

openGauss
3.0.0

特性描述

文档版本	01
发布日期	2022-03-31



版权所有 © 华为技术有限公司 2022。保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

商标声明



HUAWEI和其他华为商标均为华为技术有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受华为公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为公司对本文档内容不做任何明示或暗示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

华为技术有限公司

地址： 深圳市龙岗区坂田华为总部办公楼 邮编： 518129

网址： <https://www.huawei.com>

客户服务邮箱： support@huawei.com

客户服务电话： 4008302118

目录

1 高性能	1
1.1 CBO 优化器	1
1.2 支持 LLVM	2
1.3 向量化引擎	3
1.4 行列混合存储	4
1.5 自适应压缩	5
1.6 SQL by pass	7
1.7 鲲鹏 NUMA 架构优化	7
1.8 支持线程池高并发	8
1.9 SMP 并行执行	9
1.10 Xlog no Lock Flush	10
1.11 Parallel Page-based Redo For Ustore	10
1.12 行存转向量化	11
2 高可用	13
2.1 主备机	13
2.2 逻辑复制	14
2.3 在线节点替换	15
2.4 逻辑备份	16
2.5 物理备份	17
2.6 作业失败自动重试	18
2.7 极致 RTO	20
2.8 级联备机	21
2.9 延时回放	22
2.10 备机增加删除	23
2.11 延迟进入最大可用模式	24
2.12 并行逻辑解码	25
2.13 DCF	27
2.14 CM	28
2.15 支持 Global SysCache	28
2.16 支持备机 build 备机	30
3 维护性	31
3.1 灰度升级	31

3.2 支持 WDR 诊断报告.....	32
3.3 慢 SQL 诊断.....	34
3.4 Session 性能诊断.....	37
3.5 系统 KPI 辅助诊断.....	39
4 数据库安全.....	41
4.1 访问控制模型.....	41
4.2 控制权和访问权分离.....	42
4.3 数据库认证机制.....	43
4.4 数据加密存储.....	44
4.5 数据库审计.....	45
4.6 网络通信安全.....	45
4.7 资源标签机制.....	46
4.8 统一审计机制.....	47
4.9 动态数据脱敏机制.....	49
4.10 行级访问控制.....	52
4.11 用户口令强度校验机制.....	53
4.12 全密态数据库等值查询.....	55
4.13 账本数据库机制.....	57
4.14 透明数据加密.....	58
5 企业级特性.....	61
5.1 函数及存储过程支持.....	61
5.2 支持 SQL hint.....	62
5.3 全文索引.....	63
5.4 Copy 接口支持容错机制.....	64
5.5 分区.....	65
5.6 高级分析函数支持.....	66
5.7 物化视图.....	67
5.8 支持 HyperLogLog.....	68
5.9 在线添加索引.....	68
5.10 自治事务.....	69
5.11 全局临时表.....	70
5.12 伪列 ROWNUM.....	71
5.13 支持存储过程调试.....	72
5.14 JDBC 客户端负载均衡与读写分离.....	73
5.15 In-place Update 存储引擎.....	73
5.16 发布订阅.....	74
5.17 外键锁增强.....	75
5.18 支持 OLTP 场景数据压缩.....	76
6 应用开发接口.....	77
6.1 支持标准 SQL.....	77
6.2 支持标准开发接口.....	78

6.3 PG 接口兼容.....	78
6.4 支持 PL/Java.....	79
7 AI 能力.....	81
7.1 AI4DB: 数据库自治运维.....	81
7.1.1 数据库指标采集、预测与异常监控.....	82
7.1.2 慢 SQL 根因分析.....	82
7.1.3 索引推荐.....	83
7.1.4 参数调优与诊断.....	84
7.1.5 慢 SQL 发现.....	85
7.2 DB4AI: 数据库驱动 AI.....	86
7.3 AI in DB: 数据库内 AI 功能.....	87
7.3.1 Predictor: AI 查询时间预测.....	87
8 工具链.....	89
8.1 MySQL->openGauss 迁移工具 chameleon.....	89
8.2 openGauss 客户端工具 DataStudio.....	90
9 中间件.....	92
9.1 分布式数据库能力.....	92
9.2 使用 kubernetes 部署分布式数据库.....	93

1 高性能

- 1.1 CBO优化器
- 1.2 支持LLVM
- 1.3 向量化引擎
- 1.4 行列混合存储
- 1.5 自适应压缩
- 1.6 SQL by pass
- 1.7 鲲鹏NUMA架构优化
- 1.8 支持线程池高并发
- 1.9 SMP并行执行
- 1.10 Xlog no Lock Flush
- 1.11 Parallel Page-based Redo For Ustore
- 1.12 行存转向量化

1.1 CBO 优化器

可获得性

本特性自openGauss 1.0.0版本开始引入。

特性简介

openGauss优化器是基于代价的优化 (Cost-Based Optimization，简称CBO)。

客户价值

openGauss CBO优化器能够在众多计划中依据代价选出最高效的执行计划，最大限度的满足客户业务要求。

特性描述

在CBO优化器模型下，数据库根据表的元组数、字段宽度、NULL记录比率、distinct值、MCV值、HB值等表的特征值，以及一定的代价计算模型，计算出每一个执行步骤的不同执行方式的输出元组数和执行代价(cost)，进而选出整体执行代价最小/首元组返回代价最小的执行方式进行执行。

特性增强

无。

特性约束

无。

依赖关系

无。

1.2 支持 LLVM

可获得性

本特性自openGauss 1.0.0 版本开始引入。

特性简介

openGauss的LLVM (Low Level Virtual Machine) 技术，提供了查询动态编译执行的能力。

客户价值

通过对查询动态编译执行，大幅提高重查询性能。

特性描述

openGauss借助LLVM提供的库函数，依据查询执行计划树，将原本在执行器阶段才会确定查询实际执行路径的过程提前到执行初始化阶段，从而规避原本查询执行时候伴随的函数调用、逻辑条件分支判断以及大量的数据读取等问题，以达到提升查询性能的目的。

特性增强

无。

特性约束

无。

依赖关系

依赖于LLVM开源组件，当前使用的开源版本为10.0.0。

1.3 向量化引擎

可获得性

本特性自openGauss 1.0.0版本开始引入。

特性简介

openGauss提供向量化引擎，通常用在OLAP数据仓库类系统，因为分析型系统通常都是数据处理密集型负载，基本上都是采用顺序方式来访问表中大部分的数据，然后进行计算，最后将计算结果输出给终端用户。

客户价值

通过批量计算，大幅提高复杂类查询性能。

特性描述

传统的数据库查询执行都是采用一次一tuple的pipeline执行模式。这样CPU的大部分处理时不是用来真正的处理数据，而是在遍历查询操作树，这样CPU的有效利用率不高。同时这也会导致低指令缓存性能和频繁跳转。更加糟糕的是，这种方式的执行，不能够利用到现在新硬件的新的能力来加速查询的执行。在执行引擎中，另外一个解决方案就是改变一次一tuple 为一次一列的模式。这也是我们向量化执行引擎的一个基础。

向量化引擎是跟列存储技术绑定的，因为列存每列的数据存储在一起，可以认为这些数据是以数组的方式存储的。基于这样的特征，当该列数据需要进行某一同样操作，可以通过一个循环来高效完成对这个数据块各个值的计算。

向量化执行引擎的优势在于：

- 可以减少节点间的调度，提高CPU的利用率。
- 因为相同类型的一类数据放在一起，可以更容易的利用硬件与编译的新优化特征。

特性增强

无。

特性约束

无。

依赖关系

依赖于列式存储。

1.4 行列混合存储

可获得性

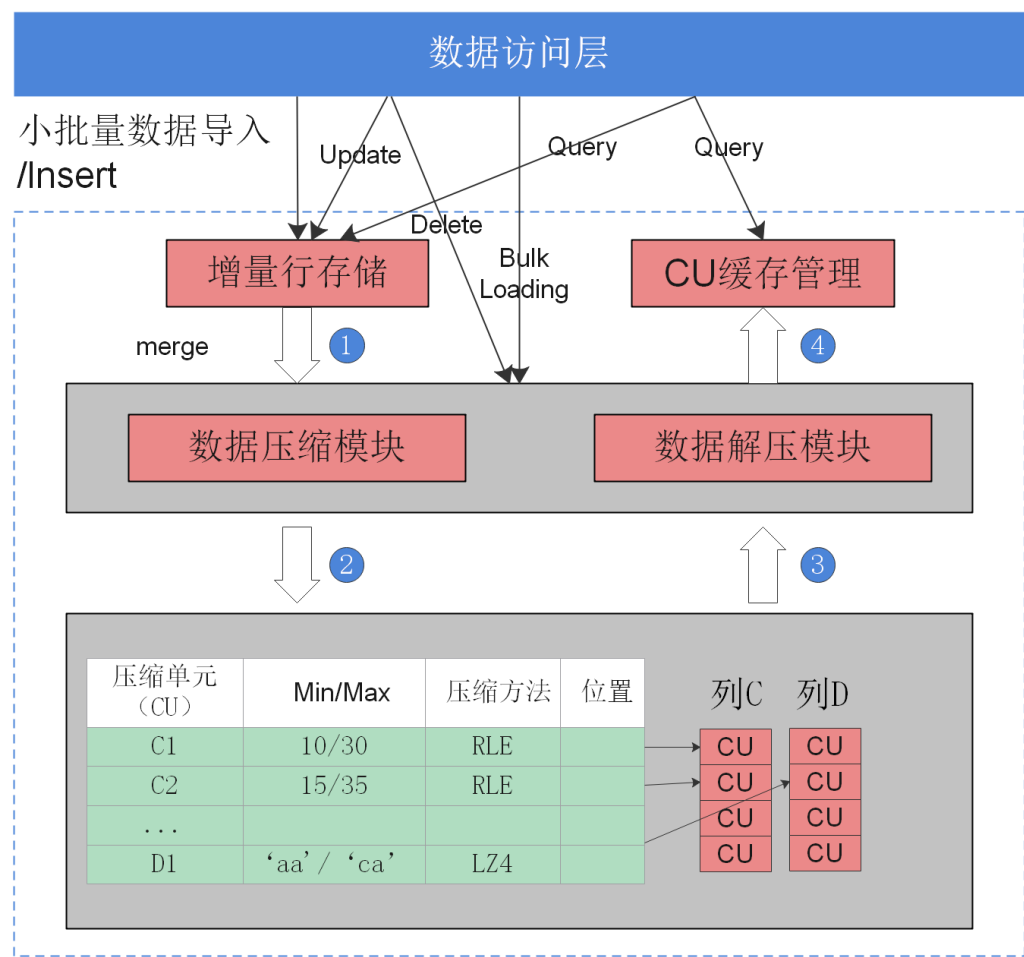
本特性自openGauss 1.0.0 版本开始引入。

特性简介

openGauss 支持行存储和列存储两种存储模型，用户可以根据应用场景，建表的时候选择行存储还是列存储表。

一般情况下，如果表的字段比较多（大宽表），查询中涉及到的列不很多的情况下，适合列存储。列存储方式如图1-1所示。如果表的字段个数比较少，查询大部分字段，那么选择行存储比较好。

图 1-1 列存储示意图



客户价值

在大宽表、数据量比较大的场景中，查询经常关注某些列、行存储引擎查询性能比较差。例如气象局的场景，单表有200~800个列，查询经常访问10个列，在类似这样的场景下，向量化执行技术和列存储引擎可以极大的提升性能和减少存储空间。

特性描述

表有行存表和列存表两种存储模型。两种存储模型各有优劣，建议根据实际情况选择。

- 行存表
默认创建表的类型。数据按行进行存储，即一行数据紧挨着存储。行存表支持完整的增删改查。适用于对数据需要经常更新的场景。
- 列存表
数据按列进行存储，即一列所有数据紧挨着存储。单列查询IO小，比行存表占用更少的存储空间。适合数据批量插入、更新较少和以查询为主统计分析类的场景。列存表不适合点查询，insert插入单条记录性能差。
- 行存表和列存表的选择
 - 更新频繁程度
数据如果频繁更新，选择行存表。
 - 插入频繁程度
频繁的少量插入，选择行存表。一次插入大批量数据，选择列存表。
 - 表的列数
表的列数很多，选择列存表。
 - 查询的列数
如果每次查询时，只涉及了表的少数（<50%总列数）几个列，选择列存表。
 - 压缩率
列存表比行存表压缩率高。但高压压缩率会消耗更多的CPU资源。

特性增强

无。

特性约束

无。

依赖关系

无。

1.5 自适应压缩

可获得性

本特性自openGauss 1.0.0 版本开始引入。

特性简介

数据压缩是当前数据库采用的主要技术。数据类型不同，适用于它的压缩算法不同。对于相同类型的数据，其数据特征不同，采用不同的压缩算法达到的效果也不相同。自适应压缩正是从数据类型和数据特征出发，采用相应的压缩算法，实现了良好的压缩比、快速的入库性能以及良好的查询性能。

客户价值

数据入库和频繁的海量数据查询是用户的主要应用场景。在数据入库场景中，自适应压缩可以大幅度地减少数据量，成倍提高IO操作效率，将数据簇集存储，从而获得快速的入库性能。当用户进行数据查询时，少量的IO操作和快速的数据解压可以加快数据获取的速率，从而在更短的时间内得到查询结果。

特性描述

目前，数据库已实现了列存储上RLE、DELTA、BYTEPACK/BITPACK、LZ4、ZLIB、LOCAL DICTIONARY等多种压缩算法。数据库支持的数据类型与压缩算法的映射关系如下表所示。

-	RLE	DELTA	BITPACK/ BYTEPACK	LZ4	ZLIB	LOCAL DICTIONARY
Smallint/Int/Bigint/Oid Decimal/Real/Double Money/Time/Date/ Timestamp	✓	✓	✓	✓	✓	-
Tinterval/Interval/Time with time zone/	-	-	-	-	✓	-
Numeric/Char/Varchar/ Text/Nvarchar2 以及其他支持数据类型	✓	✓	✓	✓	✓	✓

特性增强

支持对压缩算法进行不同压缩水平的调整。

特性约束

无。

依赖关系

开源压缩软件LZ4/ZLIB。

1.6 SQL by pass

可获得性

本特性自openGauss 1.0.0版本开始引入。

特性简介

通过对TP场景典型查询的定制化执行方案来提高查询性能。

客户价值

提升TP类查询的性能。

特性描述

在典型的OLTP场景中，简单查询占了很大一部分比例。这种查询的特征是只涉及单表和简单表达式的查询，因此为了加速这类查询，提出了SQL-BY-PASS框架，在parse层对这类查询做简单的模式判别后，进入到特殊的执行路径里，跳过经典的执行器执行框架，包括算子的初始化与执行、表达式与投影等经典框架，直接重写一套简洁的执行路径，并且直接调用存储接口，这样可以大大加速简单查询的执行速度。

特性增强

无。

特性约束

无。

依赖关系

无。

1.7 鲲鹏 NUMA 架构优化

可获得性

本特性自openGauss 1.0.0版本开始引入。

特性简介

鲲鹏NUMA架构优化，主要面向鲲鹏处理器架构特点、ARMv8指令集等，进行相应的系统优化，涉及到操作系统、软件架构、锁并发、日志、原子操作、Cache访问等一系列的多层次优化，从而大幅提升了openGauss数据库在鲲鹏平台上的处理性能。

客户价值

数据库的处理性能，如每分钟处理交易量（Transaction Per Minute），是数据库竞争力的关键性能指标，在同等硬件成本的条件下，数据库能提供的处理性能越高，那么就可以提供给用户更多的业务处理能力，从而降低客户的使用成本。

特性描述

- openGauss根据鲲鹏处理机的多核NUMA架构特点，进行一系列针对性NUMA架构相关优化。一方面尽量减少跨核内存访问的时延问题，另一方面充分发挥鲲鹏多核算力优势，所提供的关键技术包括重做日志批插、热点数据NUMA分布、CLog分区等，大幅提升TP系统的处理性能。
- openGauss基于鲲鹏芯片所使用的ARMv8.1架构，利用LSE扩展指令集实现高效的原子操作，有效提升CPU利用率，从而提升多线程间同步性能、XLog写入性能等。
- openGauss基于鲲鹏芯片提供的更宽的L3缓存cacheline，针对热点数据访问进行优化，有效提高缓存访问命中率，降低Cache缓存一致性维护开销，大幅提升系统整体的数据访问性能。
- 鲲鹏920，2P服务器（64cores*2，内存768 GB），网络10 GE，IO为4块NVME PCIE SSD时，TPCC为1000warehouse，性能是150万 tpmC。

特性增强

- 支持重做日志批插，分区CLog，提升鲲鹏平台下的数据库处理性能。
- 支持LSE扩展指令集的原子操作，提升多线程同步性能。

特性约束

无。

依赖关系

无。

1.8 支持线程池高并发

可获得性

本特性自openGauss 1.0.0版本开始引入。

特性简介

通过线程池化技术来支撑数据库大并发稳定运行。

客户价值

支撑客户大并发下，系统整体吞吐平稳。

特性描述

线程池技术的整体设计思想是线程资源池化、并且在不同连接之间复用。系统在启动之后会根据当前核数或者用户配置启动固定一批数量的工作线程，一个工作线程会服

务一到多个连接session，这样把session和thread进行了解耦。因为工作线程数是固定的，因此在高并发下不会导致线程的频繁切换，而由数据库层来进行session的调度管理。

特性增强

openGauss 1.0.0版本引入该特性。

openGauss 1.1.0版本实现了线程池的动态扩缩容。

特性约束

无。

依赖关系

无。

1.9 SMP 并行执行

可获得性

本特性自openGauss 1.1.0版本开始引入。

特性简介

openGauss的SMP并行技术是一种利用计算机多核CPU架构来实现多线程并行计算，以充分利用CPU资源来提高查询性能的技术。

客户价值

SMP并行技术充分利用了系统多核的能力，来提高重查询的性能。

特性描述

在复杂查询场景中，单个查询的执行较长，系统并发度低，通过SMP并行执行技术实现算子级的并行，能够有效减少查询执行时间，提升查询性能及资源利用率。SMP并行技术的整体实现思想是对于能够并行的查询算子，将数据分片，启动若干个工作线程分别计算，最后将结果汇总，返回前端。SMP并行执行增加数据交互算子（Stream），实现多个工作线程之间的数据交互，确保查询的正确性，完成整体的查询。

特性增强

无。

特性约束

- 索引扫描不支持并行执行。
- MergeJoin不支持并行执行。
- WindowAgg order by不支持并行执行。

- cursor不支持并行执行。
- 存储过程和函数内的查询不支持并行执行。
- 不支持子查询subplan和initplan的并行，以及包含子查询的算子的并行。
- 查询语句中带有median操作的查询不支持并行执行。
- 带全局临时表的查询不支持并行执行。
- 物化视图的更新不支持并行执行。

依赖关系

无。

1.10 Xlog no Lock Flush

可获得性

本特性自openGauss 2.0.0 版本开始引入。

特性简介

取消WalInsertLock争抢及WalWriter专用磁盘写入线程。

客户价值

在保持原有XLog功能不变的基础上，进一步提升系统性能。

特性描述

对WalInsertLock进行优化，利用LSN（Log Sequence Number）及LRC（Log Record Count）记录了每个backend的拷贝进度，取消WalInsertLock机制。在backend将日志拷贝至WalBuffer时，不用对WalInsertLock进行争抢，可直接进行日志拷贝操作。并利用专用的WalWriter写日志线程，不需要backend线程自身来保证xlog的Flush。

特性增强

无。

特性约束

无。

依赖关系

无。

1.11 Parallel Page-based Redo For Ustore

可获得性

本特性自openGauss 2.1.0 版本开始引入。

特性简介

优化Ustore Inplace Update WAL log写入，Ustore DML Operation回放提高并行度。

客户价值

对于Update的WAL使用空间减少,Ustore DML Operation回放提高并行度。

特性描述

通过利用Prefix和suffix来减少update WAL log的写入，通过把回放线程分多个类型来解决Ustore DML WAL大多都是多页面回放问题；同时把Ustore的数据页面回放按照blkno去分发。

特性增强

无。

特性约束

无。

依赖关系

依赖于Ustore引擎。

1.12 行存转向量化

可获得性

本特性自openGauss 3.0.0版本开始引入。

特性简介

将行存表的查询转换为向量化执行计划执行，提升复杂查询的执行性能。

客户价值

由于行存执行引擎在执行包含较多表达式或者关联操作的复杂查询时，性能表现不佳；而向量化执行引擎在执行复杂查询时具有优异的性能表现。所以通过将行存表的查询转换为向量化执行计划执行，能够有效提升复杂查询的查询性能。

特性描述

本特性通过对扫描算子增加一层RowToVec的操作，将行存表的数据在内存中变为向量化格式后，上层算子都能够转化为对应的向量化算子，从而使用向量化执行引擎计算。支持行转列的扫描算子包括：SeqScan、IndexOnlyscan、IndexScan、BitmapScan、FunctionScan、ValueScan、TidScan。

特性增强

无。

特性约束

- 不支持向量化的场景包括：
 - targetList存在返回set的函数。
 - targetList或者qual中存在不支持向量化的表达式：数组类表达式计算；多子查询表达式计算；Field类表达式计算；系统表列。
 - 包含不支持向量化的类型：POINTOID；LSEG OID；BOXOID；LINEOID；CIRCLEOID；POLYGONOID；PATHOID；用户自定义类型。
- MOT表不支持转向量化。

依赖关系

向量化引擎

2 高可用

- 2.1 主备机
- 2.2 逻辑复制
- 2.3 在线节点替换
- 2.4 逻辑备份
- 2.5 物理备份
- 2.6 作业失败自动重试
- 2.7 极致RTO
- 2.8 级联备机
- 2.9 延时回放
- 2.10 备机增加删除
- 2.11 延迟进入最大可用模式
- 2.12 并行逻辑解码
- 2.13 DCF
- 2.14 CM
- 2.15 支持Global SysCache
- 2.16 支持备机build备机

2.1 主备机

可获得性

本特性自openGauss 1.0.0 版本开始支持DN主备。

特性简介

为了保证故障的可恢复，需要将数据写多份，设置主备多个副本，通过日志进行数据同步，可以实现节点故障、停止后重启等情况下，openGauss能够保证故障之前的数据无丢失，满足ACID特性。

客户价值

主备机功能可以支持主机故障时切换到备机，数据不丢失，业务可以快速恢复。

特性描述

主备环境支持一主多备模式。在一主多备模式下，所有的备机都需要重做日志，都可以升主。一主多备提供更高的容灾能力，更加适合于大批量事务处理的OLTP系统。

主备之间可以通过switchover进行角色切换，主机故障后可以通过failover对备机进行升主。

初始化安装或者备份恢复等场景中，需要根据主机重建备机的数据，此时需要build功能，将主机的数据和WAL日志发送到备机。主机故障后重新以备机的角色加入时，也需要build功能将其数据和日志与新主拉齐。另外，在在线扩容的场景中，需要通过build来同步元数据到新节点上的实例。Build包含全量build和增量build，全量build要全部依赖主机数据进行重建，拷贝的数据量比较大，耗时比较长，而增量build只拷贝差异文件，拷贝的数据量比较小，耗时比较短。一般情况下，优先选择增量build来进行故障恢复，如果增量build失败，再继续执行全量build，直至故障恢复。

为了实现所有实例的高可用容灾能力，除了以上对DN设置主备多个副本，openGauss还提供了其他一些主备容灾能力，比如CM Sever（一主多备）以及ETCD（一主多备）等，使得实例故障后可以尽可能快地恢复，不中断业务，将因为硬件、软件和人为造成的故障对业务的影响降到最低，以保证业务的连续性。

特性增强

无。

特性约束

无。

依赖关系

无。

2.2 逻辑复制

可获得性

本特性自openGauss 1.0.0版本开始引入。

特性简介

openGauss 提供逻辑解码功能，将物理日志反解析为逻辑日志。通过DRS等逻辑复制工具将逻辑日志转化为SQL语句，到对端数据库回放，达到异构数据库同步数据的功能。

能。目前支持openGauss数据库与MySQL数据库、Oracle数据库之间的单向、双向逻辑复制。

客户价值

逻辑复制可以为数据库数据实时迁移、双库双活、支持滚动升级提供解决方案。

特性描述

DN通过物理日志反解析为逻辑日志，DRS等逻辑复制工具从DN抽取逻辑日志转换为sql语句，到对端数据库（MySQL）回放。逻辑复制工具同时从MySQL数据库抽取逻辑日志，反解析为SQL语句之后回放到openGauss，达到异构数据库同步数据的目的。

特性增强

- openGauss1.0.0逻辑解码新增全量+增量抽取日志的方案。
- openGauss1.1.0逻辑解码新增备机支持逻辑解码。

特性约束

不支持列存复制，不支持DDL复制。

依赖关系

依赖于逻辑复制工具对逻辑日志进行解码。

2.3 在线节点替换

可获得性

本特性自openGauss 1.0.0版本开始引入。

特性简介

数据库内某节点出现硬件故障造成节点不可用或者实例状态不正常，当数据库没有加锁，通过节点替换或修复故障实例来恢复数据库的过程中，支持用户DML操作，有限场景支持用户DDL操作。

客户价值

随着企业数据规模不断增大，节点数量急剧增加，硬件损坏概率相应增加，物理节点替换修复成为日常运维工作的常态。传统的离线节点替换方式无法满足客户业务不中断需求，日常运维操作中，经常的业务中断将给客户带来重大损失。而目前业界数据库产品在节点替换的过程中，或者需要中断业务，或者只允许部分操作，均不能满足大规模数据情况下，常态物理节点替换的需求。

特性描述

数据库内某节点出现硬件故障造成节点不可用或者实例状态不正常，通过节点替换或修复故障实例来恢复数据库的过程中，支持用户DML操作，有限场景支持用户DDL操作。

特性增强

无。

特性约束

目前节点替换已支持用户业务在线DDL：

- 在节点替换窗口期内，支持用户DML操作，有限场景支持用户DDL操作。

依赖关系

无。

2.4 逻辑备份

可获得性

本特性自openGauss1.0.0版本开始引入。

特性简介

支持将数据库内用户表的数据以通用格式备份到指定的存储介质中。

客户价值

通过逻辑备份特性，可以达成以下目的：

- 将用户数据备份到可靠性更高的存储介质中，提升用户表数据的可靠性。
- 通过采用通用的数据格式，支持跨版本恢复和异构恢复。
- 可以用于冷数据的归档。

特性描述

openGauss 提供逻辑备份能力，可以将用户表的数据以通用的text或者csv格式备份到本地磁盘文件中，并在同构/异构数据库中恢复该用户表的数据。

特性增强

无。

特性约束

逻辑备份的约束条件请参见《工具参考》中“服务端工具 > gs_dump”章节。

依赖关系

无。

2.5 物理备份

可获得性

本特性自openGauss 1.0.0 版本开始引入。

特性简介

支持将整个数据库的数据以内部格式备份到指定的存储介质中。。

客户价值

通过物理备份特性，可以达成以下目的：

- 整个数据库的数据备份到可靠性更高的存储介质中，提升系统整体的可靠性。
- 通过采用数据库内部的数据格式，极大提升备份恢复性能。
- 可以用于冷数据的归档。

典型的物理备份策略和应用场景如下：

- 周一，执行数据库全量备份
- 周二，以周一全量备份为基准点，执行增量备份
- 周三，以周二增量备份为基准点，执行增量备份
- ...
- 周日，以周六增量备份为基准点，执行增量备份

上述备份策略以一个星期为周期。

特性描述

openGauss 1.1.0 提供物理备份能力，可以将整个数据库的数据以数据库内部格式备份到本地磁盘文件、OBS对象、NBU对象或爱数对象中，并在同构数据库中恢复整个数据库的数据。在基础之上，还提供压缩、流控、断点续备等高阶功能。

物理备份主要分为全量备份和增量备份，区别如下：全量备份包含备份时刻点上数据库的全量数据，耗时时间长（和数据库数据总量成正比），自身即可恢复出完整的数据库；增量备份只包含从指定时刻点之后的增量修改数据，耗时时间短（和增量数据成正比，和数据总量无关），但是必须要和全量备份数据一起才能恢复出完整的数据库。

特性增强

支持全量备份和增量备份同时执行。

特性约束

物理备份的约束条件请参见《管理员指南》中“备份与恢复 > 物理备份恢复”章节。

依赖关系

无。

2.6 作业失败自动重试

可获得性

本特性自openGauss 1.0.0 版本开始引入。

特性简介

批处理作业中，在由于网络异常、锁冲突等出错情况下能够保证作业自动重试。

客户价值

网络异常、锁冲突等出错场景下无需人工干预，自动对出错查询进行重试，提高了可用性。

特性描述

openGauss 数据库提供作业重试机制：gsql Retry。

- gsql Retry，对于需要进行重试的错误使用唯一的错误码（SQL STATE）进行标识。对客户端工具gsql进行功能增强，使用错误码配置文件（retry_errcodes.conf，置于和gsql同级的安装目录下）对需要重试的错误列表进行配置。gsql提供元命令\set RETRY [number]来开启/关闭错误重试功能，重试次数设为范围为5~10，缺省值为5。打开该功能时gsql会读取上述配置文件，错误重试控制器通过容器记录错误码列表。如果打开后出现配置文件中某一个错误，控制器会将每次缓存的查询语句重新发送给服务端进行重试，直到成功或超过重试次数报错。

特性增强

无。

特性约束

- 功能范围限制：
仅能提高故障发生时SQL语句执行成功率，不能保证100%的执行成功。
- 错误类型约束：
SQL语句出错时能够被识别和重试的错误，仅限在表2-1中定义的错误。

表 2-1 支持的错误类型列表

错误类型	错误码	备注
对端连接重置 (CONNECTION_RESET_BY_PEER)	YY001	TCP通信错误: Connection reset by peer

错误类型	错误码	备注
对端流重置 (STREAM_CONNECTION_RESET_BY_PEER)	YY002	TCP通信错误: Stream connection reset by peer (DN和DN间通信)
锁等待超时 (LOCK_WAIT_TIMEOUT)	YY003	锁超时, Lock wait timeout
连接超时 (CONNECTION_TIMED_OUT)	YY004	TCP通信错误, Connection timed out
查询设置错误 (SET_QUERY_ERROR)	YY005	SET命令发送失败, Set query error
超出逻辑内存 (OUT_OF_LOGICAL_MEMORY)	YY006	内存申请失败, Out of logical memory
通信库内存分配 (SCTP_MEMORY_ALLOC)	YY007	SCTP通信错误, Memory allocate error
无通信库缓存数据 (SCTP_NO_DATA_IN_BUFFER)	YY008	SCTP通信错误, SCTP no data in buffer
通信库释放内存关闭 (SCTP_RELEASE_MEMORY_CLOSE)	YY009	SCTP通信错误, Release memory close
SCTP、TCP断开 (SCTP_TCP_DISCONNECT)	YY010	SCTP、TCP通信错误, SCTP、TCP disconnect
通信库断开 (SCTP_DISCONNECT)	YY011	SCTP通信错误, SCTP disconnect
通信库远程关闭 (SCTP_REMOTE_CLOSE)	YY012	SCTP通信错误, Stream closed by remote
等待未知通信库通信 (SCTP_WAIT_POLL_UNKNOW)	YY013	等待未知通信库通信, SCTP wait poll unknow
无效快照 (SNAPSHOT_INVALID)	YY014	快照非法, Snapshot invalid
通讯接收信息错误 (ERRCODE_CONNECTION_RECEIVE_WRONG)	YY015	连接获取错误, Connection receive wrong
内存耗尽 (OUT_OF_MEMORY)	53200	内存耗尽, Out of memory
连接异常 (CONNECTION_EXCEPTION)	08000	连接出现错误, 和DN的通讯失败, Connection exception
管理员关闭系统 (ADMIN_SHUTDOWN)	57P01	管理员关闭系统, Admin shutdown

错误类型	错误码	备注
关闭远程流接口 (STREAM_REMOTE_CLOSE_SOCKET)	XX003	关闭远程套接字, Stream remote close socket
重复查询编号 (ERRCODE_STREAM_DUPLICATE_QUERY_ID)	XX009	重复查询, Duplicate query id
stream查询并发更新同一行 (ERRCODE_STREAM_CONCURRENT_UPDATE)	YY016	stream查询并发更新同一行, Stream concurrent update

- 语句类型约束：
支持单语句存储过程、函数、匿名块。不支持事务块中的语句。
- 存储过程语句约束：
 - 包含EXCEPTION的存储过程，如果在执行过程中（包含语句块执行和EXCEPTION中的语句执行）错误被抛出，可以retry，如果报错被EXCEPTION捕获则不能retry。
 - 不支持使用全局变量的高级包。
 - 不支持DBE_TASK。
 - 不支持PKG_UTIL文件操作。
- 数据导入约束：
 - 不支持COPY FROM STDIN语句。
 - 不支持gsql \copy from元命令。
 - 不支持JDBC CopyManager copyIn导入数据。

依赖关系

该特性依赖gsql工具端可以正常工作、错误列表配置正确。

2.7 极致 RTO

可获得性

本特性自openGauss 1.1.0 版本开始引入。

特性简介

- 支撑数据库主机重启后快速恢复的场景。
- 支撑主机与同步备机通过日志同步，加速备机回放的场景。

客户价值

当业务压力过大时，备机的回放速度跟不上主机的速度。在系统长时间的运行后，备机上会出现日志累积。当主机故障后，数据恢复需要很长时间，数据库不可用，严重影响系统可用性。

开启极致RTO（Recovery Time Object，恢复时间目标），减少了主机故障后数据的恢复时间，提高了可用性。

特性描述

极致RTO开关开启后，xlog日志回放建立多级流水线，提高并发度，提升日志回放速度。

特性增强

无。

特性约束

极致RTO只关注同步备机的RTO是否满足需求。极致RTO去掉了自带的流控，统一使用recovery_time_target参数来做流控控制。本特性不支持备机读。如果查询备机可能导致备机core。

依赖关系

无。

2.8 级联备机

可获得性

本特性自openGauss 1.1.0 版本开始引入。

特性简介

本特性主要基于当前一主多备的架构，在此基础上，支持级联备机连接备机。

客户价值

一主多备架构在特性业务场景下，无法支持足够灵活的结构。多机房部署，不足以满足HA切换下的完整结构（主次机房三个实例，再次机房两到三个实例）。若备机个数变多，对主机可能导致压力较大。针对业务实时性要求较低的查询，可落到级联备中。因此，我们需要提供级联备能力。

特性描述

主机通过发送日志给备机实现同步，备机通过发送日志给级联备实现同步，其中主备之间日志同步可配置为同步或异步，备机与级联备之间只能以异步方式。

目前一主多备方式架构，主机通过 walsender 线程向备机同步日志，备机通过 walreceiver 线程接受日志，并刷到本地盘，备机读取redo日志，完成主备之间的数据

同步。主备机之间 walsender 与 walreceiver 一一对应。备机与级联备通过walsender与walreceiver线程异步方式同步日志，降低主机流复制的压力。

特性增强

无。

特性约束

- 级联备只能从备机同步数据，不能直接从主机同步。
- 级联备不支持直接从备机build数据，目前只能从主机build数据。如果备机需要全量build，会导致级联备也要重新全量build。
- 级联备为异步同步。
- 级联备不支持promote。
- 级联备不支持notify。
- 目前不提供查询主 - 备 - 级联备集群总体架构，需要根据主找到备，再通过备找到级联备。
- 级联备不能有再次级联备。
- 极致RTO开启时，不支持级联备。

依赖关系

无。

2.9 延时回放

可获得性

本特性自openGauss 2.0.0版本开始引入。

特性简介

支持备机延时回放。

客户价值

默认情况下，备机会尽快恢复来自于主机的XLOG记录。这个功能允许你将备机的回放延时一段指定的时间量。提供这样一份可查询的一段时间之前的数据副本，可以更方便的纠正用户误操作等错误。

特性描述

支持通过配置GUC参数recovery_min_apply_delay设定延时时间，使某个备机延时一段时间回放来自主机的事务日志。

参数取值范围：整型，0~INT_MAX，单位为毫秒。

参数默认值：0（不增加延迟）

特性增强

无。

特性约束

- recovery_min_apply_delay参数主节点设置无效，必须设置在需要延迟的备节点上。
- 延迟时间是根据主服务器上事物提交的时间戳与备机上的当前时间来计算，因此需要保证主备系统时钟一致。
- 没有事务的操作不会被延迟。
- 主备切换之后，原主机若需延迟，需要再手动配置此参数。
- 当synchronous_commit被设置为remote_apply时，同步复制会受到这个延时的影响，每一个COMMIT都需要等待备机回放结束后才会返回。
- 使用这个特性也会让hot_standby_feedback被延迟，这可能导致主服务器的膨胀，两者一起使用时要小心。
- 主机执行了持有AccessExclusive锁的DDL操作，比如DROP和TRUNCATE操作，在备机延迟回放该条记录期间，在备机上对该操作对象执行查询操作会等待锁释放之后才会返回。

依赖关系

无。

2.10 备机增加删除

可获得性

本特性自openGauss 2.0.0版本开始引入。

特性简介

支持新增和删除备节点。

客户价值

当主库的读取压力较大，或者用户想要提升数据库灾难恢复能力，需要新增备节点。当集群中的某些备节点发生故障无法短时间内进行修复时，为保证集群状态正常，用户可以先将故障节点删除。

特性描述

openGauss支持从单机或者一主多备最多扩容至一主八备，支持新增级联备机，支持在集群中存在故障备机的情况下新增备节点；支持从一主多备删除至单节点，支持备机故障的情况下将其删除。

支持在线新增和删除备节点，即可以在不影响主机业务的情况下进行新增和删除备节点。

特性增强

无。

特性约束

新增备机的约束：

- 数据库主机上存在openGauss镜像包。
- 在新增的扩容备机上创建好与主机上相同的用户和用户组。
- 已存在的数据库节点和新增的扩容节点之间需要建立好root用户互信以及数据库管理用户的互信。
- 正确配置xml文件，在已安装数据库配置文件的基础上，添加需要扩容的备机信息。
- 只能使用root用户执行扩容命令。
- 不允许同时在主节点上执行gs_dropnode命令删除其他备机。
- 执行扩容命令前需要导入主机数据库的环境变量。
- 扩容备机的操作系统与主机保持一致。
- 操作过程中不允许同时在其他备节点上执行主备倒换或者故障倒换的操作。

删除备机的约束：

- 删除备节点的操作只能在主节点上执行。
- 操作过程中不允许同时在其他备节点上执行主备倒换或者故障倒换的操作。
- 不允许同时在主节点上执行gs_expansion命令进行扩容。
- 不允许同时执行2次相同的gs_dropnode命令。
- 执行删除操作前，需要确保主节点和备节点之间建立好数据库管理用户的互信。
- 需要使用数据库管理用户执行该命令。
- 执行命令前需要通过source命令导入主机数据库的环境变量。

依赖关系

无。

2.11 延迟进入最大可用模式

可获得性

本特性自openGauss 3.0.0版本开始引入。

特性简介

支持主机延迟进入最大可用模式。

客户价值

当主机与同步备机之间的网络不稳定或其他原因导致主机感知到备机退出，且主机打开最大可用模式配置时，在一定时间窗口内仍维持主机在最大保护模式的状态，超出该时间窗口后，主机进入到最大可用模式。

可以避免因网络抖动、进程闪断等因素导致主机在最大保护模式和最大可用模式之间频繁来回切换。

特性描述

当最大可用模式`most_available_sync`配置为`on`，在主备场景下，当存在同步备发生故障，导致不满足当前所配置的同步备数量（详细可参考`synchronous_standby_names`的含义）时，如果配置了`keep_sync_window`参数，则在`keep_sync_window`设置的时间窗口内，继续保持最大保护模式，即阻塞主机的事务提交，延缓进入最大可用模式的时间。

若在`keep_sync_window`超时窗口内，同步备机故障恢复，且满足当前所配置的同步备数量，则不阻塞事务，恢复到正常状态。

特性增强

无。

特性约束

- 该特性仅在最大可用模式打开的前提下生效。
- 启用该特性可能会对RPO造成影响，若主机在所配置的超时时间窗口内发生故障，主机的事务只在本地提交，未同步到故障的同步备机。
- 该特性不适用于级联备机。

依赖关系

依赖最大可用模式。

2.12 并行逻辑解码

可获得性

本特性自openGauss 3.0.0版本开始引入。

特性简介

支持多线程并行解码。

客户价值

大幅提升逻辑解码性能，解码速度由3~5MBps可提升到标准场景（16核CPU、内存128G、网络带宽 > 200MBps、表的列数为10~100、单行数据量0.1KB~1KB、DML操作以insert为主、不涉及落盘事务即单个事务中语句数量小于4096）下的100MBps。

特性描述

在使用JDBC或pg_recvlogical解码时，设置配置选项parallel-decode-num为大于1且小于等于20的值，开启并行解码特性，使用一个读取线程、多个解码线程以及一个发送线程协同进行逻辑解码操作，显著提升解码速度。

特性增强

无。

特性约束

1. 当前的硬件和网络环境正常；由于逻辑日志一般为xlog的两倍，为保证xlog速度达到100MBps，I/O带宽至少保证200MBps；因为reader、decoder、sender线程均需预留资源，CPU需预留并发数+2的核数，如4并发场景需要预留6核。在实际场景中，使用备机解码即可保证需求，无需进行特殊的资源预留规划。为保证解码性能达标以及尽量降低对业务的影响，一台备机上应尽量仅建立一个并行解码连接，保证CPU、内存、带宽资源充足。
2. 日志级别的guc参数wal_level = logical。
3. guc参数max_replication_slots >= 每个DN所需的（物理流复制槽数+备份槽数+逻辑复制槽数）。
4. 解码配置选项parallel-decode-num > 1且<= 20，指定并行的解码线程数。
5. 不支持DDL语句解码。
6. 不支持列存、数据页复制的解码。
7. 不支持解码分布式事务，当前机制为从DN解码，无法保证分布式事务一致性解码。
8. 单条元组大小不超过1GB，考虑解码结果可能大于插入数据，因此建议单条元组大小不超过500MB。
9. 不支持压缩表的DML语句解码。
10. GaussDB Kernel支持解码的数据类型为：INTEGER、BIGINT、SMALLINT、TINYINT、SERIAL、SMALLSERIAL、BIGSERIAL、FLOAT、DOUBLE PRECISION、DATE、TIME[WITHOUT TIME ZONE]、TIMESTAMP[WITHOUT TIME ZONE]、CHAR(n)、VARCHAR(n)、TEXT。
11. 在需要ssl连接的场景，需要前置条件保证guc参数ssl = on。
12. 不支持interval partition表复制。
13. 在事务中执行DDL语句后，该DDL语句与之后的语句不会被解码。
14. 如需进行备机解码，需在对应主机上设置guc参数enable_slot_log = on。
15. 当前不支持超大CLOB解码。
16. 不允许主备，多个备机同时使用同一个复制槽解码，否则会产生数据不一致。
17. 禁止在使用逻辑复制槽时在其他节点对该复制槽进行操作，删除复制槽的操作需在该复制槽停止解码后执行。

依赖关系

依赖备机解码。

2.13 DCF

可获得性

本特性自openGauss 2.0.0版本开始引入。

特性简介

DCF（Distributed Consensus Framework，分布式共识框架）基于Paxos算法实现数据同步强一致。DCF模式开关开启后，DN可以支持基于Paxos协议的复制与仲裁能力。

客户价值

DN基于Paxos的自选主及日志复制，复制过程中支持压缩及流控，防止带宽占用过高。使用DCF可以提供日志复制、集群高可用等能力。DCF提供了自选主能力，支持少数派强起能力，日志复制支持动态流量调整。同时也提供了基于Paxos多种角色节点类型，并能进行调整。支持查询当前数据库实例的状态。

DCF是一款高性能、高度成熟可靠、易扩展、易使用的独立基础库，其他系统通过接口与DCF简单对接，就能够轻松拥有Paxos算法赋予的强一致、高可用、自动容灾等能力。

特性描述

- DCF进行日志复制时，支持对日志进行压缩后再传输，减小对网络带宽的占用。
- DCF支持SSL，包括TLS1.2和TLS1.3协议标准。当开启SSL时，DN默认将DCF配置为TLS1.2协议标准。
- DCF支持5种TLS1.3的密码套件：TLS13-AES-256-GCM-SHA384、TLS13-CHACHA20-POLY1305-SHA256、TLS13-AES-128-GCM-SHA256、TLS13-AES-128-CCM-8-SHA256、TLS13-AES-128-CCM-SHA256。
- DCF支持passive角色节点类型，passive节点不参与选举，只做日志的同步以及回放，该类型节点在高负载的情况下，日志同步会做流控。
- DCF支持logger角色节点，logger节点可以参与选举和投票，但是只复制DCF的日志，不复制xlog，不进行redo。
- DCF的follower和passive角色可以在线互换，即不中断业务的情况下，follower角色的节点转化为passive角色，passive角色的节点转化为follower。
- DCF支持少数派强起能力，在数据库实例多数派故障的情况下，从正常的备DN中选择少数派模式强启成为主DN，其余正常的备DN从主DN复制日志。
- DCF支持自选主能力，在原主DN故障的场景下，在保证数据一致性的前提下，剩余备DN自动选出新的主DN。

特性增强

无。

特性约束

若使用此功能，在安装部署阶段需要开启DCF开关。在DCF模式下通过多数派选举，安装过程中如果故障节点数加build节点数达到多数派会导致集群安装失败，如在安装一

主两备时，安装过程中一节点因内存不足导致安装失败，另外两节点正常启动，但随后备机会进行一次build，这时build节点加故障节点为2，达到多数派会导致集群安装失败，请在安装过程中检查内存和磁盘等资源是否充足。

依赖关系

无。

2.14 CM

可获得性

本特性自openGauss 3.0.0版本开始引入。

特性简介

CM（Cluster Manager）是一款数据库管理软件，由cm_server和cm_agent组成。

- cm_agent是部署在数据库每个主机上，用来启停和监控各个数据库实例进程的数据库管理组件。
- cm_server是用来进行数据库实例管理和实例仲裁的组件。

客户价值

管理和监控数据库系统中各个功能单元和物理资源的运行情况，确保整个系统的稳定运行。

特性描述

支持自定义资源监控，提供了数据库主备的状态监控、网络通信故障监控、文件系统故障监控、故障自动主备切换等能力。提供了丰富的数据库管理能力，如节点、实例级的启停，数据库实例状态查询、主备切换、日志管理等。

特性增强

无。

特性约束

一主一备模式下，CM只支持基本的安装、启停、检测能力，其他功能不支持。

依赖关系

无。

2.15 支持 Global SysCache

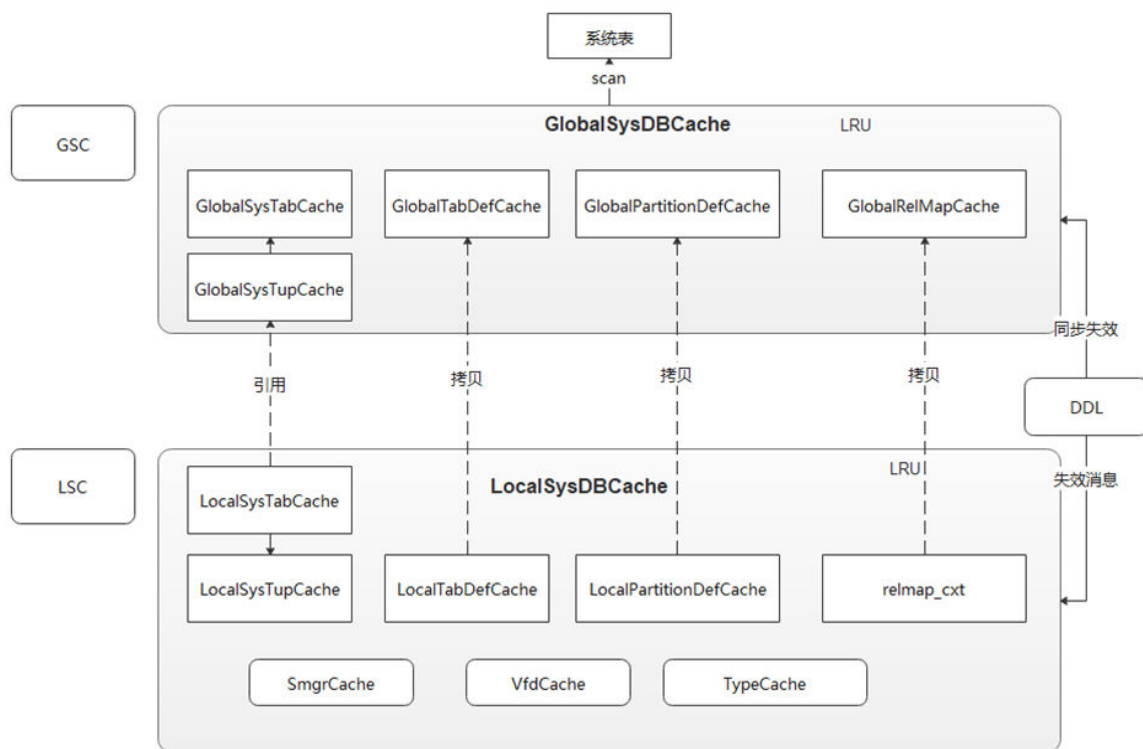
可获得性

本特性自openGauss 3.0.0版本开始引入。

特性简介

全局系统缓存（Global SysCache）是系统表数据的全局缓存和本地缓存。原理如图2-1所示。

图 2-1 Global SysCache 原理图



客户价值

全局系统缓存特性可以降低数据库进程的缓存内存占用，提升数据库的并发扩展能力。

特性描述

全局系统缓存特性指将系统缓存与会话解耦，绑定到线程上，结合线程池特性达到降低内存占用的目的，同时结合全局缓存，提升缓存命中率，保持性能稳定。

特性增强

支持更高的并发查询。

特性约束

- 设置enable_global_syscache为on。建议设置enable_thread_pool参数为on。
- 当DB数较多，且阈值global_syscache_threshold较小时，内存控制无法正常工作，性能会劣化。
- 不支持分布式时序相关的任务，这些任务的内存控制与性能不受GSC特性的影响。

- wal_level设置为minimal或者archive时，备机的查询性能会下降，会退化为短连接。

依赖关系

该特性降内存能力依赖于线程池特性。

2.16 支持备机 build 备机

可获得性

本特性自openGauss 3.0.0版本开始引入。

特性简介

备机build备机加快备机故障的恢复。减小主机io和带宽压力。

客户价值

当业务压力过大时，从主机build备机会对主机的资源造成影响，导致主机性能下降、build变慢的情况。使用备机build备机不会对主机业务造成影响。

特性描述

使用gs_ctl命令可以指定对应的备机去build需要修复的备机。具体操作可参考《工具参考》中的“系统内部使用的工具 > gs_ctl”章节。

特性增强

无。

特性约束

只支持备机build备机，只能使用指定ip和port的方式做build，同时在build前应确保需要修复备机的日志比发送数据的备机的日志落后。

依赖关系

无。

3 维护性

- [3.1 灰度升级](#)
- [3.2 支持WDR诊断报告](#)
- [3.3 慢SQL诊断](#)
- [3.4 Session性能诊断](#)
- [3.5 系统KPI辅助诊断](#)

3.1 灰度升级

可获得性

本特性自openGauss 2.0.0 版本开始引入。

特性简介

灰度升级支持全业务操作，一次性升级所有节点。

客户价值

通过灰度升级，可以提供一种在线升级的方式，保证在不中断业务的情况下，进行所有节点的升级。

特性描述

灰度升级是一种支持升级所有节点的在线升级方式。目前灰度升级只涉及数据库二进制的替换，为了尽可能降低对于业务的影响，采用同一节点两套二进制同时存在的方式，使用软连接切换的方式来进行进程版本的切换升级（闪断一次，10秒以内）。

特性增强

无。

特性约束

灰度升级的约束条件请参见《升级指导书》中“升级前必读> 升级影响和升级约束”章节。

依赖关系

无。

3.2 支持 WDR 诊断报告

可获得性

本特性自openGauss 1.0.0版本开始引入。

特性简介

WDR报告提供数据库性能诊断报告，该报告基于基线性能数据和增量数据两个版本，从性能变化得到性能报告。

客户价值

- WDR报表是长期性能问题最主要的诊断手段。基于SNAPSHOT的性能基线，从多维度做性能分析，能帮助DBA掌握系统负载繁忙程度、各个组件的性能表现及性能瓶颈。
- SNAPSHOT也是后续性能问题自诊断和自优化建议的重要数据来源。

特性描述

WDR(Workload Diagnosis Report)基于两次不同时间点系统的性能快照数据，生成这两个时间点之间的性能表现报表，用于诊断数据库内核的性能故障。

使用generate_wdr_report(...) 可以生成基于两个性能快照的性能报告。

WDR主要依赖两个组件：

- SNAPSHOT性能快照：性能快照可以配置成按一定时间间隔从内核采集一定量的性能数据，持久化在用户表空间。任何一个SNAPSHOT可以作为一个性能基线，其他SNAPSHOT与之比较的结果，可以分析出与基线的性能表现。
- WDR Reporter：报表生成工具基于两个SNAPSHOT，分析系统总体性能表现，并能计算出更多项具体的性能指标在这两个时间段之间的变化量，生成SUMMARY和DETAIL两个不同级别的性能数据。如表3-1、表3-2所示。

表 3-1 SUMMARY 级别诊断报告

诊断类别	描述
Database Stat	主要用于评估当前数据库上的负载，IO状况，负载和IO是衡量TP系统最最重要的特性。 包含当前连接到该数据库的session，提交、回滚的事务数，读取的磁盘块的数量，高速缓存中已经发现的磁盘块的次数，通过数据库查询返回、抓取、插入、更新、删除的行数，冲突、死锁发生的次数，临时文件的使用量，IO读写时间等。
Load Profile	从时间，IO，事务，SQL几个维度评估当前系统负载的表现。 包含作业运行elapsed time、CPU time，事务日质量，逻辑和物理读的量，读写IO次数、大小，登入登出次数，SQL、事务执行量，SQL P80、P95响应时间等。
Instance Efficiency Percentages	用于评估当前系统的缓存的效率。 主要包含数据库缓存命中率。
Events	用于评估当前系统内核关键资源，关键事件的性能。 主要包含数据库内核关键事件的发生次数，事件的等待时间。
Wait Classes	用于评估当前系统关键事件类型的性能。 主要包含数据内核在主要的等待事件种类上的发布：STATUS、LWLOCK_EVENT、LOCK_EVENT、IO_EVENT。
CPU	主要包含CPU在用户态、内核态、Wait IO、空闲状态下的时间发布。
IO Profile	主要包含数据库Database IO次数、Database IO数据量、Redo IO次数、Redo IO量。
Memory Statistics	包含最大进程内存、进程已经使用内存、最大共享内存、已经使用共享内存大小等。

表 3-2 DETAIL 级别诊断报告

诊断类别	描述
Time Model	主要用于评估当前系统在时间维度的性能表现。 包含系统在各个阶段上消耗的时间：内核时间、CPU时间、执行时间、解析时间、编译时间、查询重写时间、计划生成时间、网络时间、IO时间。
SQL Statistics	主要用于SQL语句性能问题的诊断。 包含归一化的SQL的性能指标在多个维度上的排序：Elapsed Time、CPU Time、Rows Returned、Tuples Reads、Executions、Physical Reads、Logical Reads。这些指标的种类包括：执行时间，执行次数、行活动、Cache IO等。

诊断类别	描述
Wait Events	主要用于系统关键资源，关键时间的详细性能诊断。 包含所有关键事件在一段时间内的表现，主要是事件发生的次数，消耗的时间。
Cache IO Stats	用于诊断用户表和索引的性能。 包含所有用户表、索引上的文件读写，缓存命中。
Utility status	用于诊断后台任务性能。 包含复制等后台任务的性能。
Object stats	用于诊断数据库对象的性能。 包含用户表、索引上的表、索引扫描活动，insert、update、delete活动，有效行数量，表维护操作的状态等。
Configuration settings	用于判断配置是否有变更。 包含当前所有配置参数的快照。
SQL detail	显示unique query text信息。

特性增强

无。

特性约束

- WDR snapshot性能快照会采集不同database的性能数据，如果数据库实例中有大量的database或者大量表，做一次WDR snapshot会花费很长时间。
- 如果在大量DDL期间做WDR snapshot可能造成WDR snapshot失败。
- 在drop database时，做WDR snapshot可能造成WDR snapshot失败。

依赖关系

无。

3.3 慢 SQL 诊断

可获得性

本特性自openGauss 1.1.0版本开始引入。重构前慢SQL相关视图已废弃，包括：
dbe_perf.gs_slow_query_info、dbe_perf.gs_slow_query_history、
dbe_perf.global_slow_query_histroy、dbe_perf.global_slow_query_info。

特性简介

慢SQL诊断提供诊断慢SQL所需要的必要信息，帮助开发者回溯执行时间超过阈值的SQL，诊断SQL性能瓶颈。

客户价值

慢SQL提供给用户对于慢SQL诊断所需的详细信息，用户无需通过复现就能离线诊断特定慢SQL的性能问题。表和函数接口方便用户统计慢SQL指标，对接第三方平台。

特性描述

慢SQL能根据用户提供的执行时间阈值(log_min_duration_statement)，记录所有超过阈值的执行完毕的作业信息。

慢SQL提供表和函数两种维度的查询接口，用户从接口中能查询到作业的执行计划，开始、结束执行时间，执行查询的语句，行活动，内核时间，CPU时间，执行时间，解析时间，编译时间，查询重写时间，计划生成时间，网络时间，IO时间，网络开销，锁开销等。所有信息都是脱敏的。

特性增强

增加对慢SQL指标信息，安全性（脱敏），执行计划，查询接口的增强。

执行命令查看数据库实例中SQL语句执行信息

```
gsql> select * from dbe_perf.get_global_full_sql_by_timestamp(start_timestamp, end_timestamp);
```

例如：

```
openGauss=# select * from DBE_PERF.get_global_full_sql_by_timestamp('2020-12-01 09:25:22', '2020-12-31 23:54:41');
```

```
-[ RECORD 1 ]-----
```

```
+-----+
node_name      | dn_6001_6002_6003
db_name        | postgres
schema_name    | "$user",public
origin_node    | 1938253334
user_name      | user_dj
application_name | gsql
client_addr    |
client_port    | -1
unique_query_id | 3671179229
debug_query_id  | 72339069014839210
query          | select name, setting from pg_settings where name in (?)
start_time     | 2020-12-19 16:19:51.216818+08
finish_time    | 2020-12-19 16:19:51.224513+08
slow_sql_threshold | 1800000000
transaction_id  | 0
thread_id      | 139884662093568
session_id     | 139884662093568
n_soft_parse   | 0
n_hard_parse   | 1
query_plan     |
               | Datanode Name: dn_6001_6002_6003
               | Function Scan on pg_show_all_settings a (cost=0.00..12.50 rows=5 width=64)
               | Filter: (name = '***'.text)
...
```

执行命令查看数据库实例中慢SQL语句执行信息

```
gsql> select * from dbe_perf.get_global_slow_sql_by_timestamp(start_timestamp, end_timestamp);
```

```
openGauss=# select * from DBE_PERF.get_global_slow_sql_by_timestamp('2020-12-01 09:25:22', '2020-12-31 23:54:41');
```

```
-[ RECORD 1 ]-----
```

```
+-----+
node_name      | dn_6001_6002_6003
db_name        | postgres
schema_name    | "$user",public
origin_node    | 1938253334
user_name      | user_dj
application_name | gsql
client_addr    |
client_port    | -1
unique_query_id | 2165004317
```



```
debug_query_id | 72339069014839319
query          | select * from DBE_PERF.get_global_slow_sql_by_timestamp(?, ?);
start_time     | 2020-12-19 16:23:20.738491+08
finish_time    | 2020-12-19 16:23:20.773714+08
slow_sql_threshold | 10000
transaction_id | 0
thread_id      | 139884662093568
session_id     | 139884662093568
n_soft_parse   | 10
n_hard_parse   | 8
query_plan     | Datanode Name: dn_6001_6002_6003
               | Result (cost=1.01..1.02 rows=1 width=0)
               | InitPlan 1 (returns $0)
               | -> Seq Scan on pgxc_node (cost=0.00..1.01 rows=1 width=64)
               | Filter: (nodeis_active AND ((node_type = '***':"char") OR (node_type = '***':"char"))))
...

查看当前节点SQL语句执行信息
gsq> select * from statement_history;
例如:
openGauss=# select * from statement_history;
-[ RECORD 1 ]-----
+-----+
db_name      | postgres
schema_name  | "$user",public
origin_node  | 1938253334
user_name    | user_dj
application_name | gsql
client_addr  |
client_port  | -1
unique_query_id | 3671179229
debug_query_id | 72339069014839210
query        | select name, setting from pg_settings where name in (?)
start_time   | 2020-12-19 16:19:51.216818+08
finish_time  | 2020-12-19 16:19:51.224513+08
slow_sql_threshold | 1800000000
transaction_id | 0
thread_id    | 139884662093568
session_id   | 139884662093568
n_soft_parse | 0
n_hard_parse | 1
query_plan   | Datanode Name: dn_6001_6002_6003
               | Function Scan on pg_show_all_settings a (cost=0.00..12.50 rows=5 width=64)
               | Filter: (name = '***':"text)
```

特性约束

- 目前的SQL跟踪信息，基于正常的执行逻辑。执行失败的SQL，其跟踪信息不具有准确的参考价值。
- 节点重启，可能导致该节点的数据丢失。
- SQL语句执行完立即退出会话，可能会丢失该会话未刷新到系统表中的数据。
- 通过GUC参数设置收集SQL语句的数量，如果超过阈值，新的SQL语句执行信息不会被收集。
- 通过GUC参数设置单条SQL语句收集的锁事件详细信息最大字节数，如果超过阈值，新的锁事件详细信息不会被收集。
- 通过异步刷新方式刷新用户执行中的SQL信息，所以用户Query执行结束后，存在查询相关视图函数结果短暂时延。
- 部分指标信息(行活动、Cache/IO、时间分布等)依赖于dbe_perf.statement视图收集，如果该视图对应记录数超过预定大小(依赖GUC:instr_unique_sql_count)，则本特性可能不收集相关指标。

- statement_history表相关函数以及视图中details字段为二进制格式，如果需要解析详细内容，请使用对应函数：pg_catalog.statement_detail_decode(details, 'plaintext', true)。
- statement_history表查询需要切换至postgres库，其它库中数据为空
- statement_history表内容受track_stmt_stat_level控制，默认为'OFF,L0'，参数第一部分代表Full SQL，第二部分是慢SQL；对于慢SQL,只有SQL运行时间超过log_min_duration_statement时才会被记录至statement_history表

依赖关系

无。

3.4 Session 性能诊断

可获得性

本特性自openGauss 1.0.0版本开始引入。

特性简介

Session性能诊断提供给用户Session级别的性能问题诊断。

客户价值

- 查看最近用户Session最耗资源的事件。
- 查看最近比较占资源的SQL把资源都消耗在哪些等待事件上。
- 查看最近比较耗资源的Session把资源都花费在哪些等待事件上。
- 查看最近最耗资源的用户的信息。
- 查看过去Session相互阻塞的等待关系。

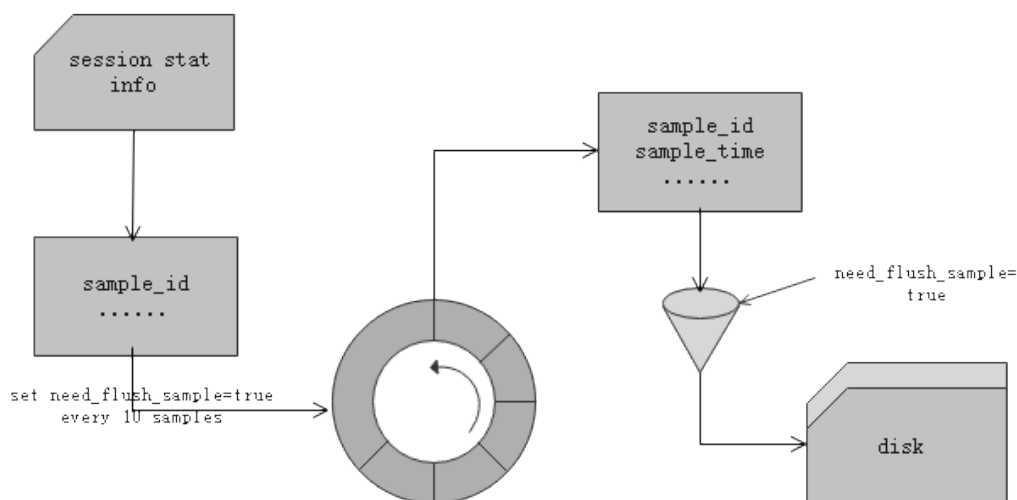
特性描述

Session性能诊断提供对当前系统所有活跃Session进行诊断的能力。由于实时采集所有活跃Session的指标对用户负载的影响加大，因此采取Session快照的技术对活跃Session的指标进行采样。从采样中统计出活跃Session的统计指标，这些统计指标从客户端信息、执行开始、结束时间，SQL文本，等待事件，当前数据库对象等维度，反映活跃Session的基本信息，状态，持有的资源。基于概率统计的活跃Session信息，可以帮助用户诊断系统中哪些Session消耗了更多的CPU、内存资源，哪些数据库对象是热对象，哪些SQL消耗了更多的关键事件资源等，从而定位出有问题Session，SQL，数据库设计。

Session采样数据分为两级，如[图3-1](#)所示：

1. 第一级为实时信息，存储在内存中，展示最近几分钟的活跃Session信息，具有最高的精度；
2. 第二级为持久化历史信息，存储在磁盘文件中，展示过去很长一段时间的历史活跃Session信息，从内存数据中抽样而来，适合长时间跨度的统计分析。

图 3-1 Session 性能诊断原理



部分使用场景如下所示：

1. 查看session之间的阻塞关系。

```
select sessionid, block_sessionid from pg_thread_wait_status;
```
2. 采样blocking session信息。

```
select sessionid, block_sessionid from DBE_PERF.local_active_session;
```
3. Final blocking session展示。

```
select sessionid, block_sessionid, final_block_sessionid from DBE_PERF.local_active_session;
```
4. 最耗资源的wait event。

```
SELECT s.type, s.event, t.count
FROM dbe_perf.wait_events s, (
SELECT event, COUNT(*)
FROM dbe_perf.local_active_session
WHERE sample_time > now() - 5 / (24 * 60)
GROUP BY event)t WHERE s.event = t.event ORDER BY count DESC;
```
5. 查看最近五分钟较耗资源的session把资源都花费在哪些event上。

```
SELECT sessionid, start_time, event, count
FROM (
SELECT sessionid, start_time, event, COUNT(*)
FROM dbe_perf.local_active_session
WHERE sample_time > now() - 5 / (24 * 60)
GROUP BY sessionid, start_time, event) as t ORDER BY SUM(t.count) OVER (PARTITION BY t.
sessionid, start_time)DESC, t.event;
```
6. 最近五分钟比较占资源的SQL把资源都消耗在哪些event上。

```
SELECT query_id, event, count
FROM (
SELECT query_id, event, COUNT(*)
FROM dbe_perf.local_active_session
WHERE sample_time > now() - 5 / (24 * 60)
GROUP BY query_id, event) t ORDER BY SUM(t.count) OVER (PARTITION BY t.query_id ) DESC,
t.event DESC;
```

特性增强

无。

特性约束

无。

依赖关系

无。

3.5 系统 KPI 辅助诊断

可获得性

本特性自openGauss 1.0.0版本开始引入。

特性简介

KPI是内核组件或者整体性能关键指标的视图呈现，基于这些指标，用户可以了解到系统运行的实时或者历史状态。

客户价值

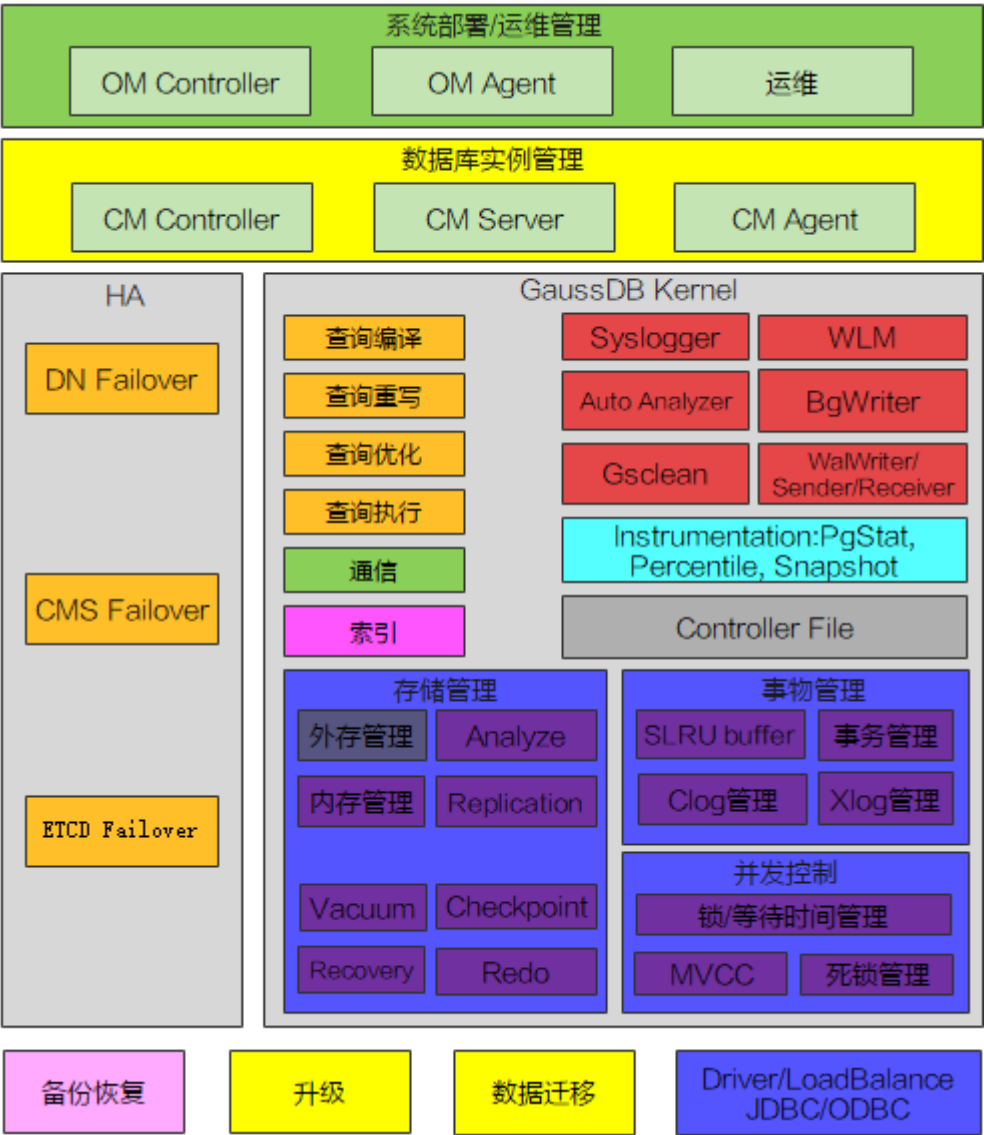
- 系统负载概要诊断
系统负载异常（过载、失速、业务SLA）精准告警，系统负载精准画像。
- 系统时间模型概要诊断
Instance和Query级别时间模型细分，诊断Instance和Query性能问题根因。
- Query性能诊断
数据库级Query概要信息，TopSQL，SQL CPU，IO消耗，执行计划，硬解析过多。
- 磁盘IO、索引、buffer性能问题
- 连接池，线程池异常
- Checkpoint，Redo（RTO）性能问题
- 系统I/O、LWLock、Waits性能问题诊断
诊断60+模块，240+关键操作性能问题。
- 函数级性能看护诊断（GSTRACE），功能诊断
50+存储和执行层函数trace。

特性描述

openGauss提供涵盖11大类，26个子类的KPI，包括：Instance、File、Object、Workload、Communication、Session、Thread、Cache IO、Lock、Wait Event、Cluster。

KPI指标内核的分布如[图3-2](#)所示。

图 3-2 KPI 指标内核分布



特性增强

无。

特性约束

- 对于utility语句不支持归一化，主要体现为非DML语句，比如：create/drop/copy/vacuum等语句。
- 当前归一化SQL仅记录顶层SQL，对于存储过程语句，不对存储过程内部的SQL进行归一化处理，只记录调用存储过程的SQL。

依赖关系

无。

4 数据库安全

- 4.1 访问控制模型
- 4.2 控制权和访问权分离
- 4.3 数据库认证机制
- 4.4 数据加密存储
- 4.5 数据库审计
- 4.6 网络通信安全
- 4.7 资源标签机制
- 4.8 统一审计机制
- 4.9 动态数据脱敏机制
- 4.10 行级访问控制
- 4.11 用户口令强度校验机制
- 4.12 全密态数据库等值查询
- 4.13 账本数据库机制
- 4.14 透明数据加密

4.1 访问控制模型

可获得性

本特性自openGauss 1.1.0版本开始引入。

特性简介

管理用户访问权限，为用户分配完成任务所需要的最小权限。

客户价值

客户依据自身需求创建对应的数据库用户并赋予相应的权限给操作人员，将数据库使用风险降到最低。

特性描述

数据库提供了基于角色的访问控制模型和基于三权分立的访问控制模型。在基于角色的访问控制模型下，数据库用户可分为系统管理员用户、监控管理员用户、运维管理员用户、安全策略管理员用户以及普通用户。系统管理员创建角色或者用户组，并为角色分配对应的权限；监控管理员查看dbe_perf模式下的监控视图或函数；安全策略管理员创建资源标签、脱敏策略、统一审计策略。用户通过绑定不同的角色获得角色所拥有的对应的操作权限。

在基于三权分立的访问控制模型下，数据库用户可分为系统管理员、安全管理员、审计管理员、监控管理员用户、运维管理员用户、安全策略管理员用户以及普通用户。安全管理员负责创建用户，系统管理员负责为用户赋权，审计管理员负责审计所有用户的行为。

默认情况下，使用基于角色的访问控制模型。客户可通过设置GUC参数enableSeparationOfDuty为on来切换。

特性增强

无。

特性约束

系统管理员的具体权限受GUC参数enableSeparationOfDuty控制；

三权分立开关和关闭切换时需要重启数据库，且无法对新模型下不合理的用户权限进行自主识别，需要DBA识别并修正；

依赖关系

无。

4.2 控制权和访问权分离

可获得性

本特性自openGauss 1.1.0版本开始引入。

特性简介

支持控制权和访问权分离。

客户价值

针对管理员用户，表对象的控制权和访问权要能够分离，提高普通用户数据安全性。

特性描述

对于有多个业务部门，各部门间使用不同的数据库用户进行业务操作，同时有一个同级的数据库维护部门使用数据库管理员进行维护操作的场景下，业务部门可能希望在未经授权的情况下，管理员用户只能对各部门的数据进行控制操作（DROP、ALTER、TRUNCATE），但是不能进行访问操作（INSERT、DELETE、UPDATE、SELECT、COPY）。即针对管理员用户，表对象的控制权和访问权要能够分离，提高普通用户数据安全。

三权分立情况下，管理员对其他用户放在属于各自模式下的表无权限。但是，这种无权限包含了无控制权限，因此不能满足上面的诉求。为此，openGauss提供了私有用户方案。即在非三权分立模式下，创建具有INDEPENDENT属性的私有用户。具备CREATEROLE权限或者是系统管理员权限的用户可以创建私有用户或者修改普通用户的属性为私有用户，普通用户也可以修改自己的属性为私有用户。

```
openGauss=# CREATE USER user_independent WITH INDEPENDENT IDENTIFIED BY "1234@abc";
```

针对该用户的表对象，系统管理员在未经其授权前，只能进行控制操作（DROP、ALTER、TRUNCATE），无权进行INSERT、DELETE、SELECT、UPDATE、COPY、GRANT、REVOKE、ALTER OWNER操作。

特性增强

无。

特性约束

对于私有用户的表，需要谨慎授予其他用户对该表的trigger权限，以免其他用户利用触发器查看私有用户的数据。

若将私有用户表的相关权限授予其他非私有用户，系统管理员也会获得同样的权限。

依赖关系

无。

4.3 数据库认证机制

可获得性

本特性自openGauss 1.1.0版本开始引入。

特性简介

提供基于客户端/服务端(C/S)模式的客户端连接认证机制。

客户价值

加密认证过程中采用单向Hash不可逆加密算法PBKDF2，有效防止彩虹攻击。

特性描述

openGauss采用基本的客户端连接认证机制，客户端发起连接请求后，由服务端完成信息校验并依据校验结果发送认证所需信息给客户端（认证信息包括盐值，token以及

服务端签名信息)。客户端响应请求发送认证信息给服务端，由服务端调用认证模块完成对客户端认证信息的认证。用户的密码被加密存储在内存中。整个过程中口令加密存储和传输。当用户下次登录时通过计算相应的hash值并与服务端存储的key值比较来进行正确性校验。

特性增强

统一加密认证过程中的消息处理流程，可有效防止攻击者通过抓取报文猜解用户名或者密码的正确性。

特性约束

无。

依赖关系

无。

4.4 数据加密存储

可获得性

本特性自openGauss 1.1.0版本开始引入。

特性简介

提供对导入数据的加密存储。

客户价值

为客户提供加密导入接口，对客户认为是敏感信息的数据进行加密后存储在表内。

特性描述

openGauss提供加密函数`gs_encrypt_aes128()`、`gs_encrypt()`和解密函数`gs_decrypt_aes128()`、`gs_decrypt()`接口。通过加密函数，可以对需要输入到表内的某列数据进行加密后再存储到表格内。调用格式为：

```
gs_encrypt_aes128(column, key),gs_encrypt (decryptstr,keystr, decrypttype)
```

其中key为用户指定的初始口令，用于派生加密密钥。当客户需要对整张表进行加密处理时，则需要为每一列单独书写加密函数。

当具有对应权限的用户需要查看具体的数据时，可通过解密函数接口对相应的属性列进行解密处理，调用格式为：

```
gs_decrypt_aes128(column, key),gs_decrypt(decryptstr,keystr, decrypttype)
```

特性增强

无。

特性约束

无。

依赖关系

无。

4.5 数据库审计

可获得性

本特性自openGauss 1.1.0版本开始引入。

特性简介

审计日志记录用户对数据库的启停、连接、DDL、DML、DCL等操作。

客户价值

审计日志机制主要增强数据库系统对非法操作的追溯及举证能力。

特性描述

数据库安全对数据库系统来说至关重要。openGauss将用户对数据库的所有操作写入审计日志。数据库安全管理员可以利用这些日志信息，重现导致数据库现状的一系列事件，找出非法操作的用户、时间和内容等。

特性增强

无。

特性约束

无。

依赖关系

无。

4.6 网络通信安全

可获得性

本特性自openGauss 1.1.0版本开始引入。

特性简介

为保护敏感数据在互联网上传输的安全性，openGauss支持通过SSL加密客户端和服务端之间的通讯。

客户价值

保证客户的客户端与服务器通讯安全。

特性描述

openGauss支持SSL协议标准。SSL（Secure Socket Layer）协议是一种安全性更高的应用层通信协议，主要用于Web安全传输，SSL包含记录层和传输层，记录层协议确定传输层数据的封装格式，传输层安全协议使用X.509认证。SSL协议利用非对称加密演算来对通信方做身份认证，之后交换对称密钥作为会谈密钥。通过SSL协议可以有效保障两个应用间通信的保密性和可靠性，使客户与服务器之间的通信不被攻击者窃听。

openGauss支持TLS 1.2协议标准。TLS 1.2协议是一种安全性更高的传输层通信协议，它包括两个协议组，TLS记录协议和TLS握手协议，每一组协议具有很多不同格式的信息。TLS协议是独立于应用协议的，高层协议可以透明地分布在TLS协议上面。通过TLS协议可保证通信双方的数据保密性和数据完整性。

特性增强

证书签名算法强度检查：对于一些强度较低的签名算法，给出告警信息，提醒客户更换包含高强度签名算法的证书。

证书超时时间检查：如果距离超期日期小于7天则给出告警信息，提醒客户端更换证书。

证书权限检查：在创建连接阶段对证书的权限进行校验。

特性约束

从CA认证中心申请到正式的服务器、客户端的证书和密钥。（假设服务器的私钥为server.key，证书为server.crt，客户端的私钥为client.key，证书为client.crt，CA根证书名称为cacert.pem。）

需要打开SSL开关，并且配置证书和连接方式。

依赖关系

OpenSSL

4.7 资源标签机制

可获得性

本特性自openGauss 1.1.0版本开始引入。

特性简介

数据库资源是指数据库所记录的各类对象，包括数据库、模式、表、列、视图、trigger等，数据库对象越多，数据库资源的分类管理就越繁琐。资源标签机制是一种通过对具有某类相同“特征”的数据库资源进行分类标记而实现资源分类管理的一种技术。当管理员对数据库内某些资源“打上”标签后，可以基于该标签进行如审计或数据脱敏的管理操作，从而实现对标签所包含的所有数据库资源进行安全管理。

客户价值

合理的制定资源标签能够有效的进行数据对象分类，提高对象管理效率，降低安全策略配置的复杂性。当管理员需要对某组数据库资源对象做统一审计或数据脱敏等安全管理动作时，可将这些资源划分到一个资源标签，该标签即包含了具有某类特征或需要统一配置某种策略的数据库资源，管理员可直接对资源标签执行管理操作，大大降低了策略配置的复杂性和信息冗余程度，提高了管理效率。

特性描述

资源标签机制是将当前数据库内包含的各种资源进行“有选择性的”分类，管理员可以使用如下SQL语法进行资源标签的创建，从而将一组数据库资源打上标签：

```
CREATE RESOURCE LABEL schm_lb ADD SCHEMA(schema_for_label);  
CREATE RESOURCE LABEL tb_lb ADD TABLE(schema_for_label.table_for_label);  
CREATE RESOURCE LABEL col_lb ADD COLUMN(schema_for_label.table_for_label.column_for_label);  
CREATE RESOURCE LABEL multi_lb ADD SCHEMA(schema_for_label), TABLE(table_for_label);
```

其中，schema_for_label、table_for_label、column_for_label分别为待标记模式、表、列。schm_lb标签包含了模式schema_for_label；tb_lb包含了表table_for_label；col_lb包含了列column_for_label；multi_lb包含模式schema_for_label和列table_for_label。对这些已配置的资源标签进行如统一审计或动态数据脱敏也即是对标签所包含的每一个数据库资源进行管理。

当前，资源标签所支持的数据库资源类型包括：SCHEMA、TABLE、COLUMN、VIEW、FUNCTION。

特性增强

无。

特性约束

- 资源标签需要由具备POLADMIN和SYSADMIN属性的用户或初始用户创建。
- 不支持对临时表创建资源标签。
- 同一个基本表的列只可能属于一个资源标签。

依赖关系

无。

4.8 统一审计机制

可获得性

本特性自openGauss 1.1.0版本开始引入。

特性简介

审计机制是行之有效的安全管理方案，可有效解决攻击者抵赖，审计的范围越大，可监控的行为就越多，而产生的审计日志就越多，影响实际审计效率。统一审计机制是一种通过定制化制定审计策略而实现高效安全审计管理的一种技术。当管理员定义审计对象和审计行为后，用户执行的任务如果关联到对应的审计策略，则生成对应的审

计行为，并记录审计日志。定制化审计策略可涵盖常见的用户管理活动，DDL和DML行为，满足日常审计诉求。

客户价值

审计是日常安全管理中必不可少的行为，当使用传统审计机制审计某种行为时，如SELECT，会导致产生大量的审计日志，进而增加整个系统的I/O，影响系统的性能；另一方面，大量的审计日志会影响管理员的审计效率。统一审计机制使得客户可以定制化生成审计日志的策略，如只审计数据库账户A查询某个表table的行为。通过定制化审计，可以大大减少生成审计日志的数量，从而在保障审计行为的同时降低对系统性能的影响。而定制化审计策略可以提升管理员的审计效率。

特性描述

统一审计机制基于资源标签进行审计行为定制化，且将当前所支持的审计行为划分为access类和privileges类。一个完整的审计策略创建的SQL语法如下所示：

```
CREATE RESOURCE LABEL auditlabel add table(table_for_audit1, table_for_audit2);
CREATE AUDIT POLICY audit_select_policy ACCESS SELECT ON LABEL(auditlabel) FILTER ON ROLES(usera);
CREATE AUDIT POLICY audit_admin_policy PRIVILEGES ALTER, DROP ON LABEL(auditlabel) FILTER ON IP(local);
```

其中，auditlabel为本轮计划审计的资源标签，该资源标签中包含了两个表对象；audit_select_policy定义了用户usera对auditlabel对象的SELECT行为的审计策略，不区分访问源；audit_admin_policy定义了从本地对auditlabel对象进行ALTER和DROP操作行为的审计策略，不区分执行用户；当不指定ACCESS和PRIVILEGES的具体行为时，表示审计针对某一资源标签的所有支持的DDL和DML行为。当不指定具体的审计对象时，表示审计针对所有对象的操作行为。统一审计策略的增删改也会记录在统一审计日志中。

当前，统一审计支持的审计行为包括：

SQL类型	支持操作和对象类型
DDL	操作: ALL ALTER ANALYZE COMMENT CREATE DROP GRANT REVOKE SET SHOW 对象: DATABASE SCHEMA FUNCTION TRIGGER TABLE SEQUENCE FOREIGN_SERVER FOREIGN_TABLE TABLESPACE ROLE/USER INDEX VIEW DATA_SOURCE
DML	操作: ALL COPY DEALLOCATE DELETE EXECUTE REINDEX INSERT PREPARE SELECT TRUNCATE UPDATE

特性增强

无。

特性约束

- 统一审计策略需要由具备POLADMIN或SYSADMIN属性的用户或初始用户创建，普通用户无访问安全策略系统表和系统视图的权限。

- 统一审计策略语法要么针对DDL行为，要么针对DML语法行为，同一个审计策略不可同时包含DDL和DML行为；统一审计策略目前支持最多设置98个。
- 统一审计监控用户通过客户端执行的SQL语句，而不会记录数据库内部SQL语句。
- 同一个审计策略下，相同资源标签可以绑定不同的审计行为，相同行为可以绑定不同的资源标签，操作"ALL"类型包括DDL或者DML下支持的所有操作。
- 同一个资源标签可以关联不同的统一审计策略，统一审计会按照SQL语句匹配的策略依次打印审计信息。
- 统一审计策略的审计日志单独记录，暂不提供可视化查询接口，整个日志依赖于操作系统自带rsyslog服务，通过配置完成日志归档。
- 在云服务场景下，日志需要存储在OBS服务中；在混合云场景下，可部署Elastic Search收集协调节点日志和可视化处理。
- FILTER中的APP项建议仅在同一信任域内使用，由于客户端不可避免的可能出现伪造名称的情况，该选项使用时需要与客户端联合形成一套安全机制，减少误用风险。一般情况下不建议使用，使用时需要注意客户端仿冒的风险。
- FILTER中的IP地址以ipv4为例支持如下格式。

ip地址格式	示例
单ip	127.0.0.1
掩码表示ip	127.0.0.1 255.255.255.0
cidr表示ip	127.0.0.1/24
ip区间	127.0.0.1-127.0.0.5

依赖关系

无。

4.9 动态数据脱敏机制

可获得性

本特性自openGauss 1.1.0版本开始引入。

特性简介

数据脱敏是行之有效的数据库隐私保护方案之一，可以在一定程度上限制非授权用户对隐私数据的窥探。动态数据脱敏机制是一种通过定制化制定脱敏策略从而实现对隐私数据保护的一种技术，可以有效地在保留原始数据的前提下解决非授权用户对敏感信息的访问问题。当管理员指定待脱敏对象和定制数据脱敏策略后，用户所查询的数据库资源如果关联到对应的脱敏策略时，则会根据用户身份和脱敏策略进行数据脱敏，从而限制非授权用户对隐私数据的访问。

客户价值

数据隐私保护是数据库安全所需要具备的安全能力之一，可以在一定程度上限制非授权用户对隐私数据的访问，保证隐私数据安全。动态数据脱敏机制可以通过配置脱敏

策略实现对指定数据库资源信息的隐私保护，另一方面，脱敏策略的配置也具有一定的灵活性，可以仅针对特定用户场景实现有针对性的隐私保护能力。

特性描述

动态数据脱敏机制基于资源标签进行脱敏策略的定制化，可根据实际场景选择特定的脱敏方式，也可以针对某些特定用户制定脱敏策略。一个完整的脱敏策略创建的SQL语法如下所示：

```
CREATE RESOURCE LABEL label_for_creditcard ADD COLUMN(user1.table1.creditcard);
CREATE RESOURCE LABEL label_for_name ADD COLUMN(user1.table1.name);
CREATE MASKING POLICY msk_creditcard creditcardmasking ON LABEL(label_for_creditcard);
CREATE MASKING POLICY msk_name randommasking ON LABEL(label_for_name) FILTER ON IP(local),
ROLES(dev);
```

其中，label_for_creditcard和msk_name为本轮计划脱敏的资源标签，分别包含了两个列对象；creditcardmasking、randommasking为预置的脱敏函数；msk_creditcard定义了所有用户对label_for_creditcard标签所包含的资源访问时做creditcardmasking的脱敏策略，不区分访问源；msk_name定义了本地用户dev对label_for_name标签所包含的资源访问时做randommasking的脱敏策略；当不指定FILTER对象时则表示对所有用户生效，否则仅对标识场景的用户生效。

当前，预置的脱敏函数包括：

脱敏函数名	示例
creditcardmasking	'4880-9898-4545-2525' 将会被脱敏为 'xxxx-xxxx-xxxx-2525'，该函数仅对后4位之前的数字进行脱敏
basicemailmasking	'abcd@gmail.com' 将会被脱敏为'xxxx@gmail.com'，对出现第一个'@'之前的文本进行脱敏
fullemailmasking	'abcd@gmail.com' 将会被脱敏为 'xxxx@xxxxx.com'，对出现最后一个'.'之前的文本（除'@'符外）进行脱敏
alldigitsmasking	'alex123alex' 将会被脱敏为 'alex000alex'，仅对文本中的数字进行脱敏
shufflemasking	'hello word' 将会被随机打乱顺序脱敏为 'hlwoeor dl'，该函数通过字符乱序排列的方式实现，属于弱脱敏函数，语义较强的字符串不建议使用该函数脱敏。
randommasking	'hello word' 将会被脱敏为 'ad5f5ghdf5'，将文本按字符随机脱敏
maskall	'4880-9898-4545-2525' 将会被脱敏为 'xxxxxxxxxxxxxxxxxxxxxx'

每个脱敏函数规格如下：

脱敏函数名	支持的数据类型
creditcardmasking	BPCHAR, VARCHAR, NVARCHAR, TEXT（注：仅针对信用卡格式的文本类数据）

basicemailmasking	BPCHAR, VARCHAR, NVARCHAR, TEXT（注：仅针对email格式的文本类型数据）
fullemailmasking	BPCHAR, VARCHAR, NVARCHAR, TEXT（注：仅针对email格式的文本类型数据）
alldigitsmasking	BPCHAR, VARCHAR, NVARCHAR, TEXT（注：仅针对包含数字的文本类型数据）
shufflemasking	BPCHAR, VARCHAR, NVARCHAR, TEXT（注：仅针对文本类型数据）
randommasking	BPCHAR, VARCHAR, NVARCHAR, TEXT（注：仅针对文本类型数据）
maskall	BOOL, RELTIME, TIME, TIMETZ, INTERVAL, TIMESTAMP, TIMESTAMPTZ, SMALLDATETIME, ABSTIME, TEXT, BPCHAR, VARCHAR, NVARCHAR2, NAME, INT8, INT4, INT2, INT1, NUMERIC, FLOAT4, FLOAT8, CASH

对于不支持的数据类型，默认使用maskall函数进行数据脱敏，BOOL类型脱敏成'0'；RELTIME类型脱敏成'1970'；TIME, TIMETZ, INTERVAL类型脱敏成'00:00:00.0000+00'；TIMESTAMP, TIMESTAMPTZ, SMALLDATETIME, ABSTIME类型脱敏成'1970-01-01 00:00:00.0000'；TEXT, CHAR, BPCHAR, VARCHAR, NVARCHAR2, NAME类型脱敏成'x'；INT8, INT4, INT2, INT1, NUMERIC, FLOAT4, FLOAT8类型脱敏成'0'，若数据类型不属于maskall支持的类型则不支持创建脱敏策略，如果脱敏列涉及隐式转换，则结果以隐式转换后的数据类型为基础进行脱敏。另外需要说明的是，如果脱敏策略应用到数据列并生效，此时对该列数据的操作将以脱敏后的结果为基础而进行。

动态数据脱敏适用于和实际业务紧密相关的场景，根据业务需要为用户提供合理的脱敏查询接口以及报错处理逻辑，以避免通过撞库而获取原始数据。

特性增强

无。

特性约束

- 动态数据脱敏策略需要由具备POLADMIN或SYSADMIN属性的用户或初始用户创建，普通用户没有访问安全策略系统表和系统视图的权限。
- 动态数据脱敏只在配置了脱敏策略的数据表上生效，而审计日志不在脱敏策略的生效范围内。
- 在一个脱敏策略中，对于同一个资源标签仅可指定一种脱敏方式，不可重复指定。
- 不允许多个脱敏策略对同一个资源标签进行脱敏，除以下脱敏场景外：使用FILTER指定策略生效的用户场景，包含相同资源标签的脱敏策略间FILTER生效场景无交集，此时可以根据用户场景明确辨别资源标签被哪种策略脱敏。
- Filter中的APP项建议仅在同一信任域内使用，由于客户端不可避免的可能出现伪造名称的情况，该选项使用时需要与客户端联合形成一套安全机制，减少误用风险。一般情况下不建议使用，使用时需要注意客户端仿冒的风险。

- 对于带有query子句的INSERT或MERGE INTO操作，如果源表中包含脱敏列，则上述两种操作中插入或更新的结果为脱敏后的值，且不可还原。
- 在内置安全策略开关开启的情况下，执行ALTER TABLE EXCHANGE PARTITION操作的源表若在脱敏列则执行失败。
- 对于设置了动态数据脱敏策略的表，需要谨慎授予其他用户对该表的trigger权限，以免其他用户利用触发器绕过脱敏策略。
- 最多支持创建98个动态数据脱敏策略。
- 仅支持使用上述七种预置脱敏策略。
- 仅支持对只包含COLUMN属性的资源标签做脱敏。
- 仅支持对基本表的列进行数据脱敏。
- 仅支持对SELECT查询到的数据进行脱敏。
- FILTER中的IP地址以ipv4为例支持如下格式。

ip地址格式	示例
单ip	127.0.0.1
掩码表示ip	127.0.0.1 255.255.255.0
cidr表示ip	127.0.0.1/24
ip区间	127.0.0.1-127.0.0.5

依赖关系

无。

4.10 行级访问控制

可获得性

本特性自openGauss 1.1.0版本开始引入。

特性简介

行级访问控制特性将数据库访问控制精确到数据表行级别，使数据库达到行级访问控制的能力。不同用户执行相同的SQL查询操作，读取到的结果是不同的。

客户价值

不同用户执行相同的SQL查询操作，读取到的结果是不同的。

特性描述

用户可以在数据表创建行访问控制（Row Level Security）策略，该策略是指针对特定数据库用户、特定SQL操作生效的表达式。当数据库用户对数据表访问时，若SQL满足数据表特定的Row Level Security策略，在查询优化阶段将满足条件的表达式，按照属性PERMISSIVE、RESTRICTIVE类型，通过AND或OR方式拼接，应用到执行计划上。

行级访问控制的目的是控制表中行级数据可见性，通过在数据表上预定义Filter，在查询优化阶段将满足条件的表达式应用到执行计划上，影响最终的执行结果。当前受影响的SQL语句包括SELECT，UPDATE，DELETE。

特性增强

无。

特性约束

- 行级访问控制策略仅可以应用到SELECT、UPDATE和DELETE操作，不支持应用到INSERT和MERGE操作。
- 支持对行存表、行存分区表、列存表、列存分区表、复制表、unlogged表、hash表定义行级访问控制策略，不支持HDFS表、外表、临时表定义行级访问控制策略。
- 不支持对视图定义行级访问控制策略。
- 同一张表上可以创建多个行级访问控制策略，一张表最多允许创建100个行级访问控制策略。
- 初始用户和系统管理员不受行级访问控制策略的影响。
- 对于设置了行级访问控制策略的表，需要谨慎授予其他用户对该表的trigger权限，以免其他用户利用触发器绕过行级访问控制策略。

依赖关系

无。

4.11 用户口令强度校验机制

可获得性

本特性自openGauss 1.1.0版本开始引入。

特性简介

对用户访问数据库所设置的口令强度进行校验。

客户价值

用户无法设置过低强度的口令，加固客户数据安全。

特性描述

初始化数据库、创建用户、修改用户时需要指定密码。密码必须满足强度校验，否则会提示用户重新输入密码。账户密码复杂度要求如下：

- 包含大写字母（A-Z）的最少个数（password_min_uppercase）
- 包含小写字母（a-z）的最少个数（password_min_lowercase）
- 包含数字（0-9）的最少个数（password_min_digital）
- 包含特殊字符的最少个数（password_min_special）

- 密码的最小长度（password_min_length）
- 密码的最大长度（password_max_length）
- 至少包含上述四类字符中的三类。
- 不能和用户名、用户名倒写相同，本要求为非大小写敏感。
- 不能和当前密码、当前密码的倒写相同。
- 不能是弱口令。

说明

弱口令指的是强度较低，容易被破解的密码，对于不同的用户或群体，弱口令的定义可能会有所区别，用户需要自己添加定制化的弱口令。

参数password_policy设置为1时表示采用密码复杂度校验，默认值为1。

弱口令字典中的口令存放在gs_global_config系统表中（name字段为weak_password的记录是储存的弱口令），当创建用户、修改用户需要设置密码时，将会把用户设置口令和弱口令字典中存放的口令进行对比，如果命中，则会提示用户该口令为弱口令，设置密码失败。

弱口令字典默认为空，用户通过以下语法可以对弱口令字典进行增加和删除，示例如下：

```
CREATE WEAK PASSWORD DICTIONARY WITH VALUES ('password1'), ('password2');  
DROP WEAK PASSWORD DICTIONARY;
```

其中"password1", "password2"是用户事先准备的弱口令，该语句执行成功后将会存入弱口令系统表中。

当用户尝试通过CREATE WEAK PASSWORD DICTIONARY 插入表中已存在的弱口令时，会只在表中保留一条该弱口令。

DROP WEAK PASSWORD DICTIONARY语句会清空整张系统表弱口令相关部分。

gs_global_config系统表没有唯一索引，不建议用户通过COPY FROM命令重复用相同数据对该表进行操作。

若用户需要对弱口令相关操作进行审计，应设置audit_system_object参数中的第三位为1。

特性增强

openGauss 1.1.0版本实现了弱口令字典功能。

特性约束

- 初始用户、系统管理员和安全管理员可以查看、新增、删除弱口令字典。
- 普通用户可以查看但是不能新增或者删除弱口令字典。

依赖关系

无。

4.12 全密态数据库等值查询

可获得性

本特性自openGauss 1.1.0版本开始引入。

特性简介

密态数据库意在解决数据全生命周期的隐私保护问题，使得系统无论在何种业务场景和环境下，数据在传输、运算以及存储的各个环节始终都处于密文状态。当数据拥有者在客户端完成数据加密并发送给服务端后，在攻击者借助系统脆弱点窃取用户数据的状态下仍然无法获得有效的价值信息，从而起到保护数据隐私的作用。

客户价值

由于整个业务数据流在数据处理过程中都是以密文形态存在，通过全密态数据库，可以实现：

1. 保护数据在云上全生命周期的隐私安全，无论数据处于何种状态，攻击者都无法从数据库服务端获取有效信息。
2. 帮助云服务提供商获取第三方信任，无论是企业服务场景下的业务管理员、运维管理员，还是消费者云业务下的应用开发者，用户通过将密钥掌握在自己手上，使得高权限用户无法获取数据有效信息。
3. 让云数据库借助全密态能力更好的遵守个人隐私保护方面的法律法规。

特性描述

从用户视角来看，整个密态等值查询的主要功能分为三部分，主要通过新增的KeyTool工具以及openGauss的增强gsql客户端来实现。

首先是客户端密钥管理功能，用户借助新增的KeyTool工具来实现CMK的生成，销毁，更新，同时支持密钥导入导出。通过该KeyTool工具的导入导出功能，CMK可以在不同的客户端间进行传输；同时，KeyTool实现了单个客户端侧的密钥管理，通过配置管理文件，可以对密钥的存储，更新进行管理。

其次该特性提供密钥创建功能和加密表创建功能，通过新增SQL语法“CREATE CLINET MASTER KEY”和“CREATE COLUMN ENCRYPTION KEY”实现在数据库侧记录和管理CMK和CEK元信息，CMK和CEK信息被记录在新增的系统表中。通过扩展“CREATE TABLE”语法为表的每一个敏感信息列指定列加密密钥和加密算法，方便后续数据密文存储。

最后该特性对用户提供的密态等值查询能力，该部分功能是整个特性的核心，虽然用户对整个密文查询无感知，但是在指定数据的加密信息后，针对该敏感数据的查询将受限于当前密态等值查询的支持规格。

从整体视角来看，该特性所实现的功能主要是为了用户基于敏感数据保护的诉求来存储管理数据并基于密文数据信息实现查询任务。

特性增强

无。

特性约束

- 数据以列级别进行加密，而无法按照行级别区分加密策略。
- 除Rename操作外，不支持通过Alter Table语法实现对加密表列的更改（包括加密列和非加密列之间的互转换），支持添加（Add）和删除（Drop）对应的加密列。
- 不支持对加密列设置大部分check限制性语法，但是支持check(column is not null)语法。
- 当support_extended_features = off时，不支持对加密列使用primary key、unique。当support_extended_features = on时，仅支持确定性加密列使用primary key、unique。
- 不支持不同数据类型之间的隐式转换。
- 不支持不同数据类型密文间的集合操作。
- 不支持加密列为数组类型。
- 不支持加密列创建范围分区。
- 加密列仅支持repeat和empty_blob()函数。
- 当前版本只支持gsq和JDBC（部署在linux操作系统）客户端，暂不支持ODBC等其他客户端实现密态等值查询。
- 使用JDBC客户端时，密态等值查询特性不能与负载均衡或自动选主特性同时开启。
- 只支持通过客户端执行copy from stdin的方式、\copy命令的方式以及insert into values(...)的方式往密态表中导入数据。
- 不支持从加密表到文件之间的copy操作。
- 不支持包括排序、范围查询以及模糊查询等在内的除等值以外的各种密态查询。
- 支持部分函数存储过程密态语法，密态支持函数存储过程具体约束查看开发者指南密态支持函数/存储过程章节。
- 不支持通过insert into...select..., merge into语法将非加密表数据插入到加密表数据中。
- 对于处于连接状态的连接请求，只有触发更新缓存的操作（更改用户，解密加密列失败等）和重新建连后才能感知服务端CEK信息变更；
- 不支持在由随机加密算法加密的列上进行密态等值查询；
- 对于密态等值查询运算中如果参与比较的两个属性条件采用不同的数据加密密钥，返回报错。
- 密态等值查询不支持时序表、外表，不支持ustore存储引擎加密。
- 对于数据库服务侧配置变更（pg_settings系统表、权限、密钥和加密列等信息），需要重新建立JDBC连接保证配置变更生效。
- 不支持多条SQL语句一起执行，insert into语句多批次执行场景不受此条约束限制。
- 密态等值查询支持的数据类型包括：

数据类	类型	描述
整型	tinyint/tinyint(n)	微整数，同int1
	smallint	小整数，同int2

	int4	常用整数
	binary_integer	Oracle兼容类型，常用整数
	bigint/bigint(n)	大整数，同int8
数值类型	numeric(p,s)	精度为p的准确数值类型
	number	Oracle兼容类型，等同numeric(p,s)
浮点类型	float4	单精度浮点数
	float8	双精度浮点数
	double precision	双精度浮点数
字符类型	char/char(n)	定长字符串，不足补空格，默认精度为1
	varchar(n)	变长字符串, n是指允许的最大字节长度
	text	文本类型
	varchar2(n)	Oracle兼容类型，等同varchar(n)
	clob	大文本类型
二进制类型	bytea	变长的二进制字符串
	blob	二进制大对象

依赖关系

无。

4.13 账本数据库机制

可获得性

本特性自openGauss 2.1.0版本开始引入。

特性简介

账本数据库特性，对用户指定的防篡改表增加校验信息，并记录用户对其数据的操作历史，通过数据和操作历史的一致性校验来保证用户数据无法被恶意篡改。在用户对防篡改表执行DML操作时，系统对防篡改表增加少量额外的行级校验信息，同时记录操作的SQL语句和数据的变化历史。通过特性提供的校验接口，用户可以方便的校验防篡改表中的数据是否与系统记录的操作信息是否一致。

客户价值

账本数据库通过提供对用户数据的操作记录、数据历史变化记录以及易用的一致性的校验接口，方便用户随时校验数据库中的敏感信息是否发生恶意篡改，有效提高数据库防篡改能力。

特性描述

账本数据库采用账本Schema对普通表和防篡改用户表进行隔离。用户在账本Schema中创建的行存表具有防篡改属性，即为防篡改用户表。用户向防篡改用户表中插入数据时，系统会自动生成少量行级校验信息。在用户执行DML时，系统会在全局区块表(GS_GLOBAL_CHAIN)中记录用户的操作、在用户表相应的历史表中记录数据的更改等信息，操作记录、数据变化记录和用户表中的数据三者严格保持一致。账本数据库提供高性能校验接口，能够供用户方便的校验数据的一致性，如果一致性校验失败，则说明数据可能发生篡改，需要及时联系审计管理员回溯操作记录历史。

特性增强

无。

特性约束

- 防篡改模式下的行存表具有防篡改属性，而临时表、UNLOGGED表、列存表、时序表等均不具有防篡改属性。
- 不允许修改防篡改用户表的结构，不允许truncate防篡改相关表，不允许将防篡改用户表切换到普通的Schema中，不允许将非防篡改表切换到防篡改Schema中。
- 防篡改表如果为分区表，则不支持exchange partition、drop partition、truncate partition等操作。
- 不支持使用函数、TRIGGER修改防篡改用户表数据。
- 防篡改用户表创建时不能有名为“hash”的列。
- 普通用户调用篡改校验接口只能校验自己有权查询的表。
- 只允许审计管理员和初始用户查询全局区块表和BLOCKCHAIN模式中的表，普通用户无权访问，所有用户均无权修改。
- 根据用户历史表命名规则，若待创建表的Schema或表名以 '_' 结尾或开头，可能会出现对应历史表名与已有表名冲突的情况，需要重新命名。

依赖关系

无。

4.14 透明数据加密

可获得性

本特性自openGauss 2.1.0版本开始引入。

特性简介

透明数据加密（Transparent Data Encryption），是数据库在将数据写入存储介质时对数据进行加密，从存储介质中读取数据时自动解密，防止攻击者绕过数据库认证机制直接读取数据文件中的数据，以解决静态数据泄露问题。该功能对于应用层几乎透明无感知，用户可根据需要决定是否启用透明数据加密功能。

客户价值

为了防止攻击者绕过数据库认证机制直接读取数据文件中的数据，可以通过透明数据加密功能对数据库的数据文件进行加密，保证用户必须在数据库启动后通过正常途径连接数据库，才可以读取解密后的数据，达到数据保护的目的。

特性描述

采用三层密钥结构实现密钥管理机制，即根密钥（RK）、主密钥（CMK）和数据加密密钥（DEK）。主密钥由根密钥加密保护，数据加密密钥由主密钥加密保护。数据加密密钥用于对用户数据进行加密和解密，每个表对应一个数据加密密钥。

支持表级加密，允许用户在创建表时指定是否对表进行加密和使用的加密算法，加密算法支持AES_128_CTR和SM4_CTR两种算法，算法一旦指定不可更改。对于创建表时指定为加密的表，数据库会自动为该表申请创建数据加密密钥，并将加密算法、密钥密文和对应主密钥ID等参数使用"keyword=value"格式保存在pg_class系统表中的reloptions字段中。

对于加密表，允许用户切换表的加密状态，即将加密表切换为非加密表，或将非加密表切换为加密表。如果在创建表时未使能加密功能，后续无法再切换为加密表。

对于加密表，支持数据加密密钥轮转。密钥轮转后，使用旧密钥加密的数据仍使用旧密钥解密，新写入的数据使用新密钥加密。密钥轮转时不更换加密算法。

特性增强

无。

特性约束

当前版本主要实现对接华为云KMS服务，支持表级密钥存储，实现对行存表加密，规格约束如下：

- 支持heap存储行存表加密。
- 不支持列存加密，不支持物化视图加密，不支持ustore存储引擎加密。
- 不支持索引和Sequence加密，不支持XLOG日志加密，不支持MOT内存表加密，不支持系统表加密。
- 用户在创建表时可以指定加密算法，加密算法一旦指定不可更改。如果创建表时设置enable_tde为on，但是不指定加密算法encrypt_algo，则默认使用AES_128_CTR加密算法。
- 如果在创建表时未开启加密功能或指定加密算法，后续无法再切换为加密表。
- 对于已分配加密密钥的表，切换表的加密和非加密状态，不会更换密钥和加密算法。
- 数据密钥轮转只有开启表加密功能时才支持轮转。
- 不支持单集群跨region的多副本主备同步，不支持单集群跨region的扩容，不支持跨region的备份恢复、集群容灾和数据迁移场景。
- 混合云场景如果使用华为云KMS和管控面功能，则可以支持透明数据加密，其他KMS服务如果接口不兼容则无法支持。
- 加密表的查询性能比不加密时会有所劣化，对于性能有较高要求的情况下需谨慎开启加密功能。

依赖关系

依赖外部KMS提供密钥管理服务。

5 企业级特性

- 5.1 函数及存储过程支持
- 5.2 支持SQL hint
- 5.3 全文索引
- 5.4 Copy接口支持容错机制
- 5.5 分区
- 5.6 高级分析函数支持
- 5.7 物化视图
- 5.8 支持HyperLogLog
- 5.9 在线添加索引
- 5.10 自治事务
- 5.11 全局临时表
- 5.12 伪列ROWNUM
- 5.13 支持存储过程调试
- 5.14 JDBC客户端负载均衡与读写分离
- 5.15 In-place Update存储引擎
- 5.16 发布订阅
- 5.17 外键锁增强
- 5.18 支持OLTP场景数据压缩

5.1 函数及存储过程支持

可获得性

本特性自openGauss 1.1.0版本开始引入。

特性简介

函数和存储过程是数据库中的一种重要对象，主要功能将用户特定功能的SQL语句集进行封装，并方便调用。

客户价值

1. 允许客户模块化程序设计，对SQL语句集进行封装，调用方便。
2. 存储过程会进行编译缓存，可以提升用户执行SQL语句集的速度。
3. 系统管理员通过限制执行某一存储过程的权限，能够实现对相应的数据的访问权限的限制，避免了非授权用户对数据的访问，保证了数据的安全。

特性描述

openGauss支持SQL标准中的函数及存储过程，其中存储过程兼容了部分主流数据库存储过程的语法，增强了存储过程的易用性。

特性增强

无。

特性约束

无。

依赖关系

无。

5.2 支持 SQL hint

可获得性

本特性自openGauss 1.1.0版本开始引入。

特性简介

支持SQL hint影响执行计划生成。

客户价值

提升SQL查询性能。

特性描述

Plan Hint为用户提供了直接影响执行计划生成的手段，用户可以通过指定join顺序，join、stream、scan方法，指定结果行数，指定重分布过程中的倾斜信息等多个手段来进行执行计划的调优，以提升查询的性能。

特性增强

无。

特性约束

无。

依赖关系

无。

5.3 全文索引

可获得性

本特性自openGauss 1.1.0版本开始引入。

特性简介

openGauss中提供的全文索引功能可以对文档进行预处理，并且可以使后续的搜索更快速。

客户价值

openGauss全文索引功能提供了查询可读性文档的能力，并且通过查询相关度将结果进行排序。

特性描述

构建全文索引的预处理过程包括：

- 将文档解析成token。
为每个文档标记不同类别的token是非常有必要的，例如：数字、文字、复合词、电子邮件地址，这样就可以做不同的处理。原则上token的类别依赖于具体的应用，但对于大多数的应用来说，可以使用一组预定义的token类。
- 将token转换为词素。
词素像token一样是一个字符串，但它已经标准化处理，这样同一个词的不同形式是一样的。例如，标准化通常包括：将大写字母折成小写字母、删除后缀（如英语中的s或者es）。这将允许通过搜索找到同一个词的不同形式，不需要繁琐地输入所有可能的变形样式。同时，这一步通常会删除停用词。这些停用词通常因为太常见而对搜索无用。（总之，token是文档文本的原片段，而词素被认为是有用的索引和搜索词。）openGauss使用词典执行这一步，且提供了各种标准的词典。
- 保存搜索优化后的预处理文档。
比如，每个文档可以呈现为标准化词素的有序组合。伴随词素，通常还需要存储词素位置信息以用于邻近排序。因此文档包含的查询词越密集其排序越高。词典能够对token如何标准化做到细粒度控制。使用合适的词典，可以定义不被索引的停用词。

特性增强

无。

特性约束

openGauss的全文检索功能当前限制约束是：

- 每个分词长度必须小于2K字节。
- tsvector结构（分词+位置）的长度必须小于1MB。
- tsvector的位置值必须大于0，且小于等于16,383。
- 每个分词在文档中位置数必须小于256，若超过将舍弃后面的位置信息。

依赖关系

无。

5.4 Copy 接口支持容错机制

可获得性

本特性自openGauss 1.0.0版本开始引入。

特性简介

支持将Copy过程中的部分错误导入到指定的错误表中，并且保持Copy过程不被中断。

客户价值

提升Copy功能的可用性和易用性，提升对于源数据格式异常等常见错误的容忍性和鲁棒性。

特性描述

openGauss提供用户封装好的Copy错误表创建函数，并允许用户在使用Copy From指令时指定容错选项，使得Copy From语句在执行过程中部分解析、数据格式、字符集等相关的报错不会报错中断事务、而是被记录至错误表中，使得在Copy From的目标文件即使有少量数据错误也可以完成入库操作。用户随后可以在错误表中对相关的错误进行定位以及进一步排查。

特性增强

无。

特性约束

支持容错的具体错误种类请参见《开发者指南》中“导入数据 > 使用COPY FROM STDIN导入数据 > 处理错误表”章节。

依赖关系

无。

5.5 分区

可获得性

本特性自openGauss 1.1.0版本开始引入。

特性简介

数据分区是在一个节点内部对数据按照用户指定的策略做进一步的水平分表，将表按照指定范围划分为多个数据互不重叠的部分。

客户价值

对于大多数用户使用场景，分区表和普通表相比具有以下优点：

- 改善查询性能：对分区对象的查询可以仅搜索自己关心的分区，提高检索效率。
- 增强可用性：如果分区表的某个分区出现故障，表在其他分区的数据仍然可用。
- 均衡I/O：可以把不同的分区映射到不同的磁盘以平衡I/O，改善整个系统性能。

特性描述

目前openGauss数据库支持范围分区表、列表分区表、哈希分区表、间隔分区表，二级分区表：

- 范围分区表：将数据基于范围映射到每一个分区，这个范围是由创建分区表时指定的分区键决定的。这种分区方式是最为常用的。
范围分区功能，即根据表的一列或者多列，将要插入表的记录分为若干个范围（这些范围在不同的分区里没有重叠），然后为每个范围创建一个分区，用来存储相应的数据。
- 列表分区表：将数据基于各个分区内包含的键值映射到每一个分区，分区包含的键值在创建分区时指定。
列表分区功能，即根据表的一列，将要插入表的记录中出现的键值分为若干个列表（这些列表在不同的分区里没有重叠），然后为每个列表创建一个分区，用来存储相应的数据。
- 哈希分区表：将数据通过哈希映射到每一个分区，每一个分区中存储了具有相同哈希值的记录。
哈希分区功能，即根据表的一列，通过内部哈希算法将要插入表的记录划分到对应的分区中。
- 间隔分区表：间隔分区是一种特殊的范围分区，相比范围分区，新增间隔值定义，当插入记录找不到匹配的分区时，可以根据间隔值自动创建分区。
- 二级分区表：二级分区表是在一级分区的基础上再进行分区，分区方案是由两个一级分区的分区方案组合而来的，目前二级分区表支持范围分区，列表分区，哈希分区交叉组合的9种分区策略。

用户在CREATE TABLE时增加PARTITION参数，即表示针对此表应用数据分区功能。用户可以在实际使用中根据需要调整建表时的分区键，使每次查询结果尽可能存储在相同或者最少的分区内（称为“分区剪枝”），通过获取连续I/O大幅度提升查询性能。

实际业务中，时间经常被作为查询对象的过滤条件。因此，用户可考虑选择时间列为分区键，键值范围可根据总数据量、一次查询数据量调整。

特性增强

支持范围分区表的合并功能。

特性约束

无。

依赖关系

无。

5.6 高级分析函数支持

可获得性

本特性自openGauss 1.1.0版本开始引入。

特性简介

无。

客户价值

我们提供窗口函数来进行数据高级分析处理。窗口函数将一个表中的数据进行预先分组，每一行属于一个特定的组，然后在这个组上进行一系列的关联分析计算。这样可以挖掘出每一个元组在这个集合里的一些属性和与其他元组的关联信息。

特性描述

简单举例说明窗口分析功能：分析某一部门内每个人的薪水和部门平均薪水的对比。

```
SELECT depname, empno, salary, avg(salary) OVER (PARTITION BY depname) FROM empsalary;
depname | empno | salary | avg
-----+-----+-----+-----
develop | 11 | 5200 | 5020.00000000000000000000
develop | 7 | 4200 | 5020.00000000000000000000
develop | 9 | 4500 | 5020.00000000000000000000
develop | 8 | 6000 | 5020.00000000000000000000
develop | 10 | 5200 | 5020.00000000000000000000
personnel | 5 | 3500 | 3700.00000000000000000000
personnel | 2 | 3900 | 3700.00000000000000000000
sales | 3 | 4800 | 4866.66666666666666666667
sales | 1 | 5000 | 4866.66666666666666666667
sales | 4 | 4800 | 4866.66666666666666666667
(10 rows)
```

可以看到，通过这个avg(salary) OVER (PARTITION BY depname)分析函数，每一个人的薪水和与部门的平均薪水很容易计算出来。

目前，系统支持row_number(), rank(), dense_rank(), percent_rank(), cume_dist(), ntile(), lag(), lead(), first_value(), last_value(), nth_value()分析函数。具体的函数用法和语句请参见《开发者指南》中“SQL参考 > 函数和操作符 > 窗口函数”章节。

特性增强

无。

特性约束

无。

依赖关系

无。

5.7 物化视图

可获得性

本特性自openGauss 1.1.0版本开始引入。

特性简介

物化视图实际上就是一种特殊的物理表，物化视图是相对普通视图而言的。普通视图是虚拟表，应用的局限性较大，任何对视图的查询实际上都是转换为对SQL语句的查询，性能并没有实际上提高。而物化视图实际上就是存储SQL所执行语句的结果，起到缓存的效果。

客户价值

使用物化视图功能提升查询效率。

特性描述

支持全量物化视图和增量物化视图，全量物化视图只支持全量更新。增量物化视图可实现异步更新功能，用户可通过执行语句把新增数据刷新到物化视图中。

特性增强

无。

特性约束

仅支持基表简单过滤查询和UNION ALL语句。

依赖关系

无。

5.8 支持 HyperLogLog

可获得性

本特性自openGauss 1.1.0版本开始引入。

特性简介

通过使用HyperLogLog相关函数，计算唯一值个数Count(Distinct)，提升性能。

客户价值

提升AP/TP类查询的性能。

特性描述

HLL（HyperLoglog）是统计数据集中唯一值个数的高效近似算法。它有着计算速度快，节省空间的特点，不需要直接存储集合本身，而是存储一种名为HLL的数据结构。每当有新数据加入进行统计时，只需要把数据经过哈希计算并插入到HLL中，最后根据HLL就可以得到结果。

HLL在计算速度和所占存储空间上都占优势。在时间复杂度上，Sort算法需要排序至少 $O(n\log n)$ 的时间，虽说Hash算法和HLL一样扫描一次全表 $O(n)$ 的时间就可以得出结果，但是存储空间上，Sort算法和Hash算法都需要先把原始数据存起来再进行统计，会导致存储空间消耗巨大。而对HLL来说不需要存原始数据，只需要维护HLL数据结构，所以占用空间始终是1280字节常数级别。

特性增强

无。

特性约束

无。

依赖关系

无。

5.9 在线添加索引

可获得性

本特性自openGauss 1.1.0版本开始引入。

特性简介

通过create index concurrently语法，以不阻塞DML的方式在线创建索引。

客户价值

创建索引时指定concurrently关键字，可以实现创建过程中不阻塞DML，不阻塞用户在线业务。

特性描述

创建索引时，一般会阻塞其他语句对该索引所依赖表的访问。指定concurrently关键字，加表加的是ShareUpdateExclusiveLock锁，可以实现创建过程中不阻塞DML。

在线添加索引，需要执行先后两次对该表的权标扫描来完成build，第一次扫描的时候创建索引，不阻塞读写操作；第二次扫描的时候合并更新第一次扫描到目前为止发生的变更。由于需要执行两次对表的扫描和build，而且必须等待现有的所有可能对该表执行修改的事务结束。这意味着该索引的创建比正常耗时更长，同时因此带来的CPU和I/O消耗对其他业务也会造成影响。

特性增强

无。

特性约束

- 在线添加索引时只能指定一个索引的名称。
- 普通CREATE INDEX命令可以在事务内执行，但是CREATE INDEX CONCURRENTLY不可以在事务内执行。
- 列存表、分区表和临时表不支持CONCURRENTLY方式创建索引。

依赖关系

无。

5.10 自治事务

可获得性

本特性自openGauss 1.1.0版本开始引入。

特性简介

子事务的提交不受主事务提交或回滚影响的一类事务。

客户价值

满足客户多样化使用场景。

特性描述

自治事务为在主事务执行过程中，以独立的事务上下文执行指定的某一类SQL语句，其commit和rollback不受主事务commit和rollback的影响。

用户自定义函数、存储过程支持自治事务。

典型使用场景是，用一个表记录主事务执行过程中的操作信息，并且在主事务失败回退的时候，表中记录的操作信息不予回退。

特性增强

无。

特性约束

- 触发器函数不支持自治事务。
- 函数或者存储过程的自治事务块中，静态sql语句不支持变量传递。
- 自治事务不支持执行嵌套。
- 包含自治事务的函数，不支持参数传递的返回值。
- 包含自治事务的存储过程/函数，不支持exception异常处理。

依赖关系

无。

5.11 全局临时表

可获得性

本特性自openGauss 1.1.0版本开始引入。

特性简介

临时表顾名思义是不保证持久化的表，其生命周期一般跟session或者事务绑定，可以方便用于表达处理过程中的一些临时数据存放，加速查询。

客户价值

提升临时表的表达能力和易用性

特性描述

全局临时表的元数据对所有会话可见，会话结束后元数据继续存在。会话与会话之间的用户数据、索引和统计信息相互隔离，每个会话只能看到和更改自己提交的数据。

全局临时表有两种模式：一种是基于会话级别的（ON COMMIT PRESERVE ROWS），当会话结束时自动清空用户数据；一种是基于事务级别的（ON COMMIT DELETE ROWS），当执行commit或rollback时自动清空用户数据。建表时如果没有指定ON COMMIT选项，则缺省为会话级别。与本地临时表不同，全局临时表建表时可以指定非pg_temp开头的schema。

特性增强

在本地临时表的基础上增加了全局临时表的处理

特性约束

- 不支持并行扫描
- 不支持temp tablespace
- 不支持partition
- 不支持GIST索引
- 不支持User-defined统计信息pg_statistic_ext
- 不支持ON COMMIT DROP
- 不支持hash bucket 聚簇存储
- 不支持列存

依赖关系

无。

5.12 伪列 ROWNUM

可获得性

本特性自openGauss 1.0.1版本开始引入。

特性简介

ROWNUM为查询出来的每一行记录生成一个序号，从1开始依次递增且不会重复。

客户价值

- 兼容Oracle特性，方便数据库迁移。
- 与LIMIT特性类似，能够筛选出结果集的前n条记录。

特性描述

ROWNUM（伪列），给SQL查询中满足条件的记录按顺序标号得来。查询结果第一行ROWNUM为1，第二行为2，依次类推，第n行为n。通常用来筛选出查询结果集中的前n行数据，与openGauss中LIMIT功能类似。

特性增强

在内部执行时，优化器会将ROWNUM重写成LIMIT去执行，加快执行速率。

特性约束

- ROWNUM是伪列，不可作为别名，以免SQL语句出现歧义；
- 创建索引时不可使用ROWNUM。例如：create index index_name on table(rownum);
- 创建表时default值不可为ROWNUM。例如：create table table_name(id int default rownum);
- Where子句中不可使用rownum的别名。例如：select rownum rn from table where rn < 5;

- 在插入数据时不可使用ROWNUM。例如：insert into table values(rownum,' blue');
- 不可在无表查询中使用ROWNUM。例如：select * from (values(rownum,1)), x(a,b);
- 若 having 子句中含有ROWNUM（且不在聚合函数中）时，group by子句中必须含有ROWNUM（且不在聚合函数中）。

依赖关系

无。

5.13 支持存储过程调试

可获得性

本特性自openGauss 1.0.0版本开始引入。后经三方库代码目录结构调整临时删除，又于openGauss 1.1.0重新引入。

特性简介

提供调试存储过程的一组API，可以进行断点调试、变量打印等对存储过程的调试功能。

客户价值

提升客户基于openGauss开发存储过程的开发体验。

特性描述

存储过程是数据库中的一种重要对象，主要功能将用户特定功能的SQL语句集进行封装，并方便调用。一个存储过程中往往包含着众多的sql以及过程化的执行结构，根据所需业务的规模大小而定。然而书写一个大的存储过程，往往会伴随着逻辑BUG，仅通过执行存储过程观察结果很难甚至无法发现其中存在的问题，因此需要一个调试工具。

存储过程调试工具，提供一组调试接口，可以使存储过程单步执行，在执行过程中设置断点、打印变量等，便于SQL开发者能够及时便捷的发现错误、修正错误，更高效、更快捷、高质量的进行功能开发。

特性增强

无。

特性约束

无。

依赖关系

无。

5.14 JDBC 客户端负载均衡与读写分离

可获得性

本特性自openGauss 2.1.0版本开始引入。

特性简介

JDBC 客户端提供负载均衡与读写分离能力。

客户价值

可从JDBC客户端配置负载均衡与读写分离。

特性描述

客户端多节点IP+Port配置,适应多AZ间高可用切换、适应异地容灾切换。支持连接级读写分离配置。支持优先连接只读节点。多个只读节点连接分布均衡。

特性增强

无。

特性约束

无。

依赖关系

无。

5.15 In-place Update 存储引擎

可获得性

本特性自openGauss 2.1.0版本开始引入。

特性简介

In-place Update存储引擎（原地更新），是openGauss新增的一种存储模式。openGauss此前的版本使用的行存储引擎是Append Update（追加更新）模式。追加更新对于业务中的增、删以及HOT（HeapOnly Tuple）Update（即同一页面内更新）有很好的表现，但对于跨数据页面的非HOT UPDATE场景，垃圾回收不够高效，Ustore存储引擎可很好解决上述问题

客户价值

In-place Update存储引擎可有效的降低多次更新元组后占用存储空间问题。

特性描述

新增的In-place update存储引擎很好的解决了Append update存储引擎空间膨胀，元组较大的劣势，高效回滚段的设计是In-place update存储引擎的基础。

特性增强

无。

特性约束

无。

依赖关系

无。

5.16 发布订阅

可获得性

本特性自openGauss 3.0.0版本开始引入。

特性简介

发布订阅基于逻辑复制实现，其中有一个或者更多订阅者订阅一个发布者节点上的一个或者更多发布。订阅者从它们所订阅的发布节点拉取数据。实现跨数据库集群的数据实时同步。

客户价值

发布订阅的典型使用场景是：

- 在一个数据库或者一个数据库的子集中发生更改时，把增量的改变发送给订阅者。
- 在更改到达订阅者时引发触发器。
- 把多个数据库联合到单一数据库中（例如用于分析目的）。

特性描述

发布者上的更改会被实时发送给订阅者。订阅者以与发布者相同的顺序应用发布的数据，这样在一个订阅中能够保证发布的事务一致性。这种数据复制的方法有时候也被称为事务性复制。

订阅者数据库的行为与任何其他openGauss实例相同，并且可以被用作其他数据库的发布者，只需要定义它自己的发布。当订阅者被应用当作只读时，单一的订阅中不会有冲突。在另一方面，如果应用或者对相同表集合的订阅者执行了其他的写动作，冲突可能会发生。

特性增强

无。

特性约束

发布订阅基于逻辑复制实现，继承所有逻辑复制的限制，同时发布订阅还有下列额外的限制或者缺失的功能。

- 数据库模式和DDL命令不会被复制。初始模式可以手工使用pg_dump --schema-only进行拷贝。后续的模式改变需要手工保持同步。
- 序列数据不被复制。后台由序列支撑的serial或者标识列中的数据当然将被作为表的一部分复制，但是序列本身在订阅者上仍将显示开始值。如果订阅者被用作一个只读数据库，那么这通常不会是什么问题。不过，如果订阅者数据库预期有某种转换或者容错，那么序列需要被更新到最后的值，要么通过从发布者拷贝当前数据的防范（也许使用pg_dump），要么从表本身决定一个足够高的值。
- 只有表支持复制，包括分区表。试图复制其他类型的关系，例如视图、物化视图或外部表，将会导致错误。
- 同一数据库内的多个订阅不应当订阅内容重复的发布（指发布相同的表），否则会产生数据重复或者主键冲突。
- 如果被发布的表中包含不支持btree/hash索引的数据类型（如地理类型等），那么该表需要有主键，才能成功的复制UPDATE/DELETE操作到订阅端。否则复制会失败，同时订阅端会出现“FATAL: could not identify an equality operator for type xx”的日志。

依赖关系

发布订阅依赖逻辑复制功能。

5.17 外键锁增强

可获得性

本特性自openGauss 3.0.0版本开始引入。

特性简介

新增两类行锁，由share和update锁扩展到key share、share、no key update和update。非主键的更新获取的是no key update锁，外键触发器获取的行锁为key share锁，这两种类型的锁互不冲突，以此提升了外键锁的并发性。

客户价值

绝大多数的表更新操作为非主键的更新，该特性有效地减少了有外键约束的场景下多并发更新的阻塞，提升效率。

特性描述

当对父表一行元组的非主键列进行更新时，获取no key update锁；对子表对应元组的更新或插入，触发外键触发器，获取父表元组的key share锁。两者互不阻塞。

由于增加了互不冲突的行锁，多事务不再只由share锁组成，而有多种不同行锁的组合方式，依据如下的冲突表。

锁模式	key share	share	no key update	update
key share				X
share			X	X
no key update		X	X	X
update	X	X	X	X

特性增强

无。

特性约束

- 新增的行锁暂不支持ustore表

依赖关系

无

5.18 支持 OLTP 场景数据压缩

可获得性

本特性自openGauss 3.0.0版本开始引入。

特性简介

支持OLTP场景行存数据压缩，提供通用压缩算法，通过对数据页的透明页压缩和维护页面存储位置的方式，做到高压缩、高性能。磁盘持久化用2个文件存储，分别是压缩地址文件（扩展名.pca）和压缩数据文件（扩展名.pcd）。

客户价值

典型使用场景是：希望降低数据库磁盘的空间需求。

特性描述

新增的支持OLTP场景数据压缩可以降低行表、索引数据的磁盘存储空间需求，在IO密集的数据系统，可以有一定的性能提升。

特性约束

- 仅支持行存表、BTree索引，不可以在USTORE、段页式存储引擎上使用。
- 压缩表索引文件采用mmap访问，需要基于压缩表文件数量，设置足够的max_map_count。

6 应用开发接口

- 6.1 支持标准SQL
- 6.2 支持标准开发接口
- 6.3 PG接口兼容
- 6.4 支持PL/Java

6.1 支持标准 SQL

可获得性

本特性自openGauss 1.0.0版本开始引入。

特性简介

SQL是用于访问和处理数据库的标准计算机语言。SQL标准的定义分成核心特性以及可选特性，绝大部分的数据库都没有100%支撑SQL标准。

openGauss数据库支持SQL:2011大部分的核心特性，同时还支持部分的可选特性，为使用者提供统一的SQL界面。

客户价值

标准SQL的引入为所有的数据库厂商提供统一的SQL界面，减少使用者的学习成本和应用程序的迁移代价。

特性描述

具体的特性列表请参见《开发者指南》中“SQL参考>SQL语法”章节。

特性增强

无。

特性约束

无。

依赖关系

无。

6.2 支持标准开发接口

可获得性

本特性自openGauss 1.0.0版本开始引入。

特性简介

支持ODBC 3.5及JDBC 4.0标准接口。

客户价值

提供业界标准的ODBC及JDBC接口，保证用户业务快速迁移至openGauss。

特性描述

目前支持标准的ODBC 3.5及JDBC 4.0接口，其中ODBC支持SUSE、Win32、Win64平台，JDBC无平台差异。

特性增强

增加JDBC对接第三方日志框架功能。JDBC对接第三方日志框架功能可满足用户对日志管控的需求。

特性约束

无。

依赖关系

无。

6.3 PG 接口兼容

可获得性

本特性自openGauss 1.0.0版本开始引入。

特性简介

兼容PSQL客户端，兼容PostgreSQL标准接口。

客户价值

兼容PSQL客户端，兼容PostgreSQL标准接口，能够与PG生态工具无缝对接。

特性描述

兼容PSQL客户端，兼容PostgreSQL标准接口。

特性增强

无。

特性约束

无。

依赖关系

无。

6.4 支持 PL/Java

可获得性

本特性自openGauss 1.0.0版本开始引入。

特性简介

支持java UDF。

客户价值

提供多种函数的开发环境。

特性描述

使用openGauss数据库的PL/Java函数，用户可以使用自己喜欢的Java IDE编写Java方法，并将包含这些方法的jar文件安装到openGauss数据库中，然后使用该方法。openGauss PL/Java基于开源tada PL/Java 1.5.2开发，PL/Java所使用的JDK版本为Huawei JDK V100R001C00SPC190B003-b09。

特性增强

无。

特性约束

- Java UDF可以实现一些较为简单的java计算，强烈建议不要在Java UDF中封装业务。
- 强烈建议不要在Java函数中使用任何方式连接数据库，包括但不限于JDBC。
- 建议用户使用Huawei JDK V100R001C00SPC190B003-b09编译Java方法和jar文件。

- 暂不支持的数据类型：除表6-1提及之外的数据类型，包括自定义类型，复杂数据类型（Java Array类及派生类）。
- 暂不支持UDAF，UDTF。

表 6-1 PL/Java 默认数据类型映射关系

openGauss	Java
BOOLEAN	boolean
"char"	byte
bytea	byte[]
SMALLINT	short
INTEGER	int
BIGINT	long
FLOAT4	float
FLOAT8	double
CHAR	java.lang.String
VARCHAR	java.lang.String
TEXT	java.lang.String
name	java.lang.String
DATE	java.sql.Timestamp
TIME	java.sql.Time (stored value treated as local time)
TIMETZ	java.sql.Time
TIMESTAMP	java.sql.Timestamp
TIMESTAMPTZ	java.sql.Timestamp

依赖关系

PL/Java依赖JDK环境，目前openGauss中已包含JDK环境，无需用户安装。如果用户已安装（相同或不同版本的）JDK，也不会引起冲突，openGauss会使用Huawei JDK V100R001C00SPC190B003-b09来运行PL/Java。

7 AI 能力

人工智能技术最早可以追溯到上世纪50年代，甚至比数据库系统的发展历史还要悠久。但是，由于各种各样客观因素的制约，在很长的一段时间内，人工智能技术并没有得到大规模的应用，甚至还经历了几次明显的低谷期。到了近些年，随着信息技术的进一步发展，从前限制人工智能发展的因素已经逐渐减弱，所谓的ABC（AI、Big data、Cloud computing）技术也随之而诞生。

AI与数据库结合是近些年的行业研究热点，openGauss较早地参与了该领域的探索，并取得了阶段性的成果。AI特性子模块名为DBMind，相对数据库其他功能更为独立，大致可分为AI4DB、DB4AI以及AI in DB三个部分。

- AI4DB就是指用人工智能技术优化数据库的性能，从而获得更好地执行表现；也可以通过人工智能的手段实现自治、免运维等。主要包括自调优、自诊断、自安全、自运维、自愈等子领域；
- DB4AI就是指打通数据库到人工智能应用的端到端流程，通过数据库来驱动AI任务，统一人工智能技术栈，达到开箱即用、高性能、节约成本等目的。例如通过SQL-like语句实现推荐系统、图像检索、时序预测等功能，充分发挥数据库的高并行、列存储等优势，既可以避免数据和碎片化存储的代价，又可以避免因信息泄漏造成的安全风险；
- AI in DB 就是对数据库内核进行修改，实现原有数据库架构模式下无法实现的功能，如利用AI算法改进数据库的优化器，实现更精确的代价估计等。

本章节所涉及的功能独立存在于数据库安装目录(\$GAUSSHOME)的bin/dbmind目录中，各个子功能存在于dbmind的子目录components中。提供gs_dbmind命令行供用户调用。与此同时，对于数据库内置AI的功能（如DB4AI），以SQL语法和系统函数的形式呈现。

7.1 AI4DB: 数据库自治运维

7.2 DB4AI: 数据库驱动AI

7.3 AI in DB: 数据库内AI功能

7.1 AI4DB: 数据库自治运维

7.1.1 数据库指标采集、预测与异常监控

可获得性

本特性自openGauss 1.1.0版本开始引入。

特性简介

本特性是openGauss集成的、可以用于数据库指标采集、预测以及异常监控与诊断的AI工具，是DBMind套件中的一个组件。当前通过兼容Prometheus平台来采集数据库系统的指标，提供Prometheus exporter用于采集和加工数据库监控指标。通过监控指标时序数据，可以用来预测未来负载走向，诊断问题等。

客户价值

- 极大简化运维人员工作，释放大量劳动力，为公司节省成本。
- 用户可以通过指标采集、监控和预测功能提前感知问题，从而防止数据库发生意外，导致更大的损失。

特性描述

Prometheus是业内非常流行的开源监控系统，同时本身也是一款时序数据库。Prometheus的采集端称之为exporter，用来收集被监控模块的指标项。为了与Prometheus平台完成对接，DBMind分别实现了两款exporter，分别是用来采集数据库指标的openGauss-exporter，以及对采集到的指标进行二次加工的reprocessing-exporter。

本特性支持对采集到的指标进行预测，用户可通过修改配置文件来指定需要进行预测的关键系统指标（KPI），进而便于用户发现指标的走势，及时进行对应的运维操作。如预测内存使用率可以发现内存泄漏、预测磁盘使用情况可以在合适的时候扩容。

特性增强

在 openGauss 3.0.0 版本中，进行了大幅度改进，兼容Prometheus 平台，实现两个 exporter 用于对接Prometheus。

特性约束

- 数据库状态正常，并且用户已将数据目录写入环境变量；
- Python 版本要求3.6及以上；
- 配置Prometheus 监控平台，并启动本服务，以便监控数据可被收集。

依赖关系

Prometheus

7.1.2 慢 SQL 根因分析

可获得性

本特性自openGauss 3.0.0开始引入。

特性简介

慢SQL一直是数据运维中的痛点问题，如何有效诊断慢SQL根因是当前一大难题，工具结合openGauss自身特点融合了现网DBA慢SQL诊断经验，该工具可以支持慢SQL根因15+，能同时按照可能性大小输出多个根因并提供针对性的建议。

客户价值

为客户提供快速可靠的慢SQL发现及根因分析功能，极大简化了运维人员的工作。

特性描述

基于Prometheus数据采集方案，收集慢SQL根因分析需要的数据，包括系统资源信息（cpu usage、memory usage、IO）、负载信息（QPS）、大进程信息（包括外部大进程和数据库定时任务）、慢SQL文本信息、慢SQL开始执行时间和结束执行时间、慢SQL执行计划，临时文件信息等信息，而后，本功能根据AI算法计算最匹配的慢SQL根因，并给出对应的建议和置信度。

特性增强

无

特性约束

- 数据库状态正常、客户端能够正常连接；
- 具备Python3.6+的环境；
- 其中慢SQL的信息通过WDR报告获取，数据库WDR报告中会标记SQL是否是慢SQL，其相关GUC参数track_stmt_stat_level默认打开，否则需要用户手动打开，一般设置为track_stmt_stat_level='off, L0'，更高级别对性能会有一些的影响。数据采集部分由Prometheus方案实现，故需要用户配置Prometheus数据采集平台，本功能只专注于算法并从Prometheus中获取指标的序列信息；

依赖关系

无

7.1.3 索引推荐

可获得性

本特性自openGauss 1.1.0开始引入。

特性简介

本功能是一个覆盖多种任务级别和使用场景的数据库智能索引推荐工具，其具备单Query索引推荐功能、虚拟索引功能、workload级别索引推荐功能，可以为用户提供可靠的索引建议。

客户价值

为客户提供快速可靠的索引推荐功能，极大简化了运维人员的工作。

特性描述

单query索引推荐功能支持用户在数据库中直接进行操作，本功能基于查询语句的语义信息和数据库的统计信息，对用户输入的单条查询语句生成推荐的索引；虚拟索引功能支持用户在数据库中直接进行操作，本功能将模拟真实索引的建立，避免真实索引创建所需的时间和空间开销，用户基于虚拟索引，可通过优化器评估该索引对指定查询语句的代价影响；对于workload级别的索引推荐，用户可通过运行数据库外的脚本使用此功能，本功能将包含有多条DML语句的workload作为输入，最终生成一批可对整体workload的执行表现进行优化的索引。

特性增强

无

特性约束

数据库状态正常、客户端能够正常连接。

当前执行用户下安装有gspl工具，该工具路径已被加入到PATH环境变量中。

具备Python3.6+的环境。

依赖关系

无

7.1.4 参数调优与诊断

可获得性

本特性自openGauss 1.0.0版本开始引入

特性简介

本功能是一款数据库集成的参数调优工具，通过结合深度强化学习和全局搜索算法等AI技术，实现在无需人工干预的情况下，获取最佳数据库参数配置。本功能不强制与数据库环境部署到一起，支持独立部署，脱离数据库安装环境独立运行。

客户价值

该工具可以在任意场景下，快速给出当前负载的调参配置，减少DBA的人工干预，提升运维效果，满足客户期望。

特性描述

调优程序包含三种运行模式，分别是：

- recommend: 通过用户指定的用户名等信息登录到数据库环境中，获取当前正在运行的workload特征信息，根据上述特征信息生成参数推荐报告。报告当前数据库中不合理的参数配置和潜在风险等；输出根据当前正在运行的workload行为和特征；输出推荐的参数配置。**该模式是秒级的，不涉及数据库的重启操作，其他模式可能需要反复重启数据库。**

- train: 通过用户提供的benchmark信息，不断地进行参数修改和benchmark的执行。通过反复的迭代过程，训练强化学习模型，以使用户在后面通过tune模式加载该模型进行调优。
- tune: 使用优化算法进行数据库参数的调优，当前支持两大类算法，一种是深度强化学习，另一种是全局搜索算法（全局优化算法）。深度强化学习模式要求先运行train模式，生成训练后的调优模型，而使用全局搜索算法则不需要提前进行训练，可以直接进行搜索调优。

特性增强

无

特性约束

- 数据库状态正常、客户端能够正常连接、且要求数据库内导入数据，以便调优程序可以执行benchmark测试调优效果。
- 使用本工具需要指定登录到数据库的用户身份，要求该登录到数据库上的用户具有足够的权限，以便可以获得充足的数据库状态信息。
- 使用登录到数据库宿主机上的Linux用户，需要将\$GAUSSHOME/bin添加到PATH环境变量中，即能够直接运行gsqsl、gs_guc、gs_ctl等数据库运维工具。
- Python版本建议为Python3.6及以上，且运行环境中已经安装相应依赖，并能够正常启动调优程序。您可以独立安装一个python3.6+的环境，无需设置到全局环境变量中。不建议使用root用户权限安装本工具，如果以root身份安装本完并工具，使用其他用户身份运行本工具时，需要确保配置文件有读取权限。
- 本工具支持以三种模式运行，其中tune和train模式要求用户配置好benchmark运行环境，并导入数据，本工具将会通过迭代运行benchmark来判断修改后的参数是否有性能提升。
- recommend模式建议在数据库正在执行workload的过程中执行，以便获得更准确的实时workload信息。
- 本工具默认带有TPC-C、TPC-H、TPC-DS以及sysbench的benchmark运行脚本样例，如果用户使用上述benchmark对数据库系统进行压力测试，则可以对上述配置文件进行适度修改或配置。如果需要适配用户自己的业务场景，需要您参照benchmark目录中的template.py文件编写驱动您自定义benchmark的脚本文件。

依赖关系

无

7.1.5 慢 SQL 发现

可获得性

本特性自openGauss 1.1.0版本开始引入。

特性简介

本功能是一个SQL语句执行时间预测工具，通过模板化方法，实现在不获取SQL语句执行计划的前提下，依据语句逻辑相似度与历史执行记录，预测SQL语句的执行时间。

客户价值

- 工具不需要用户提供SQL执行计划，对数据库性能不会有任何影响。
- 不同于业内其他算法只局限于OLAP或者OLTP，本工具场景更加广泛。

特性描述

SQLdiag着眼于数据库的历史SQL语句，通过对历史SQL语句的执行表现进行总结归纳，将之再用于推断新的未知业务上。由于短时间内数据库SQL语句执行时长不会有太大的差距，SQLdiag可以从历史数据中检测出与已执行SQL语句相似的语句结果集，并基于SQL向量化技术和模板化方法预测SQL语句执行时长。

特性增强

无

特性约束

- 需要保证用户提供的历史日志及待预测负载的格式符合要求，可以使用数据库GUC参数开启收集，也可以通过监控工具采集。
- 为保证预测准确率，用户提供的历史语句日志应尽可能全面并具有代表性。
- 按照要求配置python环境。

依赖关系

无

7.2 DB4AI: 数据库驱动 AI

可获得性

本特性自openGauss 2.1.0版本开始引入。

特性简介

DB4AI是指利用数据库的能力驱动AI任务，实现数据存储、技术栈的同构。通过在数据库内集成AI算法，令openGauss具备数据库原生AI计算引擎、模型管理、AI算子、AI原生执行计划的能力，为用户提供普惠AI技术。不同于传统的AI建模流程，DB4AI“一站式”建模可以解决数据在各平台的反复流转问题，同时简化开发流程，并可通过数据库规划出最优执行路径，让开发者更专注于具体业务和模型的调优上，具备同类产品不具备的易用性与性能优势。

客户价值

- 通过本功能，用户无需手动编写AI模型代码，直接通过开箱即用的SQL语句即可执行机器学习模型的训练和预测，学习和使用成本极低；
- 避免数据碎片化存储和反复搬迁导致的额外开销；
- 更高的执行效率，本功能的AI模型训练效率极高，相比用户自行手动训练模型有数倍性能收益；
- 更严密的安全防护，从而避免训练AI模型导致数据泄露。

特性描述

openGauss的原生DB4AI能力，通过引入原生AI算子，简化操作流程，充分利用数据库优化器、执行器的优化与执行能力，获得高性能的数据库内模型训练能力。更简化的模型训练与预测流程、更高的性能表现，让开发者在更短时间内能更专注于模型的调优与数据分析上，而避免了碎片化的技术栈与冗余的代码实现。

特性增强

在openGauss 3.0.0 版本中支持更多算法。

特性约束

- 数据库状态正常

依赖关系

无

7.3 AI in DB: 数据库内 AI 功能

7.3.1 Predictor: AI 查询时间预测

可获得性

本特性自openGauss 1.0.0版本开始引入。

特性简介

Predictor是基于机器学习且具有在线学习能力的查询时间预测工具。在数据库内核中，通过不断学习数据库内收集的历史执行信息，实现计划的执行时间预测功能。

客户价值

在SQL语句执行之前，根据AI模型预测SQL执行时间。用户基于推测的SQL时间信息，可以发现可能的异常SQL语句，提前进行调优或者调度，以免影响系统运行。

特性描述

AI查询时间预测，依赖于本地查询计划收集。用户根据需求在查询执行时，收集实际查询计划（包括计划结构、算子类型、相关数据源、过滤条件等）、各算子节点实际执行时间、优化器估算代价、实际返回行数、优化器估算行数、并行数等信息。这些记录将保存在数据表中，并进行持久化管理包括定期进行数据失效清理。

数据库内核在自动收集历史数据后，由管理员进行编码，然后通过CURL接口向python端发送请求以配置模型、发送数据、触发训练。训练过程中可以通过调用tensorboard接口监控训练过程。模型最终返回各项目预测准确率，并保存最终模型。数据库根据aiEngine端返回结果更新模型信息相关系统表。

执行explain (analyze on, predictor <model_name>) SELECT ...” 时且模型收敛时触发，数据库根据Plan进行编码判断是否符合当前模型。如果符合模型，先向python端

发送加载模型请求（模型训练完成后只需加载一次），模型加载后对计划编码文件进行推理并返回各节点预测结果。

特性增强

无。

特性约束

- 数据库系统处于正常状态，用户通过身份验证成功登陆数据库访问拥有权限的数据。
- 用户执行的SQL语法正确无报错。
- 用户执行的SQL不会导致数据库异常等。
- 历史性能数据窗口内集群并发量稳定，集群规模、表结构、表数量不变，数据量无突变，涉及查询性能的guc参数不变。如果上述条件被破坏，模型失效，需要对所有历史性能数据进行失效并重新进行数据收集和重新训练模型。
- 加载模型时，原模型的结构、参数和训练信息等保存完整。若原模型信息丢失，模型无法正常加载，需要重新训练。
- 历史性能数据正常收集和编码，正常解析编码数据不报错。
- 用户通过预提供的安装脚本或者自行安装Python=3.6.4，configparser==3.8.1，Flask==0.12.2，Keras==2.2.4，numpy==1.16.4，scikit-learn==0.19.1，pandas==0.25.1，tensorboard==1.14.0，tensorflow-gpu==1.14.0或tensorflow==1.14.0。
- 用户环境中已经安装了OpenSSL，使用脚本或者按照使用说明生成了证书。
- 目前跨database namespace的模型同步需要管理员用户手动执行，不支持自动同步。

依赖关系

无。

8 工具链

[8.1 MySQL->openGauss迁移工具chameleon](#)

[8.2 openGauss客户端工具DataStudio](#)

8.1 MySQL->openGauss 迁移工具 chameleon

可获得性

本特性自openGauss 3.0.0版本开始引入。

特性简介

chameleon工具是一个基于Python语言的MySQL到openGauss的实时复制工具。该工具提供了初始全量数据的复制以及增量数据的实时复制能力，可实现数据从MySQL迁移至openGauss。对于数据的全量和增量迁移，支持MySQL中各种数据类型的迁移，同时对于MySQL中的浮点数据类型，包括decimal、dec、numeric、float、float4、float8、real、double、double precision、fixed数据类型，可保证迁移后数据精度不丢失。

客户价值

通过使用chameleon工具，可完成数据从MySQL搬迁至openGauss数据库。

特性描述

chameleon工具提供数据全量和增量复制功能，使得数据可以从MySQL迁移至openGauss数据库。对于数据的全量和增量迁移，chameleon工具中存储了MySQL数据类型与openGauss数据类型之间的映射关系，可支持MySQL中各种数据类型的迁移。特别地，对于MySQL中的浮点数据类型，包括decimal、dec、numeric、float、float4、float8、real、double、double precision、fixed数据类型，若数据类型中显示指定或默认含有精度，将转化为openGauss中的numeric[p, s]类型；若数据类型中未显示指定精度，将转化为openGauss中的numeric数据类型，基于此，可保证离线迁移和在线迁移后数据精度不丢失。

特性增强

无。

特性约束

- 支持MySQL 5.7版本。
- 对于数据类型映射后仍存在不兼容的情形，将导致表数据迁移失败，但不会终止后续的数据离线迁移过程。

依赖关系

无。

8.2 openGauss 客户端工具 DataStudio

可获得性

DataStudio可在openGauss官网获取安装包，或在gitee上获取源码自行编译。

特性简介

DataStudio 是面向开发人员和数据库管理员的通用集成开发环境。它简化了openGauss 数据库的开发和管理。

DataStudio 允许数据库开发人员:

- 管理和创建数据库对象
- 执行 SQL 语句或 SQL 脚本
- 编辑和执行PL/SQL语句
- 导入和导出表格数据

客户价值

Data Studio作为一款配套openGauss数据库使用的客户端软件（当前只支持openGauss），辅助数据库开发人员便捷地构建应用程序，以图形化界面形式提供数据库关键特性。降低数据库的使用门槛，提高用户的使用粘性。

特性描述

用户在Windows环境上，使用DataStudio远程连接openGauss，并在DataStudio上对openGauss进行操作。例如：

- 图形化的数据库客户端集成开发环境，允许连接openGauss数据库。
- 允许数据库开发者可以高效的进行SQL开发
- 管理/创建数据库对象(数据库、模式、函数、存储过程、表、序列、列、索引、约束、视图、用户/角色和表空间)
- 执行SQL语句或SQL脚本
- 创建和执行存储过程

- 表数据增、删、改、查
- 导入/导出表数据
- 显示DDL、导出DDL、导出DDL和数据
- 导入/导出连接信息
- SQL 格式化
- SQL 执行历史记录
- 显示执行计划、ER图

特性增强

无。

特性约束

- 支持windows系统
- 支持jdk11及以上版本

依赖关系

当前只支持openGauss

9 中间件

9.1 分布式数据库能力

9.2 使用kubernetes部署分布式数据库

9.1 分布式数据库能力

可获得性

本特性自openGauss 2.1.0版本开始引入。

特性简介

基于分布式中间件shardingsphere使openGauss具备分布式数据库能力。使用16个鲲鹏920(128核)节点组网(1*shardingsphere-proxy,7*shardingsphere-jdbc,8*openGauss)时，完美sharding性能>1000万tpmc。

客户价值

通过中间件构建逻辑上无资源限制的分布式数据库。

特性描述

通过shardingsphere中间件的分库分表能力，使多个openGauss数据库可以在逻辑上组成一个更大的数据库，同时具备分布式事务和弹性伸缩的能力，使用方式与openGauss数据库并无不同。

特性增强

无。

特性约束

无。

依赖关系

shardingsphere中间件。

9.2 使用 kubernetes 部署分布式数据库

可获得性

本特性自openGauss 2.1.0版本开始引入。

特性简介

一键式部署分布式数据库。

客户价值

快速完成分布式数据库搭建，验证和使用分布式能力。

特性描述

通过patroni实现计划内switchover和故障场景自动failover, 通过haproxy实现openGauss主备节点读写负载均衡，通过shardingsphere实现分布式能力，所有功能打包至镜像并提供一键式部署脚本。

特性增强

无。

特性约束

仅支持centos或openEuler操作系统。

依赖关系

shardingsphere、patroni、haproxy。