pequeño, el rendimiento de la consulta puede deteriorarse.

GaussDB(DWS) soporta el particionamiento de rango y el particionamiento de lista. En la partición de rango, los registros se dividen e insertan en varias particiones de una tabla. Cada partición almacena datos de un rango específico (los rangos en diferentes particiones no se superponen). El particionamiento de listas solo es compatible con clústeres de 8.1.3 y versiones posteriores.

Al diseñar un almacén de datos, debe tener en cuenta estos factores y realizar experimentos para determinar el esquema de diseño óptimo.

9 Copia de respaldo y restauración de instantáneas

9.1 ¿Por qué la creación de una instantánea automatizada es tan lenta?

Esto ocurre cuando los datos que se van a hacer una copia de seguridad son grandes. Las instantáneas automatizadas son copias de seguridad incrementales, y cuanto menor sea la frecuencia establecida (por ejemplo, una semana), más tiempo tardará. Aumente la frecuencia de backup para acelerar el proceso.

En la siguiente tabla se enumeran las tasas de copia de respaldo y restauración de instantáneas. (Las tasas se obtienen del entorno de pruebas de laboratorio con SSD locales como medio de copia de respaldo. Las tarifas son solo de referencia. La velocidad real depende de los recursos de disco, red y ancho de banda.)

- Velocidad de copia de respaldo: 200 MB/s/DN
- Tasa de restauración: 125 MB/s/DN

9.2 ¿Una instantánea de DWS tiene la misma función que una instantánea de EVS?

No.

Las instantáneas de GaussDB(DWS) se utilizan para restaurar todas las configuraciones y los datos de servicio de un clúster. Las instantáneas de EVS se utilizan para restaurar los datos de servicio de un disco de datos o disco de sistema dentro de un período de tiempo específico.

Instantánea de DWS

Una instantánea de GaussDB(DWS) es una copia de respaldo completa o incremental de un clúster de GaussDB(DWS) en un punto específico en el tiempo. Registra los datos actuales de la base de datos y la información del clúster, incluido el número de nodos, las especificaciones de nodo y el nombre del administrador. Las instantáneas se pueden crear de forma manual o automática.

Cuando se utiliza una instantánea para la restauración, GaussDB(DWS) crea un nuevo clúster basado en la información del clúster registrada en la instantánea y restaura los datos de la instantánea.

Para obtener más información acerca de las instantáneas, consulte [Gestión de instantáneas](#) .

Instantánea de EVS

Una instantánea de EVS es una copia o imagen completa de los datos del disco tomados en un punto de tiempo específico. Instantánea es un enfoque importante de recuperación ante desastres, y puede restaurar completamente los datos de una instantánea al momento en que se creó la instantánea.

Puede crear las instantáneas para guardar rápidamente los datos del disco en puntos de tiempo especificados. Además, puede utilizar las instantáneas para crear nuevos discos de modo que los discos creados contengan los datos de instantáneas al principio.

Puede crear instantáneas para guardar rápidamente los datos del disco en puntos de tiempo especificados para implementar la recuperación ante desastres de datos.

- Si se produce una pérdida de datos, puede utilizar una instantánea para restaurar completamente los datos en el momento en que se creó la instantánea.
- Puede utilizar instantáneas para crear nuevos discos de modo que los discos creados contengan los datos de instantáneas.

Para obtener más información acerca de las instantáneas, consulte [Instantáneas de EVS](#).

10 Facturación

10.1 ¿Cómo puedo renovar el servicio?

Acerca de la renovación

Actualmente, GaussDB(DWS) es compatible con el modo de facturación de pago por uso y el modo de facturación de descuento.

- Modo de pago por uso: El sistema liquida las tarifas por hora. Puede utilizar el servicio siempre y cuando el saldo de su cuenta sea suficiente. Si el saldo de su cuenta es insuficiente, estará en mora. Recargue su cuenta de manera oportuna.
- Modo de paquete de descuento: Usted realiza un pago único y no se incurrirá en ningún cargo cuando utilice el servicio. Después de que el paquete caduque, el sistema le cobrará automáticamente en modo de pago por uso. El servicio no se interrumpirá mientras el saldo de su cuenta sea suficiente. Puede renovar su paquete en la página de **Renewals**.

Modos de renovación

Paquetes de descuento:

Realice las siguientes operaciones para renovar su cuenta:

Paso 1 Inicie sesión en la consola de gestión.

Paso 2 Seleccione **Billing Center** > **Renewal** en la parte superior derecha de la página.

Paso 3 En la página que se muestra, encuentre el paquete de descuento GaussDB(DWS) pertinente en la lista y seleccione el modo de renovación que prefiera.

Paso 4 Complete la renovación como se le solicite.

- No puede cambiar el modo de facturación de un pedido durante la renovación.
- Después de pagar por un nuevo pedido, solo se aplicará después de que expire el pedido anterior.

----Fin

Pago por uso

Para recargar su cuenta, realice los siguientes pasos:

- Paso 1** Inicie sesión en la consola de gestión.
- Paso 2** Seleccione **Billing Center > Renewal** en la parte superior derecha de la página.
- Paso 3** En el panel de navegación de la izquierda, haga clic en **Overview**. A continuación, haga clic en **Top Up** para que su saldo sea suficiente para continuar con los servicios.

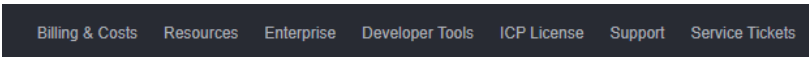
----Fin

10.2 ¿Se admite el reembolso?

Sí.

Darse de baja de un paquete de descuento

- Paso 1** Inicie sesión en la consola de gestión.
- Paso 2** En la esquina superior derecha, haga clic en **Billing & Costs**.



- Paso 3** En el árbol de navegación de la izquierda, elija **Orders > Unsubscriptions**.
- Paso 4** Busque el recurso en la lista, haga clic en **Unsubscribe from Resources** en la fila donde se encuentra el recurso y cancele la suscripción del recurso según se le solicite.

Para obtener más información, vaya a [Reglas de cancelación de suscripción](#).

----Fin

10.3 ¿Por qué se me dedujeron las tarifas después de que venciera mi prueba gratuita de GaussDB(DWS)?

Como se indica en la página de actividad de GaussDB(DWS), el clúster que creó no se liberará automáticamente después de que expire la prueba gratuita. Si no elimina el clúster manualmente, el sistema deducirá la tarifa por el clúster.

Para evitar cargos adicionales, inicie sesión en la consola de gestión de GaussDB(DWS) y vaya a la página de **Clusters** para eliminar su clúster si no desea usarlo después de que finalice el período de prueba. Si el clúster tiene una EIP, seleccione **Release the EIP bound to the cluster** en el cuadro de diálogo **Delete Cluster**. Si no libera la EIP, se facturará según la regla de precios de la EIP de VPC.

10.4 ¿Por qué no puedo ver un clúster después de suscribirme a una prueba gratuita de GaussDB(DWS)?

El sistema no crea automáticamente un clúster después de suscribirse a una prueba gratuita. Inicie sesión en la consola de gestión de GaussDB(DWS) para crear una manualmente.

10.5 ¿Por qué no se muestra ningún clúster después de comprar un paquete de descuento?

Comprar un paquete de descuento y crear un clúster son procesos diferentes. Un paquete de descuento es un concepto de facturación y puede considerarse como un cupón. Cuando compra un paquete de descuento, el sistema no crea automáticamente un clúster. Necesita ir a la consola de GaussDB(DWS) para crear un clúster con la variante de región y nodo correspondiente.

Si la variante de nodo de un clúster creado es diferente del de un paquete de descuento comprado, el clúster no se mostrará. Por lo tanto, asegúrese de que las variantes de nodo sean consistentes.

10.6 ¿Por qué todavía se deducen las tarifas después de darme de baja de un paquete de descuento?

Los cargos de pago por uso siguen siendo aplicables después de cancelar la suscripción si no elimina manualmente el clúster que creó.

Los grupos de prueba gratuitos son válidos durante un mes, después de lo cual todavía se aplican cargos de pago por uso.

Para evitar cargos adicionales, inicie sesión en la consola de gestión de GaussDB(DWS) y vaya a la página de **Clusters** para eliminar el clúster que ya no usa. Si el clúster tiene una EIP, seleccione **Release the EIP bound to the cluster**. en el cuadro de diálogo **Delete Cluster**. Si no libera la EIP, se facturará según las reglas de precios de la EIP de VPC.

10.7 ¿Cómo puedo detener la facturación de GaussDB(DWS)?

A continuación se describe cómo detener la facturación en los modos de pago por uso y de paquete de descuento.

- **Pago por uso**

Si ya no usa un clúster de GaussDB(DWS) que está en modo de pago por uso, elimínelo y sus recursos para detener la facturación. Para ello, inicie sesión en la consola de gestión de GaussDB(DWS) y vaya a la página de **Clusters**. La siguiente tabla muestra los elementos facturables de GaussDB(DWS).

Tabla 10-1 Conceptos facturables de GaussDB(DWS)

Concepto de facturación	Descripción
Nodo de almacén de datos	La facturación se detendrá después de eliminar el clúster. Para obtener más información acerca de cómo eliminar un clúster, consulte Eliminación de un clúster .

Concepto de facturación	Descripción
Espacio de almacenamiento de instantáneas	<p>Cuando se elimina un clúster, se eliminan las instantáneas automatizadas, pero se conservan las instantáneas manuales. Después de eliminar las instantáneas manuales, se detiene la facturación. Para eliminar las instantáneas manuales, inicie sesión en la consola de gestión de GaussDB(DWS) y vaya a la página de Clusters.</p> <p>Si las instantáneas manuales se conservan después de eliminar el clúster y el tamaño total de la instantánea supera el espacio libre, la parte sobrante se facturará según la tarifa de facturación de OBS.</p>
(Opcional) EIP y ancho de banda	<p>Si un clúster tiene una EIP, seleccione Release the EIP bound to the cluster. al eliminar el clúster. La facturación se detendrá después de que se publique la EIP. Si no libera la EIP, se facturará de acuerdo con las reglas de precios de la EIP de VPC.</p>
(Opcional) DEK	<p>Si ha habilitado la función Encrypt DataStore para un clúster y compró una clave en DEW, la clave no se elimina después de eliminar el clúster. Cancelar la suscripción y eliminar manualmente la clave para detener la facturación. Para ello, inicie sesión en la consola de gestión de DEW y elija Data Encryption Workshop > Key Pair Service.</p>

- **Paquetes de descuento:**

Los paquetes de descuento son un pago único, por lo que la facturación se detiene cuando el servicio expira. Para obtener más información sobre cómo darse de baja del paquete por adelantado, consulte la sección [¿Se admite el reembolso?](#). Si no se da de baja del paquete, el sistema no reembolsará las tarifas.

Además, debe eliminar el clúster correspondiente para evitar que se le cobren cargos de pago por uso. Para obtener más información, consulte [Modo de pago por uso](#).

10.8 ¿Se detiene la facturación de pago por uso cuando se detiene mi clúster?

No. El sistema liquida las tarifas por hora para el modo de pago por uso. Puede utilizar el servicio siempre y cuando el saldo de su cuenta sea suficiente. Para reducir costes:

- Elimine los clústeres comprados si no los necesita y cree nuevos cuando sea necesario.
- Cambie al modo de paquete de descuento que le permite utilizar el servicio dentro del período de tiempo específico sin cargos adicionales.

10.9 ¿Cómo puedo reutilizar un paquete de descuento después de que mi paquete caduque?

1. Inicie sesión en la consola de gestión de GaussDB(DWS) para volver a suscribirse a un paquete de descuento.

Si el clúster cambia a facturación de pago por uso, después de volver a comprar los nodos de paquete de descuento, el sistema de facturación cambia al modo de paquete de descuento para la región, la variante, el proyecto de empresa y la cantidad correspondientes.

2. Si la renovación automática está activada, tu cuenta volverá a suscribirse a un paquete de descuento cuando expire el paquete actual.

Para obtener más información, véase [¿Cómo puedo renovar el servicio?](#).

3. Si la compra pone su cuenta en mora, no se puede crear el clúster y el sistema muestra un mensaje que indica que el cálculo está en curso.

10.10 ¿Por qué el botón de compra no está disponible cuando creo un clúster?

Las posibles causas del botón de compra no disponible durante la creación del clúster son las siguientes:

- La variante a comprar podría estar agotada, o la región no tiene tal variante.

Solución: Antes de comprar, compruebe si la variante está disponible en la región o compre un paquete después de crear el clúster. Después del paquete, el paquete se asocia automáticamente con el clúster.

- La cuenta podría estar en mora o estar restringida, por lo que no se pueden crear nuevos recursos.

Solución: Si su saldo es insuficiente, recargue la cuenta para cancelar la cantidad vencida.

10.11 Si no uso el paquete de un año que compré y elimino todos los grupos, ¿se deducirá la cuota del paquete?

Sí.

Un paquete de descuento entra en vigor a partir de la fecha de compra y caduca automáticamente cuando finaliza el período de suscripción. Este período se fija incluso si no ejecuta un clúster de GaussDB(DWS). Para ahorrar costos, cree un clúster inmediatamente después de comprar un paquete de descuento o compre el paquete después de crear un clúster.

El paquete caduca al final de la duración de uso, independientemente de la creación del clúster. Si el paquete caduca pero no se elimina el clúster, el modo de facturación cambiará a pago por uso.

10.12 ¿Se solaparán los períodos de validez de los paquetes de GaussDB(DWS) cuando compre otro paquete antes de que expire el paquete actual?

Sí. Su nuevo paquete es efectivo justo después de que usted pague por él, independientemente de la expiración actual del paquete.

Recomendaciones:

- Seleccione **Renew** en la página **Renewals** para que el paquete actual amplíe el período de validez original.
- O espere hasta que su paquete actual caduque para comprar uno nuevo. Su modo de facturación cambia a pago por uso en el ínterin.

10.13 ¿Puedo crear varios clústeres cuando compre un paquete que tenga más de seis nodos?

Sí. Cada clúster necesita al menos tres nodos, por lo que puede crear uno más si tiene más de tres nodos restantes.

Por ejemplo, un paquete que contiene nueve nodos le permite crear un clúster con tres nodos y un segundo con seis nodos.

10.14 ¿Cómo descongelar un clúster?

Análisis de las causas

Si el saldo de su cuenta es insuficiente y la deducción de la tarifa falla, el período de retención comienza. Durante el período de retención, los recursos de servicio se congelarán y no se podrán usar, pero los recursos y los datos se reservan.

Procedimiento de manejo

Para descongelar un clúster, debe recargar su cuenta para asegurarse de que el saldo de la cuenta no sea **0**. Para obtener más información, consulte [¿Cómo puedo renovar el servicio?](#) Después de descongelar un clúster, el estado del clúster cambia a **Available**.

Si ha comprado un paquete de descuento, el sistema cambia a la facturación de pago por uso después de que el paquete expire. En este caso, si el clúster está congelado debido a una deducción de tarifas fallida, solo puede descongelar el clúster recargando su cuenta. A continuación, puede renovar el servicio o comprar un nuevo paquete de descuento. Un clúster no se puede descongelar renovando su paquete de descuento o comprando un nuevo paquete.

10.15 ¿Puedo congelar o cerrar un clúster de GaussDB(DWS) para detener la facturación?

No. Si no necesita un clúster, elimínelo y sus recursos.

10.16 ¿Se pueden asociar automáticamente los nodos de descuento adquiridos después de la creación del clúster con los nodos del clúster antiguo?

Sí. Una vez creado el clúster de pago por uso, si ha adquirido un paquete de descuento, los nodos del paquete se asociarán automáticamente con los nodos correspondientes del clúster creado.

Por ejemplo, si crea un clúster de tres nodos y compra un paquete de descuento para un solo nodo, los otros dos nodos se cobran en modo de pago por uso. Si luego adquiere un paquete de descuento para dos nodos, el sistema de facturación se asocia automáticamente con los nodos del clúster que se ha utilizado.

10.17 ¿Cómo sé cuándo caduca mi paquete de descuento GaussDB(DWS)?

Paso 1 Inicie sesión en la consola de gestión de GaussDB(DWS).

Paso 2 En la esquina superior derecha, selecciona **Billing Center > My Orders**.

Paso 3 Busque el pedido de su paquete en la lista y haga clic en **Details** en la columna **Operation**.

Paso 4 En la página mostrada, vea la fecha de inicio y la fecha de finalización del pedido.

NOTA

La facturación de pago por uso no tiene fecha de vencimiento. Si el saldo de la cuenta es insuficiente, renueve el servicio de manera oportuna para evitar que el clúster se vea afectado. Para obtener más información, véase [¿Cómo puedo renovar el servicio?](#)

---Fin