

云数据库 GaussDB

# 产品介绍

文档版本 01  
发布日期 2024-05-14



版权所有 © 华为云计算技术有限公司 2024。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云计算技术有限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，华为云计算技术有限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。

# 华为云计算技术有限公司

地址：贵州省贵安新区黔中大道交兴功路华为云数据中心 邮编：550029

网址：<https://www.huaweicloud.com/>

# 目录

<b>1 图解云数据库 GaussDB.....</b>	<b>1</b>
<b>2 什么是云数据库 GaussDB.....</b>	<b>3</b>
<b>3 应用场景.....</b>	<b>6</b>
<b>4 常用概念.....</b>	<b>7</b>
<b>5 产品优势.....</b>	<b>9</b>
<b>6 实例说明.....</b>	<b>10</b>
6.1 数据库实例状态.....	10
6.2 数据库实例规格.....	11
6.3 数据库实例存储类型.....	16
6.4 数据库引擎版本说明.....	17
6.5 数据库实例类型.....	18
6.6 数据库实例参数.....	20
6.7 数据库实例系统账号.....	36
<b>7 与 Oracle 兼容性说明.....</b>	<b>38</b>
7.1 分布式.....	38
7.1.1 Oracle 数据库兼容性概述.....	38
7.1.2 SQL 的基本元素.....	38
7.1.2.1 数据类型.....	39
7.1.2.2 数据类型比较规则.....	44
7.1.2.3 字面量.....	46
7.1.2.4 格式模型.....	47
7.1.2.5 空值.....	48
7.1.2.6 注释.....	48
7.1.2.7 数据库对象.....	48
7.1.2.8 数据库对象名称和限定符.....	50
7.1.2.9 SQL 语句中的引用架构对象和部件的语法.....	50
7.1.3 伪列.....	51
7.1.4 操作符.....	52
7.1.5 表达式.....	54
7.1.6 条件.....	55
7.1.7 常见的 SQL DDL 子句.....	56

7.1.8 SQL 查询和子查询.....	57
7.1.9 PL/SQL 语言.....	58
7.1.9.1 PL/SQL 基本语法.....	58
7.1.9.2 数据类型兼容性.....	61
7.1.9.3 控制语句.....	61
7.1.9.4 集合和 Record.....	63
7.1.9.5 静态 SQL.....	69
7.1.9.6 动态 SQL.....	72
7.1.9.7 Trigger.....	72
7.1.10 系统函数.....	81
7.1.10.1 单行函数.....	81
7.1.10.2 其它函数.....	95
7.1.11 系统视图.....	97
7.1.12 高级包.....	103
7.2 主备版.....	163
7.2.1 Oracle 数据库兼容性概述.....	163
7.2.2 SQL 的基本元素.....	163
7.2.2.1 数据类型.....	164
7.2.2.2 数据类型比较规则.....	169
7.2.2.3 字面量.....	172
7.2.2.4 格式模型.....	172
7.2.2.5 空值.....	173
7.2.2.6 注释.....	173
7.2.2.7 数据库对象.....	173
7.2.2.8 数据库对象名称和限定符.....	175
7.2.2.9 SQL 语句中的引用架构对象和部件的语法.....	176
7.2.3 伪列.....	176
7.2.4 操作符.....	177
7.2.5 表达式.....	180
7.2.6 条件.....	181
7.2.7 常见的 SQL DDL 子句.....	184
7.2.8 SQL 查询和子查询.....	185
7.2.9 PL/SQL 语言.....	186
7.2.9.1 PL/SQL 基本语法.....	186
7.2.9.2 数据类型兼容性.....	189
7.2.9.3 控制语句.....	190
7.2.9.4 集合和 Record.....	192
7.2.9.5 静态 SQL.....	199
7.2.9.6 动态 SQL.....	203
7.2.9.7 Trigger.....	203
7.2.10 系统函数.....	214
7.2.10.1 单行函数.....	214

7.2.10.2 其它函数.....	232
7.2.11 系统视图.....	234
7.2.12 高级包.....	241
<b>8 与 MySQL 兼容性说明.....</b>	<b>307</b>
8.1 分布式.....	307
8.1.1 MySQL 数据库兼容 MySQL 模式.....	307
8.1.1.1 MySQL 数据库兼容性概述.....	307
8.1.1.2 数据类型.....	307
8.1.1.2.1 数值数据类型.....	308
8.1.1.2.2 日期与时间数据类型.....	314
8.1.1.2.3 字符串数据类型.....	325
8.1.1.2.4 二进制数据类型.....	328
8.1.1.2.5 JSON 数据类型.....	329
8.1.1.2.6 数据类型支持的属性.....	330
8.1.1.2.7 数据类型转换.....	330
8.1.1.3 系统函数.....	332
8.1.1.3.1 流量控制函数.....	333
8.1.1.3.2 日期和时间函数.....	335
8.1.1.3.3 字符串函数.....	345
8.1.1.3.4 强制转换函数.....	349
8.1.1.3.5 加密函数.....	350
8.1.1.3.6 JSON 函数.....	350
8.1.1.3.7 聚合函数.....	352
8.1.1.3.8 数字操作函数.....	354
8.1.1.3.9 其他函数.....	355
8.1.1.4 操作符.....	355
8.1.1.5 字符集.....	357
8.1.1.6 排序规则.....	357
8.1.1.7 SQL.....	358
8.1.1.7.1 DDL.....	358
8.1.1.7.2 DML.....	365
8.1.1.7.3 DCL.....	376
8.1.1.8 驱动.....	376
8.1.1.8.1 JDBC.....	376
8.2 主备版.....	377
8.2.1 概述.....	377
8.2.2 MySQL 数据库兼容性 B 模式.....	377
8.2.2.1 MySQL 数据库兼容性概述.....	377
8.2.2.2 数据类型.....	377
8.2.2.2.1 数值数据类型.....	378
8.2.2.2.2 日期与时间数据类型.....	384
8.2.2.2.3 字符串数据类型.....	395

8.2.2.2.4 二进制数据类型.....	398
8.2.2.2.5 JSON 数据类型.....	399
8.2.2.2.6 数据类型支持的属性.....	400
8.2.2.2.7 数据类型转换.....	400
8.2.2.3 系统函数.....	402
8.2.2.3.1 流量控制函数.....	403
8.2.2.3.2 日期和时间函数.....	405
8.2.2.3.3 字符串函数.....	415
8.2.2.3.4 强制转换函数.....	421
8.2.2.3.5 加密函数.....	422
8.2.2.3.6 信息函数.....	422
8.2.2.3.7 JSON 函数.....	422
8.2.2.3.8 聚合函数.....	424
8.2.2.3.9 数字操作函数.....	425
8.2.2.3.10 其他函数.....	426
8.2.2.4 操作符.....	426
8.2.2.5 字符集.....	428
8.2.2.6 排序规则.....	428
8.2.2.7 表达式.....	429
8.2.2.8 SQL.....	429
8.2.2.8.1 DDL.....	429
8.2.2.8.2 DML.....	438
8.2.2.8.3 DCL.....	449
8.2.2.9 驱动.....	449
8.2.2.9.1 JDBC.....	449
8.2.3 MySQL 数据库兼容性 M-Compatibility 模式.....	450
8.2.3.1 MySQL 数据库兼容性概述.....	450
8.2.3.2 数据类型.....	451
8.2.3.2.1 数值数据类型.....	452
8.2.3.2.2 日期与时间数据类型.....	454
8.2.3.2.3 字符串数据类型.....	455
8.2.3.2.4 二进制数据类型.....	458
8.2.3.2.5 数据类型支持的属性.....	461
8.2.3.2.6 数据类型转换.....	461
8.2.3.3 系统函数.....	466
8.2.3.3.1 流程控制函数.....	467
8.2.3.3.2 日期和时间函数.....	467
8.2.3.3.3 字符串函数.....	472
8.2.3.3.4 强制转换函数.....	475
8.2.3.3.5 加密函数.....	476
8.2.3.3.6 比较函数.....	476
8.2.3.3.7 聚合函数.....	478

8.2.3.3.8 数字操作函数.....	479
8.2.3.3.9 其他函数.....	480
8.2.3.4 操作符.....	480
8.2.3.5 字符集.....	488
8.2.3.6 排序规则.....	489
8.2.3.7 事务.....	490
8.2.3.8 SQL.....	493
8.2.3.8.1 关键字.....	494
8.2.3.8.2 标识符.....	494
8.2.3.8.3 DDL.....	496
8.2.3.8.4 DML.....	510
8.2.3.8.5 DCL.....	518
8.2.3.8.6 其它语句.....	518
8.2.3.8.7 用户与权限.....	519
8.2.3.8.8 系统表和系统视图.....	524
<b>9 安全.....</b>	<b>529</b>
9.1 责任共担.....	529
9.2 身份认证与访问控制.....	530
9.2.1 服务的访问控制.....	530
9.3 数据保护技术.....	531
9.4 审计与日志.....	531
9.5 监控安全风险.....	532
9.6 故障恢复.....	532
9.7 认证证书.....	533
<b>10 权限管理.....</b>	<b>535</b>
<b>11 约束与限制.....</b>	<b>541</b>
<b>12 计费说明.....</b>	<b>544</b>
<b>13 GaussDB 与其他服务的关系.....</b>	<b>546</b>
<b>A 修订记录.....</b>	<b>547</b>

# 1 图解云数据库 GaussDB

---



图 1-1 初识华为云数据库 GaussDB

**初识华为云数据库 GaussDB**

新一代企业级分布式数据库

**1. 产品介绍**

GaussDB是华为自主创新研发的**分布式关系型数据库**。该产品具备企业级复杂事务混合负载能力，同时支持分布式事务，同城跨AZ部署，数据0丢失，支持**1000+**的扩展能力，**PB级**海量存储。同时拥有云上高可用，高可靠，高安全，弹性伸缩，一键部署，快速备份恢复，监控告警等关键能力，能为企业提供功能全面，稳定可靠，扩展性强，性能优越的企业级数据库服务。

适用行业：金融、互联网、政府、大企业

CM CN ETCD GTM OM

Computing network: TCP/RDMA

Data Node: 分布式SQL, 分布式执行, 存储引擎

主备 HA切换

X86/鲲鹏

工具集：数据迁移, SQL开发, 集群管理

# 2 什么是云数据库 GaussDB

GaussDB是华为自主创新研发的分布式关系型数据库。该产品支持分布式事务，同城跨AZ部署，数据0丢失，支持1000+的扩展能力，PB级海量存储。同时拥有云上高可用，高可靠，高安全，弹性伸缩，一键部署，快速备份恢复，监控告警等关键能力，能为企业提供功能全面，稳定可靠，扩展性强，性能优越的企业级数据库服务。

## 成长地图

您可以通过[云数据库GaussDB成长地图](#)快速了解GaussDB的相关概念、入门使用和高手进阶等。

## 如何使用云数据库 GaussDB

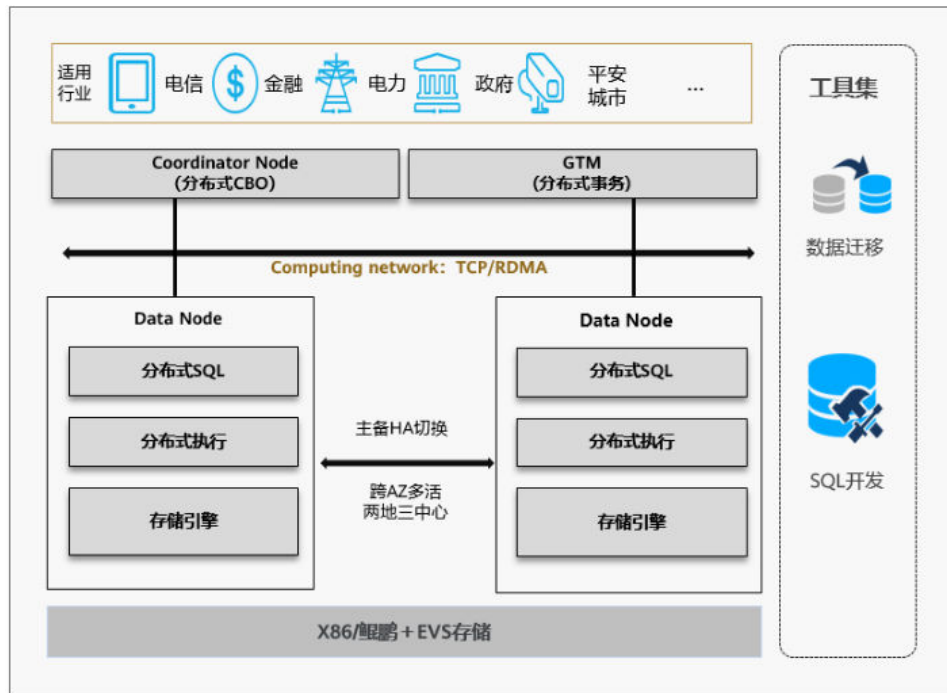
您可以通过如下方式使用GaussDB。

- 管理控制台：您可以使用[管理控制台](#)为您提供的Web界面完成GaussDB的相关操作。
- API：您可以编写代码调用API使用GaussDB，请参考《[云数据库GaussDB API参考](#)》。

## GaussDB 分布式形态整体架构

GaussDB分布式形态整体架构如下：

图 2-1 GaussDB 分布式形态整体架构图

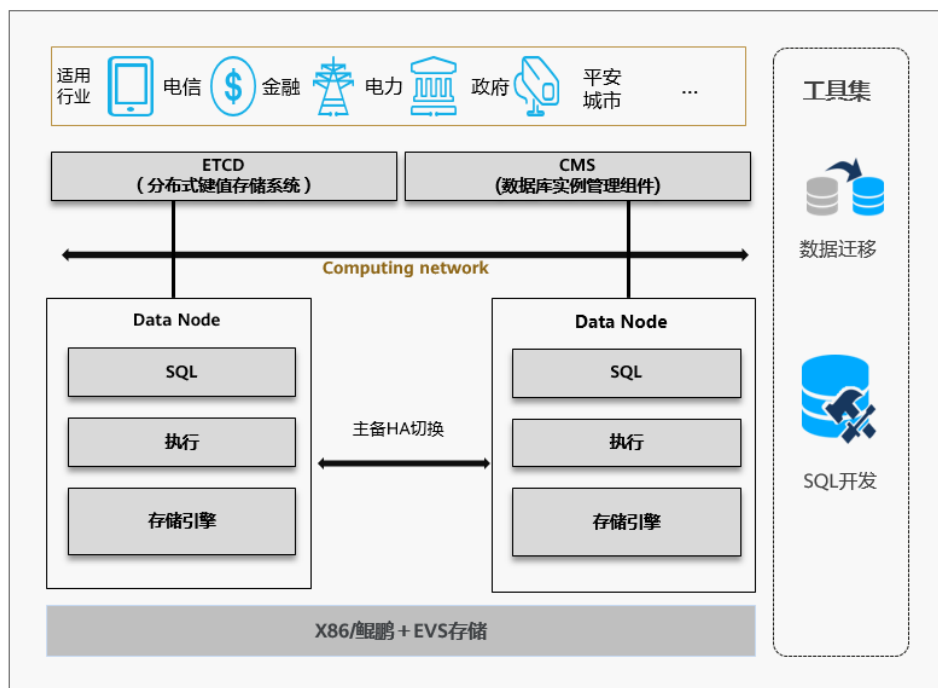


- Coordinator Node: 协调节点CN，负责接收来自应用的访问请求，并向客户端返回执行结果；负责分解任务，并调度任务分片在各DN上并行执行。
- GTM: 全局事务管理器（Global Transaction Manager），负责生成和维护全局事务ID、事务快照、时间戳、Sequence信息等全局唯一的信息。
- Data Node: 数据节点DN，负责存储业务数据、执行数据查询任务以及向CN返回执行结果。

## GaussDB 主备版形态整体架构

GaussDB 主备版形态整体架构如下：

图 2-2 GaussDB 主备版形态整体架构图



- ETCD：分布式键值存储系统（ Editable Text Configuration Daemon ）。用于共享配置和服务发现（ 服务注册和查找 ）。
- CMS：集群管理模块（ Cluster Manager ）。管理和监控分布式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。
- Data Node：数据节点DN，负责存储业务数据、执行数据查询任务以及返回执行结果。

# 3 应用场景

---

- **交易型应用**  
大并发、大数据量、以联机事务处理为主的交易型应用，如政务、金融、电商、O2O、电信CRM/计费等，服务能力支持高扩展、弹性扩缩，应用可按需选择不同的部署规模。
- **详单查询**  
具备PB级数据负载能力，通过内存分析技术满足海量数据边入库边查询要求，适用于安全、电信、金融、物联网等行业的详单查询业务。

# 4 常用概念

## 实例

GaussDB的最小管理单元是实例，一个实例代表了一个独立运行的数据库。用户可以在控制台创建和管理GaussDB实例。实例的状态、规格、存储类型和版本，请参考[实例说明](#)。

## 数据库引擎版本

GaussDB目前支持3.223、3.226、8.102版本。

## 实例类型

GaussDB支持分布式版和主备版实例。

- 分布式形态能够支撑较大的数据量，且提供了横向扩展的能力，可以通过扩容的方式提高实例的数据容量和并发能力。
- 主备版适用于数据量较小，且长期来看数据不会大幅度增长，但是对数据的可靠性，以及业务的可用性有一定诉求的场景。

## 实例规格

数据库实例各种规格（vCPU个数、内存（GB））请参考[数据库实例规格](#)。

## CN

协调节点（Coordinator Node），负责接收来自应用的访问请求，并向客户端返回执行结果；负责分解任务，并调度任务分片在各DN上并行执行。

## DN

数据节点（Data Node），负责存储业务数据、执行数据查询任务以及向CN返回执行结果。

## 自动备份

购买实例时，GaussDB服务默认开启自动备份策略，备份默认保留7天。实例创建成功后，您可对其进行[修改](#)，GaussDB服务会根据您设置的自动备份策略，自动创建数据库实例的备份。

## 手动备份

手动备份是由用户启动的数据库实例的全量备份，它会一直保存，直到用户手动删除。

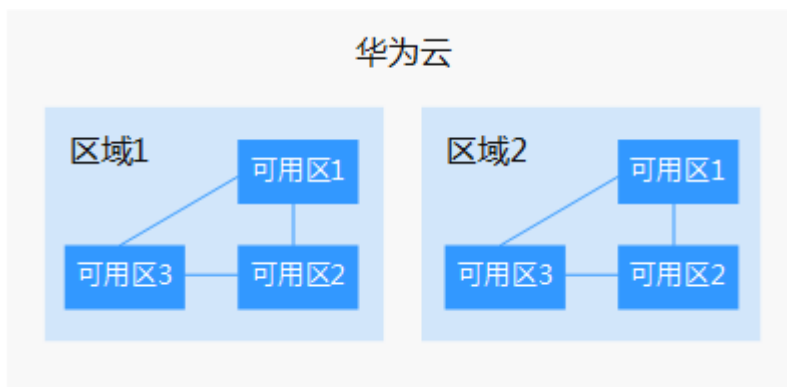
## 区域和可用区

区域和可用区用来描述数据中心的位置，您可以在特定的区域、可用区创建资源。

- 区域（Region）：从地理位置和网络时延维度划分，同一个Region内共享弹性计算、块存储、对象存储、VPC网络、弹性公网IP、镜像等公共服务。Region分为通用Region和专属Region，通用Region指面向公共租户提供通用云服务的Region；专属Region指只承载同一类业务或只面向特定租户提供业务服务的专用Region。
- 可用区（AZ，Availability Zone）：一个AZ是一个或多个物理数据中心的集合，有独立的风火水电，AZ内逻辑上再将计算、网络、存储等资源划分成多个实例。一个Region中的多个AZ间通过高速光纤相连，以满足用户跨AZ构建高可用性系统的需求。

图4-1阐明了区域和可用区之间的关系。

图 4-1 区域和可用区



目前，华为云已在全球多个地域开放云服务，您可以根据需求选择适合自己的区域和可用区。更多信息请参见[华为云全球站点](#)。

## 项目

Project用于将OpenStack的资源（计算、存储和网络资源）进行分组和隔离。Project可以是一个部门或者一个项目组。一个账户中可以创建多个Project。

# 5 产品优势

- **高安全**

GaussDB拥有TOP级的商业数据库安全特性，如下所示，能够满足政企和金融级客户的核心安全诉求。

数据动态脱敏，行级访问控制，密态计算。

- **健全的工具与服务化能力**

GaussDB已经拥有华为云，商用服务化部署能力，同时支持DAS、DRS等生态工具。有效保障用户开发、运维、优化、监控、迁移等日常工作需要。

- **全栈自研**

GaussDB基于鲲鹏生态，是当前国内唯一能够做到全栈自主可控的国产品牌。同时GaussDB能够基于硬件优势在底层不断进行优化，提升产品综合性能。

- **开源生态**

GaussDB已经支持开源社区，并提供主备版版本下载。



# 6 实例说明

## 6.1 数据库实例状态

### 数据库实例状态

数据库实例状态是数据库实例的运行情况。用户可以使用管理控制台和API操作查看数据库实例状态。

表 6-1 状态及说明

状态	说明
正常	数据库实例正常和可用。
异常	数据库实例不可用。
创建中	正在创建数据库实例。
创建失败	数据库实例创建失败。
重启中	数据库实例重启中。
启动中	数据库实例启动中。
节点启动中	数据库实例节点启动中。
停止中	数据库实例停止中。
节点停止中	数据库实例节点停止中。
已停止	数据库实例/节点已停止。
扩容中	数据库实例磁盘扩容中。
添加节点中	数据库实例节点扩容中。
备份中	正在备份数据库实例。
恢复中	正在恢复备份到实例中。

状态	说明
恢复失败	实例恢复失败。
冻结	账户余额小于或等于0元，系统对该用户下的实例进行冻结。您需前往费用中心充值成功，欠款核销后，冻结的实例才会解冻。
存储空间满	实例的磁盘空间已满，此时不可进行数据库写入操作，您需要扩容磁盘使实例恢复到正常状态。
已删除	数据库实例已被删除，对于已经删除的实例，将不会在实例列表中显示。
版本升级中	实例版本正在升级中。
参数变更，等待重启	数据库参数修改后，有些参数修改，需等待用户重启实例才能生效。
变更副本中	数据库实例正在执行降副本操作。

## 备份状态

表 6-2 备份状态及说明

状态	说明
备份完成	表明备份任务执行成功。
备份失败	表明备份任务执行失败。
备份中	表明正在进行备份中。

## 6.2 数据库实例规格

GaussDB的实例规格CPU架构分为X86架构和ARM架构，不同架构支持的区域请参见表6-3，支持的规格列表请参见表6-4。

- X86架构：独享型（1:4）、独享型（1:8）、通用型（1:4）
- ARM架构：鲲鹏独享型（1:4）、鲲鹏独享型（1:8）

表 6-3 支持的区域

部署架构	规格类型	计费模式	区域
分布式	独享型 (1:4) / 独享型 (1:8)	按需	亚太-新加坡
			华东-上海一
			华北-北京四

部署架构	规格类型	计费模式	区域			
			华南-广州			
			西南-贵阳一			
			非洲-约翰内斯堡			
			拉美-圣保罗一			
		包周期			亚太-新加坡	
					华东-上海一	
					华北-北京四	
					华南-广州	
					西南-贵阳一	
					非洲-约翰内斯堡	
					拉美-圣保罗一	
					鲲鹏独享型 (1:8)	按需
		华东-上海一				
		华北-北京四				
	华南-广州					
	西南-贵阳一					
	非洲-约翰内斯堡					
	拉美-圣保罗一					
	包周期			亚太-新加坡		
				华东-上海一		
				华北-北京四		
				华南-广州		
				西南-贵阳一		
	鲲鹏独享型 (1:4)	不区分		华东-上海一		
华北-北京四						
华南-广州						
西南-贵阳一						

部署架构	规格类型	计费模式	区域
	通用型 (1:4)	按需	亚太-新加坡
			华东-上海一
			华北-北京四
			华南-广州
			西南-贵阳一
			非洲-约翰内斯堡
			拉美-圣保罗一
		包周期	亚太-新加坡
			华东-上海一
			华北-北京四
			华南-广州
			西南-贵阳一
			非洲-约翰内斯堡
			拉美-圣保罗一
主备版	独享型 (1:4) / 独享型 (1:8)	按需	亚太-新加坡
			华东-上海一
			华北-北京四
			华南-广州
			西南-贵阳一
			非洲-约翰内斯堡
			拉美-圣保罗一
		包周期	亚太-新加坡
			华东-上海一
			华北-北京四
			华南-广州
			西南-贵阳一
			非洲-约翰内斯堡
			拉美-圣保罗一
通用型 (1:4)	按需	亚太-新加坡	
		华东-上海一	

部署架构	规格类型	计费模式	区域	
			华北-北京四	
			华南-广州	
			西南-贵阳一	
			非洲-约翰内斯堡	
			拉美-圣保罗一	
			包周期	亚太-新加坡
			华东-上海一	
			华北-北京四	
			华南-广州	
			西南-贵阳一	
		非洲-约翰内斯堡		
		拉美-圣保罗一		

表 6-4 支持的性能规格

产品类型	实例类型	规格	vCPU(个)	内存(GB)	默认最大连接数 (单个CN)
企业版	分布式	独享型 (1:4) <b>说明</b> 独享型 (1:4) 基于X86架构。	8	32	100
			16	64	1000
			32	128	2000
			64	256	4000
		独享型 (1:8) <b>说明</b> 独享型 (1:8) 基于X86架构。	8	64	1000
			16	128	2000
			32	256	4000
			64	512	8000
		鲲鹏独享型 (1:4) <b>说明</b> 鲲鹏独享型 (1:4) 基于ARM架构。	8	32	100
			16	64	1000
			32	128	2000
			64	256	4000

产品类型	实例类型	规格	vCPU(个)	内存(GB)	默认最大连接数 (单个CN)			
		鲲鹏独享型 (1:8)	8	64	1000			
		说明 鲲鹏独享型 (1:8)基于 ARM架构。	16	128	2000			
			32	256	4000			
			60	480	8000			
	主备版	独享型(1:4)	说明 独享型(1:4)基 于X86架构。	4	16	说明 仅3.208及以 上的主备版实 例支持。	300	
				8	32		600	
				16	64		2048	
				32	128		5000	
		独享型(1:8)	说明 独享型(1:8)基 于X86架构。	4	32	600		
				8	64	2048		
				16	128	5000		
				32	256	11000		
		64	512	25000				
					4	16	说明 仅3.208及以 上的主备版实 例支持。	300
					8	32		600
					16	64		2048
		32	128	5000				
		鲲鹏独享型 (1:4)	说明 鲲鹏独享型 (1:4)基于 ARM架构。	4	32	600		
				8	64	2048		
				16	128	5000		
	32			256	11000			
	60			480	24000			
	鲲鹏独享型 (1:8)	说明 鲲鹏独享型 (1:8)基于 ARM架构。	4	64	2048			
			8	128	5000			
16			256	11000				
32			512	24000				
60			1024	48000				
基础版	分布式	独享型(1:4)	说明 独享型(1:4)基 于X86架构。	16	64	200		
				32	128	350		
				64	256	900		

产品类型	实例类型	规格	vCPU(个)	内存(GB)	默认最大连接数 (单个CN)	
		独享型 (1:8) <b>说明</b> 独享型 (1:8) 基于X86架构。	16	128	350	
			32	256	900	
			64	512	2000	
		鲲鹏独享型 (1:4) <b>说明</b> 鲲鹏独享型 (1:4) 基于ARM架构。	16	64	200	
			32	128	350	
		鲲鹏独享型 (1:8) <b>说明</b> 鲲鹏独享型 (1:8) 基于ARM架构。	16	128	350	
			32	256	900	
			60	480	1800	
		主备版	通用型 (1:4) <b>说明</b> 通用型 (1:4) 基于X86架构。	4	16	300
				8	32	600
			独享型 (1:4) <b>说明</b> 独享型 (1:4) 基于X86架构。	4	16	300
				8	32	600
	16			64	2048	
	32			128	5000	
	64			256	11000	
	鲲鹏独享型 (1:4) <b>说明</b> 鲲鹏独享型 (1:4) 基于ARM架构。		4	16	300	
			8	32	600	
			16	64	2048	
			32	128	5000	

## 6.3 数据库实例存储类型

数据库系统通常是IT系统最为重要的系统，对存储IO性能要求高，GaussDB支持“超高IO”和“极速型SSD”存储类型。

- 超高IO：超高性能云硬盘，可用于企业关键性业务，适合高吞吐、低时延的工作负载。最大吞吐量为800MB/S。
- 极速型SSD：适用于需要超大带宽和超低时延的场景。最大吞吐量为2500MB/S。

 说明

极速型SSD存储类型仅针对特定用户开放，如需配置白名单权限，您可以在管理控制台右上角，选择“[工单 > 新建工单](#)”，提交开通白名单的申请。

## 6.4 数据库引擎版本说明

### GaussDB 数据库引擎版本号说明

GaussDB采用点分式版本号。格式为：A.BCD。其中：

- “A”：表示大版本，该版本一般会有架构升级和较大的新特性发布。
- “BCD”：首位（B）代表大的新特性发布版本，第二位（C）和第三位（D）代表迭代计划内含有重要的更新和提升。

### GaussDB 数据库和内核引擎版本对应关系

GaussDB版本的对应关系如[表6-5](#)所示。

GaussDB数据库引擎版本和内核引擎版本查询方法请参考[查看云数据库GaussDB数据库和内核引擎版本](#)。

表 6-5 GaussDB 数据库和内核引擎版本对应关系

GaussDB数据库引擎版本	GaussDB内核引擎版本
8.102.0	505.1.0
3.226.0	503.1.0.SPC2300
3.223.0	503.1.0.SPC2000
3.222.0	503.1.0.SPC1700
3.220.0	503.1.0.SPC1500
3.208.0	503.1.0.SPC1200
3.207.0	503.1.0.SPC1100
3.201.0	503.1.0.SPC0200

### 查看云数据库 GaussDB 数据库和内核引擎版本



1. [登录管理控制台](#)。
2. 单击管理控制台左上角的 ，选择区域和项目。
3. 在页面左上角单击 ，选择“数据库 > 云数据库 GaussDB”，进入云数据库 GaussDB信息页面。
4. 在“实例管理”页面，选择指定的实例，单击实例名称，进入基本信息页面。
5. 在实例“基本信息”页面，“数据库信息”模块的“数据库引擎版本”和“内核引擎版本”处查看数据库实例和内核的版本号。



图 6-1 基本信息



## 6.5 数据库实例类型

GaussDB的实例类型如表1所示。

表 6-6 实例类型

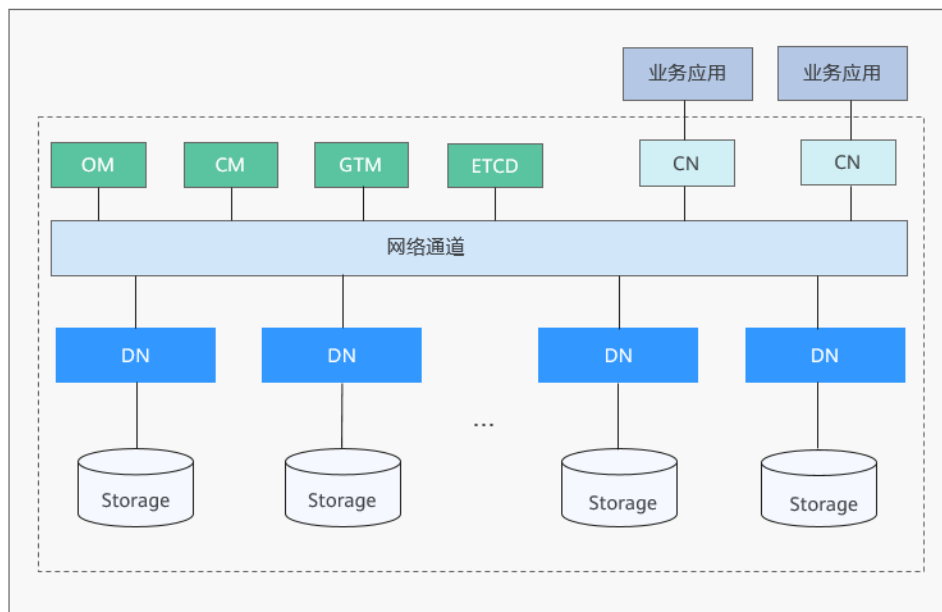
实例类型	支持的部署形态	是否支持实例扩容	适用场景	包含的组件	业务处理流程
分布式版	<p>独立部署：将数据库组件部署在不同节点上。适用于可靠性、稳定性要求较高，实例规模较大的场景。</p> <p>混合部署：采用一主两备三节点的部署模式，包含三个分片。</p>	是	数据量较大，对数据容量和并发能力有一定诉求。	<ul style="list-style-type: none"> <li>• O M</li> <li>• CM</li> <li>• GT M</li> <li>• ETC D</li> <li>• CN</li> <li>• DN</li> </ul>	业务应用下发 SQL 给 CN，CN 利用数据库的优化器生成执行计划，下发给 DN，每个 DN 会按照执行计划的要求去处理数据，处理完成后 DN 将结果集返回给 CN 进行汇总，最后 CN 将汇总后的结果返回给业务应用。
主备版	<p>高可用（1主2备）：采用一主两备三节点的部署模式，包含一个分片。</p> <p>单副本：采用单节点的部署模式，仅包含一个 CMS 和一个 DN 组件，不适用于生产环境。单副本实例仅支持 2.2 及以上版本实例。由于部署在单台机器上，因此无法保障可用性（SLA）。</p> <p>1主1备1日志：采用一主一备一日志三节点的部署模式，包含一个分片。</p>	否	数据量较小，且长期来看数据不会大幅度增长，但是对数据的可靠性，以及业务的可用性有一定诉求。	<ul style="list-style-type: none"> <li>• O M</li> <li>• CM</li> <li>• ETC D</li> <li>• DN</li> </ul>	业务应用直接下发任务给 DN，DN 处理完成后再将结果返回给业务应用。

### 📖 说明

- OM: 运维管理模块 (Operation Manager)。提供集群日常运维、配置管理的管理接口、工具。
- CM: 集群管理模块 (Cluster Manager)。管理和监控分布式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。
- GTM: 全局事务管理器 (Global Transaction Manager), 负责生成和维护全局事务ID、事务快照、时间戳、sequence信息等全局唯一的信息。
- ETCD: 分布式键值存储系统 (Editable Text Configuration Daemon)。用于共享配置和服务发现 (服务注册和查找)。
- CN: 协调节点 (Coordinator Node)。负责接收来自应用的访问请求, 并向客户端返回执行结果; 负责分解任务, 并调度任务分片在各DN上并行执行。
- DN: 数据节点 (Data Node)。负责存储业务数据、执行数据查询任务以及返回执行结果。

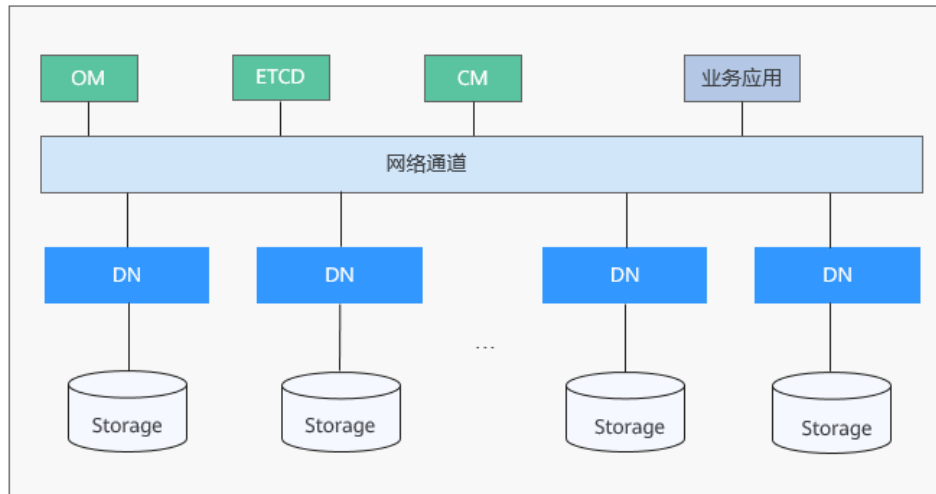
GaussDB分布式版逻辑架构图如下:

图 6-2 分布式版逻辑架构图



GaussDB主备版逻辑架构图如下:

图 6-3 主备版逻辑架构图



## 6.6 数据库实例参数

本章节介绍 GaussDB 数据库实例支持修改的实例参数。

- 3.x 支持修改的参数。
  - 分布式参数。
  - 主备版参数。
- 2.x 支持修改的参数。
  - 分布式参数。
  - 主备版参数。

### 3.x 支持修改的参数：

支持修改的参数如下：

表 6-7 分布式参数

参数名称	描述
audit_system_object	该参数决定是否对 GaussDB Kernel 数据库对象的 CREATE、DROP、ALTER 操作进行审计。GaussDB Kernel 数据库对象包括 DATABASE、USER、schema、TABLE 等。通过修改该配置参数的值，可以只审计需要的数据库对象的操作，在主备强制选主场景建议 audit_system_object 取最大值，所有 DDL 对象全部审计，不当修改该参数会导致丢失 DDL 审计日志，请在技术支持人员指导下进行修改。
autoanalyze	标识是否允许在生成计划的时候，对于没有统计信息的表进行统计信息自动收集。
autoanalyze_timeout	设置 autoanalyze 的超时时间。在对某张表做 autoanalyze 时，如果该表的 analyze 时长超过了 autoanalyze_timeout，则自动取消该表此次 analyze。0 表示不超时，单位为秒。

参数名称	描述
cn:effective_cache_size	设置CN节点优化器在一次单一的查询中可用的磁盘缓冲区的有效大小，单位为8KB。
cn:enable_hotkeys_collection	是否开启对数据库内的被访问的键值进行统计。
cn:track_stmt_session_slot	设置CN一个session缓存的最大的全量/慢SQL的数量。
datestyle	设置日期和时间值的显示格式。
dn:effective_cache_size	设置DN节点优化器在一次单一的查询中可用的磁盘缓冲区的有效大小，单位为8KB。
dn:enable_hotkeys_collection	是否开启对数据库内的被访问的键值进行统计。
dn:track_stmt_session_slot	设置DN一个session缓存的最大的全量/慢SQL的数量。
enable_seqscan	控制优化器对顺序扫描规划类型的使用。完全消除顺序扫描是不可能的，但是关闭这个变量会让优化器在存在其他方法的时候优先选择其他方法。
enable_slot_log	是否开启逻辑复制槽主备同步特性。
enable_stream_operator	控制优化器对stream的使用。当该参数关闭时，可能会有大量关于计划不能下推的日志记录到日志文件中。
failed_login_attempts	在任意时候，如果输入密码错误的次数达到failed_login_attempts则当前账户被锁定，password_lock_time秒后被自动解锁，仅sysadmin用户可以访问。
log_min_duration_statement	当某条语句的持续时间大于或者等于特定的毫秒数时，记录每条完成语句的持续时间。设置过低的阈值可能影响负载吞吐，-1表示关闭此功能，单位是毫秒。
max_replication_slots	设置主机端的日志复制slot个数。
max_wal_senders	备DN连接主DN获取物理日志，逻辑复制工具连接主DN获取逻辑日志都会占用walsender线程，此参数标识最大允许创建walsender线程的个数。
password_effect_time	该字段决定账户密码的有效时间，单位为天。
password_lock_time	设置账户被锁定后的自动解锁时间，单位为天。
recovery_time_target	设置recovery_time_target秒能够让备机完成日志写入和回放。单位为秒。
session_timeout	表明与服务器建立链接后，不进行任何操作的最长时间。0表示关闭超时设置，单位为秒。

参数名称	描述
timezone	设置显示和解释时间类型数值时使用的时区。
track_stmt_stat_level	控制语句执行跟踪的级别。
update_lockwait_timeout	允许并发更新参数开启情况下，该参数控制并发更新同一行时单个锁的最长等待时间。当申请的锁等待时间超过设定值时，系统会报错，单位是毫秒。
wal_level	设置写入WAL信息量的级别，不能为空或被注释掉。此参数设置为logical后才允许进行逻辑日志抽取，此参数设置为logical后xlog日志中会额外记录主键信息。
cn:audit_thread_num	审计线程的个数，取值范围 1~48。
dn:audit_thread_num	审计线程的个数，取值范围 1~48。
cn:qrw_inlist2join_optmode	控制是否使用inlist-to-join查询重写。
dn:qrw_inlist2join_optmode	控制是否使用inlist-to-join查询重写。
cn:audit_xid_info	是否在审计日志字段detail_info中记录SQL语句的事务ID,0表示关闭,1表示开启。
dn:audit_xid_info	是否在审计日志字段detail_info中记录SQL语句的事务ID,0表示关闭,1表示开启。
cn:default_limit_rows	设置生成genericplan的缺省limit估算行数。此参数设置为正数时意为直接将设置的值作为估算limit的行数，为负数时代表使用百分比的形式设置默认的估算值，负数转换为默认百分比，即-5代表5%。
dn:default_limit_rows	设置生成genericplan的缺省limit估算行数。此参数设置为正数时意为直接将设置的值作为估算limit的行数，为负数时代表使用百分比的形式设置默认的估算值，负数转换为默认百分比，即-5代表5%。
cn:audit_dml_state_select	是否对SELECT操作进行审计。
dn:audit_dml_state_select	是否对SELECT操作进行审计。
cn:audit_dml_state	是否对具体表的INSERT、UPDATE、DELETE操作进行审计,0表示关闭,1表示开启。
dn:audit_dml_state	是否对具体表的INSERT、UPDATE、DELETE操作进行审计,0表示关闭,1表示开启。
cn:random_page_cost	设置优化器计算一次非顺序抓取磁盘页面的开销。

参数名称	描述
dn:random_page_cost	设置优化器计算一次非顺序抓取磁盘页面的开销。
cn:enable_security_policy	安全策略开关，控制统一审计和数据动态脱敏策略是否生效。
dn:enable_security_policy	安全策略开关，控制统一审计和数据动态脱敏策略是否生效。
cn:audit_set_parameter	是否对SET操作进行审计,0表示关闭,1表示开启。
dn:audit_set_parameter	是否对SET操作进行审计,0表示关闭,1表示开启。
cn:enable_pbe_optimization	设置优化器是否对以PBE ( Parse Bind Execute ) 形式执行的语句进行查询计划的优化。
dn:enable_pbe_optimization	设置优化器是否对以PBE ( Parse Bind Execute ) 形式执行的语句进行查询计划的优化。
wdr_snapshot_interval	后台Snapshot自动对数据库监控数据执行快照操作的时间间隔，单位分钟。
cn:enable_auto_explain	控制是否开启自动打印执行计划。该参数是用来定位慢存储过程或慢查询。
dn:enable_auto_explain	控制是否开启自动打印执行计划。该参数是用来定位慢存储过程或慢查询。
enable_wdr_snapshot	是否开启WDR 性能快照，开启时内核将定期进行性能视图快照。
cn:max_concurrent_autonomous_transactions	自治事务最大连接数，同一时间自治事务执行的最大并发数。当设置为0时，将无法执行自治事务。
dn:max_concurrent_autonomous_transactions	自治事务最大连接数，同一时间自治事务执行的最大并发数。当设置为0时，将无法执行自治事务。
cn:max_standby_archive_delay	当开启双机热备模式时，如果备机正处理归档WAL日志数据，此时进行查询会产生冲突，此参数就是设置备机取消查询之前所等待的时间，单位为ms。
dn:max_standby_archive_delay	当开启双机热备模式时，如果备机正处理归档WAL日志数据，此时进行查询会产生冲突，此参数就是设置备机取消查询之前所等待的时间，单位为ms。
cn:max_standby_streaming_delay	设置备机取消查询之前等待的时间，单位为ms。

参数名称	描述
dn:max_standby_streaming_delay	设置备机取消查询之前等待的时间，单位为ms。
cn:recovery_max_workers	并行回放线程个数。
dn:recovery_max_workers	并行回放线程个数。
cn:auto_explain_log_min_duration	控制自动打印执行计划的耗时阈值，整体耗时大于auto_explain_log_min_duration的执行计划才会被打印。单位为ms。
dn:auto_explain_log_min_duration	控制自动打印执行计划的耗时阈值，整体耗时大于auto_explain_log_min_duration的执行计划才会被打印。单位为ms。
cn:audit_function_exec	这个参数决定在执行存储过程、匿名块或自定义函数（不包括系统自带函数）时是否记录审计信息。
dn:audit_function_exec	这个参数决定在执行存储过程、匿名块或自定义函数（不包括系统自带函数）时是否记录审计信息。
cn:local_syscache_threshold	系统表cache在单个session缓存的大小。单位为KB。
dn:local_syscache_threshold	系统表cache在单个session缓存的大小。单位为KB。
cms:datastorage_threshold_value_check	设置磁盘占用率只读检测阈值，当数据目录所在磁盘占用超过这个阈值，自动将数据库节点设置为只读模式，单位为百分比。
wdr_snapshot_retention_days	系统中数据库监控快照数据的保留天数，单位天。
cn:enable_default_ustore_table	指定是否开启默认支持Ustore存储引擎，该参数为on时，创建的表类型都为Ustore表。
dn:enable_default_ustore_table	指定是否开启默认支持Ustore存储引擎，该参数为on时，创建的表类型都为Ustore表。
cn:undo_space_limit_size	用于控制undo强制回收阈值，达到阈值的80%启动强制回收，单位为8kb。建议不小于undo_limit_size_per_transaction的值。
dn:undo_space_limit_size	用于控制undo强制回收阈值，达到阈值的80%启动强制回收，单位为8kb。建议不小于undo_limit_size_per_transaction的值。

参数名称	描述
cn:undo_limit_size_per_transaction	用于控制单事务undo分配空间阈值，达到阈值时事务报错回滚，单位为8kb。建议undo_limit_size_per_transaction的值小于undo_space_limit_size的值，若大于undo_space_limit_size的值，可以设置成功，但是在使用时取两个参数的最小值作为实际的单事务undo分配空间阈值。
dn:undo_limit_size_per_transaction	用于控制单事务undo分配空间阈值，达到阈值时事务报错回滚，单位为8kb。建议undo_limit_size_per_transaction的值小于undo_space_limit_size的值，若大于undo_space_limit_size的值，可以设置成功，但是在使用时取两个参数的最小值作为实际的单事务undo分配空间阈值。
cn:enable_recyclebin	用来控制回收站的实时打开和关闭。
dn:enable_recyclebin	用来控制回收站的实时打开和关闭。
cn:recyclebin_retention_time	设置回收站对象保留时间，超过该时间的回收站对象将被自动清理，单位为秒。
dn:recyclebin_retention_time	设置回收站对象保留时间，超过该时间的回收站对象将被自动清理，单位为秒。
cn:undo_retention_time	该参数用于闪回查询功能，设置undo日志保留时间，单位为秒。提示：1.本地磁盘undo空间会变大；2.后续增量备份中备份集大小会变大，保留了额外的undo内容。
dn:undo_retention_time	该参数用于闪回查询功能，设置undo日志保留时间，单位为秒。提示：1.本地磁盘undo空间会变大；2.后续增量备份中备份集大小会变大，保留了额外的undo内容。
cn:cost_model_version	此参数用来指定优化器代价模型的版本。可以视作一个保护参数，用来禁用最新的优化器代价模型，保持和旧版本计划一致。改变此参数，可能会导致很多SQL计划的改变。因此修改前请谨慎评估。
dn:cost_model_version	此参数用来指定优化器代价模型的版本。可以视作一个保护参数，用来禁用最新的优化器代价模型，保持和旧版本计划一致。改变此参数，可能会导致很多SQL计划的改变。因此修改前请谨慎评估。
cn:enable_dynamic_sample_size	是否动态调整采样行数。对于超过一百万行的大表，收集统计信息时动态调整采样行数，提高统计信息准确性。
dn:enable_dynamic_sample_size	是否动态调整采样行数。对于超过一百万行的大表，收集统计信息时动态调整采样行数，提高统计信息准确性。
cn:resilience_ctrlslot_available_maxpercent	最多允许慢SQL占用线程池的的线程占用百分比。仅对非sysadmin/monitoradmin用户执行的select类型的语句生效。



参数名称	描述
dn:resilience_ct rlslot_available _maxpercent	最多允许慢SQL占用线程池的的线程占用百分比。仅对非sysadmin/monitoradmin用户执行的select类型的语句生效。
cn:resilience_ct rlstmt_control_ iopslimit	正常SQL语句被标记为慢SQL后，慢SQL可使用的最大IOPS上限。仅对非sysadmin/monitoradmin用户执行的select类型的语句生效。0(None)：表示不对IOPS做限制。10(Low)：表示对IOPS限制等级为Low。20(Medium)：表示对IOPS限制等级为Medium。50(High)：表示对IOPS限制等级为High。
dn:resilience_ct rlstmt_control_ iopslimit	正常SQL语句被标记为慢SQL后，慢SQL可使用的最大IOPS上限。仅对非sysadmin/monitoradmin用户执行的select类型的语句生效。0(None)：表示不对IOPS做限制。10(Low)：表示对IOPS限制等级为Low。20(Medium)：表示对IOPS限制等级为Medium。50(High)：表示对IOPS限制等级为High。
dn:resilience_ct rlstmt_detect_t imelimit	正常SQL语句被标记为慢SQL的执行时间，0不做慢SQL识别，大于0表示执行的SQL执行时间超过该时间后备标记为慢SQL。仅对非sysadmin/monitoradmin用户执行的select类型的语句生效。单位为ms。

表 6-8 主备版参数

参数名称	描述
audit_system_ object	该参数决定是否对GaussDB Kernel数据库对象的CREATE、DROP、ALTER操作进行审计。GaussDB Kernel数据库对象包括DATABASE、USER、schema、TABLE等。通过修改该配置参数的值，可以只审计需要的数据库对象的操作，在主备强制选主场景建议audit_system_object取最大值，所有DDL对象全部审计，不当修改该参数会导致丢失DDL审计日志，请在技术支持人员指导下进行修改。
autoanalyze	标识是否允许在生成计划的时候，对于没有统计信息的表进行统计信息自动收集。
autoanalyze_ti meout	设置autoanalyze的超时时间。在对某张表做autoanalyze时，如果该表的analyze时长超过了autoanalyze_timeout，则自动取消该表此次analyze。0表示不超时，单位为秒。
datestyle	设置日期和时间值的显示格式。
dn:wal_keep_s egments	“pg_xlog”目录下保留事务日志文件的最小数目。备机通过获取主机此处的日志进行流复制。
enable_seqsc an	控制优化器对顺序扫描规划类型的使用。完全消除顺序扫描是不可能的，但是关闭这个变量会让优化器在存在其他方法的时候优先选择其他方法。
enable_slot_lo g	是否开启逻辑复制槽主备同步特性。

参数名称	描述
failed_login_attempts	在任意时候，如果输入密码错误的次数达到failed_login_attempts则当前账户被锁定，password_lock_time秒后被自动解锁，仅sysadmin用户可以访问。
log_min_duration_statement	当某条语句的持续时间大于或者等于特定的毫秒数时，记录每条完成语句的持续时间。设置过低的阈值可能影响负载吞吐，-1表示关闭此功能，单位是毫秒。
max_replication_slots	设置主机端的日志复制slot个数。
max_wal_senders	备DN连接主DN获取物理日志，逻辑复制工具连接主DN获取逻辑日志都会占用walsender线程，此参数标识最大允许创建walsender线程的个数。
password_effect_time	该字段决定账户密码的有效时间，单位为天。
password_lock_time	设置账户被锁定后的自动解锁时间，单位为天。
session_timeout	表明与服务器建立链接后，不进行任何操作的最长时间。0表示关闭超时设置，单位为秒。
timezone	设置显示和解释时间类型数值时使用的时区。
update_lockwait_timeout	允许并发更新参数开启情况下，该参数控制并发更新同一行时单个锁的最长等待时间。当申请的锁等待时间超过设定值时，系统会报错，单位是毫秒。
wal_level	设置写入WAL信息量的级别，不能为空或被注释掉。此参数设置为logical后才允许进行逻辑日志抽取，此参数设置为logical后xlog日志中会额外记录主键信息。
dn:audit_thread_num	审计线程的个数,取值范围 1~48。
dn:qrw_inlist2join_optmode	控制是否使用inlist-to-join查询重写。
dn:audit_xid_info	是否在审计日志字段detail_info中记录SQL语句的事务ID,0表示关闭,1表示开启。
dn:default_limit_rows	设置生成genericplan的缺省limit估算行数。此参数设置为正数时意为直接将设置的值作为估算limit的行数，为负数时代表使用百分比的形式设置默认的估算值，负数转换为默认百分比，即-5代表5%。
dn:audit_dml_state_select	是否对SELECT操作进行审计。
dn:audit_dml_state	是否对具体表的INSERT、UPDATE、DELETE操作进行审计,0表示关闭,1表示开启。

参数名称	描述
dn:random_page_cost	设置优化器计算一次非顺序抓取磁盘页面的开销。
dn:enable_security_policy	安全策略开关，控制统一审计和数据动态脱敏策略是否生效。
dn:audit_set_parameter	是否对SET操作进行审计,0表示关闭,1表示开启。
dn:max_standby_streaming_delay	设置备机取消查询之前等待的时间，单位为ms。
dn:vacuum_defer_cleanup_age	指定VACUUM使用的事务数。
dn:enable_pbe_optimization	设置优化器是否对以PBE（Parse Bind Execute）形式执行的语句进行查询计划的优化。
wdr_snapshot_interval	后台Snapshot自动对数据库监控数据执行快照操作的时间间隔，单位分钟。
undo_retention_time	该参数用于闪回查询功能，设置undo日志保留时间，单位为秒。 提示：1.本地磁盘undo空间会变大；2.后续增量备份中备份集大小会变大，保留了额外的undo内容。
track_stmt_stat_level	控制语句执行跟踪的级别。
dn:enable_auto_explain	控制是否开启自动打印执行计划。该参数是用来定位慢存储过程或慢查询。
enable_wdr_snapshot	是否开启WDR 性能快照，开启时内核将定期进行性能视图快照。
dn:max_concurrent_autonomous_transactions	自治事务最大连接数，同一时间自治事务执行的最大并发数。当设置为0时，将无法执行自治事务。
dn:max_standby_archive_delay	当开启双机热备模式时，如果备机正处理归档WAL日志数据，此时进行查询会产生冲突，此参数就是设置备机取消查询之前所等待的时间，单位为ms。
dn:max_standby_streaming_delay	设置备机取消查询之前等待的时间，单位为ms。
dn:recovery_max_workers	并行回放线程个数。
dn:auto_explain_log_min_duration	控制自动打印执行计划的耗时阈值，整体耗时大于auto_explain_log_min_duration的执行计划才会被打印。单位为ms。

参数名称	描述
dn:recovery_time_target	设置recovery_time_target秒能够让备机完成日志写入和回放。单位为秒。
dn:audit_function_exec	这个参数决定在执行存储过程、匿名块或自定义函数（不包括系统自带函数）时是否记录审计信息。
dn:local_syscache_threshold	系统表cache在单个session缓存的大小。单位为KB。
cms:datastorage_threshold_value_check	设置磁盘占用率只读检测阈值，当数据目录所在磁盘占用超过这个阈值，自动将数据库节点设置为只读模式，单位为百分比。
wdr_snapshot_retention_days	系统中数据库监控快照数据的保留天数，单位天。
dn:undo_space_limit_size	用于控制undo强制回收阈值，达到阈值的80%启动强制回收，单位为8kb。建议不小于undo_limit_size_per_transaction的值。
dn:group_concat_max_len	函数GROUP_CONCAT()结果的最大长度。
dn:enable_extension	控制是否支持创建数据库扩展插件。扩展插件属于实验室特性，不推荐使用。
dn:cost_model_version	此参数用来指定优化器代价模型的版本。可以视作一个保护参数，用来禁用最新的优化器代价模型，保持和旧版本计划一致。改变此参数，可能会导致很多SQL计划的改变。因此修改前请谨慎评估。
dn:immediate_analyze_threshold	插入数据后自动做analyze的阈值。当新增数据量达到原有数据量的immediate_analyze_threshold倍，且总行数超过一百时，会自动触发一次analyze。
dn:enable_dynamic_sample_size	是否动态调整采样行数。对于超过一百万行的大表，收集统计信息时动态调整采样行数，提高统计信息准确性。
dn:max_io_capacity	设置后端写进程批量刷页每秒的IO上限，需要根据具体业务场景和机器磁盘IO能力进行设置，要求RTO很短时间或者数据量比共享内存大多倍的情况，业务访问数据量又是随机访问时，该值不宜过小。设置较小的max_io_capacity会减小后端写进程刷页个数，如果业务触发页面淘汰多时，该值设置小会影响业务，单位为KB。
dn:max_connections	设置DN允许连接的最大并发连接数。
log_autovacuum_min_duration	设置自动清理时间阈值，当自动清理的执行时间大于或者等于某个特定的阈值，向服务器日志中记录自动清理执行的每一步操作。当参数设置为0时，表示所有的自动清理操作都记录到日志中。当参数设置为-1时，表示所有的自动清理操作都不记录到日志中。

## 2.x 支持修改的参数：

支持修改的参数如下：

**表 6-9** 分布式参数

参数名称	描述
audit_system_object	该参数决定是否对GaussDB Kernel数据库对象的CREATE、DROP、ALTER操作进行审计。GaussDB Kernel数据库对象包括DATABASE、USER、schema、TABLE等。通过修改该配置参数的值，可以只审计需要的数据库对象的操作，在主备强制选主场景建议audit_system_object取最大值，所有DDL对象全部审计，不当修改该参数会导致丢失DDL审计日志，请在技术支持人员指导下进行修改。
autoanalyze	标识是否允许在生成计划的时候，对于没有统计信息的表进行统计信息自动收集。
autoanalyze_timeout	设置autoanalyze的超时时间。在对某张表做autoanalyze时，如果该表的analyze时长超过了autoanalyze_timeout，则自动取消该表此次analyze。0表示不超时，单位为秒。
cn:effective_cache_size	设置CN节点优化器在一次单一的查询中可用的磁盘缓冲区的有效大小，单位为8KB。
cn:enable_hotkeys_collection	是否开启对数据库内的被访问的键值进行统计。
cn:track_stmt_session_slot	设置CN一个session缓存的最大的全量/慢SQL的数量。
datestyle	设置日期和时间值的显示格式。
dn:effective_cache_size	设置DN节点优化器在一次单一的查询中可用的磁盘缓冲区的有效大小，单位为8KB。
dn:enable_hotkeys_collection	是否开启对数据库内的被访问的键值进行统计。
dn:track_stmt_session_slot	设置DN一个session缓存的最大的全量/慢SQL的数量。
enable_seqscan	控制优化器对顺序扫描规划类型的使用。完全消除顺序扫描是不可能的，但是关闭这个变量会让优化器在存在其他方法的时候优先选择其他方法。
enable_slot_log	是否开启逻辑复制槽主备同步特性。
enable_stream_operator	控制优化器对stream的使用。当该参数关闭时，可能会有大量关于计划不能下推的日志记录到日志文件中。
failed_login_attempts	在任意时候，如果输入密码错误的次数达到failed_login_attempts则当前账户被锁定，password_lock_time秒后被自动解锁，仅sysadmin用户可以访问。

参数名称	描述
log_min_duration_statement	当某条语句的持续时间大于或者等于特定的毫秒数时，记录每条完成语句的持续时间。设置过低的阈值可能影响负载吞吐，-1表示关闭此功能，单位是毫秒。
max_replication_slots	设置主机端的日志复制slot个数。
max_wal_senders	备DN连接主DN获取物理日志，逻辑复制工具连接主DN获取逻辑日志都会占用walsender线程，此参数标识最大允许创建walsender线程的个数。
password_effect_time	该字段决定账户密码的有效时间，单位为天。
password_lock_time	设置账户被锁定后的自动解锁时间，单位为天。
recovery_time_target	设置recovery_time_target秒能够让备机完成日志写入和回放。单位为秒。
session_timeout	表明与服务器建立链接后，不进行任何操作的最长时间。0表示关闭超时设置，单位为秒。
timezone	设置显示和解释时间类型数值时使用的时区。
track_stmt_stat_level	控制语句执行跟踪的级别。
update_lockwait_timeout	允许并发更新参数开启情况下，该参数控制并发更新同一行时单个锁的最长等待时间。当申请的锁等待时间超过设定值时，系统会报错，单位是毫秒。
wal_level	设置写入WAL信息量的级别，不能为空或被注释掉。此参数设置为logical后才允许进行逻辑日志抽取，此参数设置为logical后xlog日志中会额外记录主键信息。
cn:audit_thread_num	审计线程的个数,取值范围 1~48。
dn:audit_thread_num	审计线程的个数,取值范围 1~48。
cn:qrw_inlist2join_optmode	控制是否使用inlist-to-join查询重写。
dn:qrw_inlist2join_optmode	控制是否使用inlist-to-join查询重写。
cn:audit_xid_info	是否在审计日志字段detail_info中记录SQL语句的事务ID,0表示关闭,1表示开启。
dn:audit_xid_info	是否在审计日志字段detail_info中记录SQL语句的事务ID,0表示关闭,1表示开启。

参数名称	描述
cn:default_limit_rows	设置生成genericplan的缺省limit估算行数。此参数设置为正数时意为直接将设置的值作为估算limit的行数，为负数时代表使用百分比的形式设置默认的估算值，负数转换为默认百分比，即-5代表5%。
dn:default_limit_rows	设置生成genericplan的缺省limit估算行数。此参数设置为正数时意为直接将设置的值作为估算limit的行数，为负数时代表使用百分比的形式设置默认的估算值，负数转换为默认百分比，即-5代表5%。
cn:audit_dml_state_select	是否对SELECT操作进行审计。
dn:audit_dml_state_select	是否对SELECT操作进行审计。
cn:audit_dml_state	是否对具体表的INSERT、UPDATE、DELETE操作进行审计,0表示关闭,1表示开启。
dn:audit_dml_state	是否对具体表的INSERT、UPDATE、DELETE操作进行审计,0表示关闭,1表示开启。
cn:random_page_cost	设置优化器计算一次非顺序抓取磁盘页面的开销。
dn:random_page_cost	设置优化器计算一次非顺序抓取磁盘页面的开销。
cn:enable_security_policy	安全策略开关，控制统一审计和数据动态脱敏策略是否生效。
dn:enable_security_policy	安全策略开关，控制统一审计和数据动态脱敏策略是否生效。
cn:audit_set_parameter	是否对SET操作进行审计,0表示关闭,1表示开启。
dn:audit_set_parameter	是否对SET操作进行审计,0表示关闭,1表示开启。
cn:enable_pbe_optimization	设置优化器是否对以PBE ( Parse Bind Execute ) 形式执行的语句进行查询计划的优化。
dn:enable_pbe_optimization	设置优化器是否对以PBE ( Parse Bind Execute ) 形式执行的语句进行查询计划的优化。
wdr_snapshot_interval	后台Snapshot自动对数据库监控数据执行快照操作的时间间隔，单位分钟。
enable_wdr_snapshot	是否开启WDR 性能快照，开启时内核将定期进行性能视图快照。
cn:max_standby_archive_delay	当开启双机热备模式时，如果备机正处理归档WAL日志数据，此时进行查询会产生冲突，此参数就是设置备机取消查询之前所等待的时间，单位为ms。

参数名称	描述
dn:max_standby_archive_delay	当开启双机热备模式时，如果备机正处理归档WAL日志数据，此时进行查询会产生冲突，此参数就是设置备机取消查询之前所等待的时间，单位为ms。
cn:max_standby_streaming_delay	设置备机取消查询之前等待的时间，单位为ms。
dn:max_standby_streaming_delay	设置备机取消查询之前等待的时间，单位为ms。
cn:recovery_max_workers	并行回放线程个数。
dn:recovery_max_workers	并行回放线程个数。
cn:local_syscache_threshold	系统表cache在单个session缓存的大小。单位为KB。
dn:local_syscache_threshold	系统表cache在单个session缓存的大小。单位为KB。
cms:storage_threshold_value_check	设置磁盘占用率只读检测阈值，当数据目录所在磁盘占用超过这个阈值，自动将数据库节点设置为只读模式，单位为百分比。
wdr_snapshot_retention_days	系统中数据库监控快照数据的保留天数，单位天。

表 6-10 主备版参数

参数名称	描述
audit_system_object	该参数决定是否对GaussDB Kernel数据库对象的CREATE、DROP、ALTER操作进行审计。GaussDB Kernel数据库对象包括DATABASE、USER、schema、TABLE等。通过修改该配置参数的值，可以只审计需要的数据库对象的操作，在主备强制选主场景建议audit_system_object取最大值，所有DDL对象全部审计，不当修改该参数会导致丢失DDL审计日志，请在技术支持人员指导下进行修改。
autoanalyze	标识是否允许在生成计划的时候，对于没有统计信息的表进行统计信息自动收集。
autoanalyze_timeout	设置autoanalyze的超时时间。在对某张表做autoanalyze时，如果该表的analyze时长超过了autoanalyze_timeout，则自动取消该表此次analyze。0表示不超时，单位为秒。
datestyle	设置日期和时间值的显示格式。



参数名称	描述
dn:wal_keep_segments	“pg_xlog”目录下保留事务日志文件的最小数目。备机通过获取主机此处的日志进行流复制。
enable_seqscan	控制优化器对顺序扫描规划类型的使用。完全消除顺序扫描是不可能的，但是关闭这个变量会让优化器在存在其他方法的时候优先选择其他方法。
enable_slot_log	是否开启逻辑复制槽主备同步特性。
failed_login_attempts	在任意时候，如果输入密码错误的次数达到failed_login_attempts则当前账户被锁定，password_lock_time秒后被自动解锁，仅sysadmin用户可以访问。
log_min_duration_statement	当某条语句的持续时间大于或者等于特定的毫秒数时，记录每条完成语句的持续时间。设置过低的阈值可能影响负载吞吐，-1表示关闭此功能，单位是毫秒。
max_replication_slots	设置主机端的日志复制slot个数。
max_wal_senders	备DN连接主DN获取物理日志，逻辑复制工具连接主DN获取逻辑日志都会占用walsender线程，此参数标识最大允许创建walsender线程的个数。
password_effect_time	该字段决定账户密码的有效时间，单位为天。
password_lock_time	设置账户被锁定后的自动解锁时间，单位为天。
session_timeout	表明与服务器建立链接后，不进行任何操作的最长时间。0表示关闭超时设置，单位为秒。
timezone	设置显示和解释时间类型数值时使用的时区。
update_lockwait_timeout	允许并发更新参数开启情况下，该参数控制并发更新同一行时单个锁的最长等待时间。当申请的锁等待时间超过设定值时，系统会报错，单位是毫秒。
wal_level	设置写入WAL信息量的级别，不能为空或被注释掉。此参数设置为logical后才允许进行逻辑日志抽取，此参数设置为logical后xlog日志中会额外记录主键信息。
dn:audit_thread_num	审计线程的个数,取值范围 1~48。
dn:qrw_inlist2join_optmode	控制是否使用inlist-to-join查询重写。
dn:audit_xid_info	是否在审计日志字段detail_info中记录SQL语句的事务ID,0表示关闭,1表示开启。

参数名称	描述
dn:default_limit_rows	设置生成genericplan的缺省limit估算行数。此参数设置为正数时意为直接将设置的值作为估算limit的行数，为负数时代表使用百分比的形式设置默认的估算值，负数转换为默认百分比，即-5代表5%。
dn:audit_dml_state_select	是否对SELECT操作进行审计。
dn:audit_dml_state	是否对具体表的INSERT、UPDATE、DELETE操作进行审计,0表示关闭,1表示开启。
dn:random_page_cost	设置优化器计算一次非顺序抓取磁盘页面的开销。
dn:enable_security_policy	安全策略开关，控制统一审计和数据动态脱敏策略是否生效。
dn:audit_set_parameter	是否对SET操作进行审计,0表示关闭,1表示开启。
dn:max_standby_streaming_delay	设置备机取消查询之前等待的时间，单位为ms。
dn:vacuum_defer_cleanup_age	指定VACUUM使用的事务数。
dn:enable_pbe_optimization	设置优化器是否对以PBE（Parse Bind Execute）形式执行的语句进行查询计划的优化。
wdr_snapshot_interval	后台Snapshot自动对数据库监控数据执行快照操作的时间间隔，单位分钟。
undo_retention_time	该参数用于闪回查询功能，设置undo日志保留时间，单位为秒。 提示：1.本地磁盘undo空间会变大；2.后续增量备份中备份集大小会变大，保留了额外的undo内容。
track_stmt_stat_level	控制语句执行跟踪的级别。
enable_wdr_snapshot	是否开启WDR 性能快照，开启时内核将定期进行性能视图快照。
dn:max_standby_archive_delay	当开启双机热备模式时，如果备机正处理归档WAL日志数据，此时进行查询会产生冲突，此参数就是设置备机取消查询之前所等待的时间，单位为ms。
dn:max_standby_streaming_delay	设置备机取消查询之前等待的时间，单位为ms。
dn:recovery_max_workers	并行回放线程个数。

参数名称	描述
dn:recovery_time_target	设置recovery_time_target秒能够让备机完成日志写入和回放。单位为秒。
dn:local_syscache_threshold	系统表cache在单个session缓存的大小。单位为KB。
cms:datastorage_threshold_value_check	设置磁盘占用率只读检测阈值，当数据目录所在磁盘占用超过这个阈值，自动将数据库节点设置为只读模式，单位为百分比。
wdr_snapshot_retention_days	系统中数据库监控快照数据的保留天数，单位天。
log_autovacuum_min_duration	设置自动清理时间阈值，当自动清理的执行时间大于或者等于某个特定的阈值，向服务器日志中记录自动清理执行的每一步操作。当参数设置为0时，表示所有的自动清理操作都记录到日志中。当参数设置为-1时，表示所有的自动清理操作都不记录到日志中。

## 6.7 数据库实例系统账号

您在创建GaussDB数据库实例时，系统会自动为实例创建如下系统账户（用户不可使用），用于给数据库实例提供完善的后台运维管理服务。

### 须知

如果试图删掉、重命名、修改这些系统账户的密码和权限，会导致出错，请谨慎操作。

表 6-11 系统账号

账户名称	默认密码	账户登录权限	账户说明	密码有效期
rdsAdmin	系统随机分配	数据库账号，禁止登录	GaussDB管理账户。	永久
rdsBackup	系统随机分配	数据库账号，禁止登录	用于实例数据库做备份。	永久
rdsRepl	系统随机分配	数据库账号，禁止登录	主备同步账户。	永久
root	用户购买实例时手动设置	SSH远程登录	系统管理员账户	永久

账户名称	默认密码	账户登录权限	账户说明	密码有效期
rdsMetric	系统随机分配	数据库账号，禁止登录	获取GaussDB的性能指标，用于上报到Cloud Eye供租户查看GaussDB运行情况的账户。	永久

# 7 与 Oracle 兼容性说明

---

## 7.1 分布式

### 7.1.1 Oracle 数据库兼容性概述

本章节主要介绍GaussDB数据库的Oracle兼容模式与Oracle数据库19C版本的兼容性对比信息。具体的相关信息和规格约束请参考《开发者指南》相应章节。

GaussDB数据库在数据类型、SQL功能和数据库对象等基本功能上与Oracle数据库兼容，PLSQL方面也基本兼容。由于GaussDB数据库与Oracle数据库底层框架实现存在差异，GaussDB数据库与Oracle数据库仍存在部分差异。GaussDB与Oracle的兼容性如下文所示。

### 7.1.2 SQL 的基本元素

GaussDB数据库基本兼容常用的SQL基本元素，但是部分存在差异，详细信息可参考下表。

### 7.1.2.1 数据类型

表 7-1 数值类型

序号	Oracle数据类型	GaussDB数据库是否支持	差异
1	NUMBER [( p [, s ] )]	支持，有差异。	<p>精度和用法存在差异。</p> <ul style="list-style-type: none"> <li>NUMBER带参数时，GaussDB的precision与scale的最大边界值较Oracle更大。</li> <li>NUMBER不带参数时，GaussDB的precision的默认值远大于带参数时的最大边界值；而在Oracle中，precision的默认值等于带参数时的最大边界值。</li> <li>GaussDB不支持scale为负值；在Oracle中，scale为负值时会精确到相应的整数位。</li> </ul>
2	FLOAT [( p )]	支持。	-
3	BINARY_FLOAT	不支持。	-
4	BINARY_DOUBLE	支持。	-

表 7-2 日期时间类型

序号	Oracle数据类型	GaussDB数据库是否支持	差异
1	DATE	支持，有差异。	精度有差异，GaussDB支持的公元时间相较于Oracle范围更大。
2	TIMESTAMP [( fractional_seconds_precision )]	支持。	-

序号	Oracle数据类型	GaussDB数据库是否支持	差异
3	TIMESTAMP [ ( fractional_seconds_precision ) ] WITH TIME ZONE	支持，有差异。	GaussDB的 timestamptz等价于Oracle的 timestampwithlocaltimezone，缺少Oracle对应的 timestamptz类型。  时区更新：部分国家或地区因为政治、经济、战争等因素经常会更新时区信息，数据库系统也因此常常需要同步修改时区文件以确保时间内容的正确性。  GaussDB时区类型目前只涉及 timestamp with timezone，当新的时区文件生效时，不会对已有的数据进行变更，新数据会随时区文件信息进行同步调整。与Oracle的同类型数据能力存在差异。
4	TIMESTAMP [ ( fractional_seconds_precision ) ] WITH LOCAL TIME ZONE	不支持。	-
5	INTERVAL YEAR [ ( year_precision ) ] TO MONTH	支持。	-
6	INTERVAL DAY [ ( day_precision ) ] TO SECOND [ ( fractional_seconds_precision ) ]	支持。	-

表 7-3 字符类型

序号	Oracle数据类型	GaussDB数据库是否支持	差异
1	VARCHAR2 ( size [ BYTE   CHAR ] )	支持，有差异。	在GaussDB中，size单位为字节，即仅支持BYTE，不支持在BYTE和CHAR之间选择，最大10MB；而在Oracle中，size的单位可以在BYTE和CHAR之间选择，最大容量与使用的字符集有关。
2	NVARCHAR2 ( size )	支持，有差异。	在GaussDB中，NVARCHAR2(n)类型是VARCHAR2(n)类型的别名；而在Oracle中NVARCHAR2(n)与VARCHAR2(n)有一定的区别。
3	CHAR [ ( size [ BYTE   CHAR ] ) ]	支持，有差异。	在GaussDB中，size单位为字节，即仅支持BYTE，不支持在BYTE和CHAR之间选择，最大10MB；而在Oracle中，size的单位可以在BYTE和CHAR之间选择，最大容量为2000个字节，实际能容纳的字符数与使用的字符集有关。
4	NCHAR [ ( size ) ]	支持，有差异。	在GaussDB中，size单位为字节，最大10MB；而在Oracle中，size单位为字符，最大容量与使用的字符集有关。
5	CLOB	支持，有差异。	不支持定位器概念。
6	NCLOB	不支持。	-
7	LONG	不支持。	-

表 7-4 二进制类型

序号	Oracle数据类型	GaussDB数据库是否支持	差异
1	RAW ( size )	支持，有差异。	在GaussDB中，size是指字节长度建议值，不会用于校验输入raw类型的字节长度。
2	LONG RAW	不支持。	-



序号	Oracle数据类型	GaussDB数据库是否支持	差异
3	BLOB	支持。	-
4	BFILE	不支持。	-

表 7-5 ROWID 类型

序号	Oracle数据类型	GaussDB数据库是否支持
1	ROWID	不支持
2	UROWID	不支持

表 7-6 用户自定义类型

序号	Oracle数据类型	GaussDB数据库是否支持
1	对象类型	不支持
2	REF数据类型	不支持
3	可变数组	支持
4	嵌套表	支持

表 7-7 伪类型

序号	Oracle数据类型	GaussDB数据库是否支持
1	anytype	不支持
2	anydata	不支持
3	anydataset	不支持

表 7-8 xml 类型

序号	Oracle数据类型	GaussDB数据库是否支持	差异
1	XMLTYPE	支持，有差异。	GaussDB中不支持部分操作，如不能通过使用XMLELEMENT函数将字符串转变为XMLTYPE类型，而是转变成XML类型。具体请参见《开发者指南》中“SQL参考 > 数据类型 > XMLTYPE类型”章节。
2	URIType	不支持。	-

表 7-9 空间类型

序号	Oracle数据类型	GaussDB数据库是否支持
1	SDO_GEOMETRY	不支持
2	SDO_TOPO_GEOMETRY	不支持
3	SDO_GEORASTER	不支持

表 7-10 锁模式

级别	Oracle锁模式	GaussDB锁模式	备注
-	-	NULL	GaussDB的NULL表示未被赋予锁，与Oracle的0级锁对应。
0	none	INVALID	GaussDB的INVALID表示被赋予了非法锁。仅在运行过程中出现了GaussDB无法识别的锁时会被赋予INVALID锁。
1	null	AccessShare	-
2	RS	RowShare	-
3	RX	RowExclusive	-
4	S	ShareUpdateExclusive	-
5	SRX	Share	-
6	\	ShareRowExclusive	-
7	X	Exclusive	-

级别	Oracle锁模式	GaussDB锁模式	备注
8	\	AccessExclusive	-

### 7.1.2.2 数据类型比较规则

数据类型比较（排序）规则是指相同数据类型的值之间发生比较（排序）时遵循的比较（排序）规则。

表 7-11 比较规则

序号	Oracle数据库	Gauss DB数据库	差异	规则描述
1	Numeric值	支持。	-	支持基于数值大小的比较，位于数轴中相对靠近正方向的数字大于位于数轴中相对靠近反方向的数字，如 $5 > 3$ , $1 > -1$ , $-1 > -2$ 。
2	日期时间值	支持。	-	支持基于日期早晚的比较，较晚的日期或时间戳大于较早的日期或时间戳，如 '2000-01-01 12:00:00' > '2000-01-01 11:59:59'。
3	二进制值	支持。	-	支持基于二进制值大小的比较，位于数轴中相对靠近正方向的数字大于位于数轴中相对靠近反方向的数字，如 $1001\ 0101 > 1001\ 0011$ 。

序号	Oracle数据库	Gauss DB数据库	差异	规则描述
4	字符值	支持，有差异。	<ul style="list-style-type: none"> <li>● 有差异的内容                             <ul style="list-style-type: none"> <li>- 在GaussDB和Oracle中，支持的比较规则不完全相同，相同比较规则的名称也可能不同。</li> <li>- GaussDB和Oracle的比较规则在可指定性上有差异，例如GaussDB不支持指定表级别的比较规则，而Oracle支持。</li> <li>- GaussDB和Oracle在指定比较规则的语法上有差异，例如在GaussDB中，使用ENCODING、LC_CTYPE和LC_COLLATE三个参数决定创建数据库时使用的字符集、字符分类和比较规则，具体请参见《开发者指南》中“SQL参考 &gt; SQL语法 &gt; C &gt; CREATE DATABASE”章节。而在Oracle中，各级别的比较规则通常由一系列带有NLS前缀的参数确定。</li> </ul> </li> <li>● 无差异的内容                             <ul style="list-style-type: none"> <li>- 在GaussDB和Oracle中，对字符串的比较都遵循如下的基本规则：从字符串的第一个字符开始逐个比较，如果相等则继续比较下一个字符，直到出现不相等的字符或者有一个字符串已经结束为止。如果两个字符串的所有字符都相等，则认为它们相等；否则，以第一个不相等的字符的大</li> </ul> </li> </ul>	-

序号	Oracle数据库	Gauss DB数据库	差异	规则描述
			<p>小关系作为比较结果进行判断。另外，部分数据类型会在比较前对较短的字符串添加空格，直到两个字符串等长后再排序，如CHAR类型。</p> <ul style="list-style-type: none"> <li>- 对于单字符的比较，常见的比较规则有按字符集中的二进制值排序、单语言排序等。在选用的字符集和比较规则都为默认的情况下GaussDB和Oracle都使用单语言排序。</li> <li>- GaussDB和Oracle中，字符类型的比较规则与所用字符集紧密相关，比如大部分非Unicode字符集不支持单语言排序，因而默认使用二进制值排序。</li> </ul>	
5	对象值	不支持。	-	-
6	Varrays和嵌套表	支持，有差异。	GaussDB和Oracle均支持Varrays的比较，与Oracle不同的是，GaussDB不仅支持比较两个Varrays中的元素个数，还支持同类型的Varrays之间的比较。	-
7	数据类型优先级	支持。	-	-
8	数据转换（显示/隐式类型转换）	支持。	-	-

### 7.1.2.3 字面量

表 7-12 字面量

序号	Oracle数据库	GaussDB数据库
1	文本字面量	支持

序号	Oracle数据库	GaussDB数据库
2	数值字面量	支持
3	日期时间字面量	支持
4	区间字面量	支持

### 7.1.2.4 格式模型

表 7-13 格式

序号	Oracle数据库	GaussDB数据库	差异
1	数字格式	支持，有差异。	GaussDB仅在参数 a_format_version值为10c和 a_format_dev_version值为s1的情况下，支持\$、C、TM、TM9、TME、U格式。同时在该参数下，不支持TH、PL、SG格式的格式。 GaussDB具体支持情况请参见《开发者指南》中“SQL参考>函数和操作符>类型转换函数”章节的“number类型fmt参数表”。
2	日期时间格式	支持，有差异。	GaussDB中用于时间截断和时间四舍五入的参数仅在参数 a_format_version值为10c和 a_format_dev_version值为s1的情况下有效。 GaussDB具体支持情况请参见《开发者指南》中“SQL参考>函数和操作符>时间和日期处理函数和操作符”章节的“用于日期/时间格式化的模式”。
3	格式模型修饰符	支持。	-
4	字符串到日期转换规则	支持，有差异。	GaussDB中to_timestamp_tz函数在参数a_format_version值为10c和 a_format_dev_version值为s1的情况下有效。 GaussDB具体支持情况请参见《开发者指南》中“SQL参考 > 函数和操作符 > 类型转换函数”章节的“to_date/to_timestamp/to_timestamp_tz”。
5	xml格式模式	不支持。	-

### 7.1.2.5 空值

表 7-14 空值

序号	Oracle数据库	GaussDB数据库
1	IS NULL和IS NOT NULL	支持
2	NULLS in conditions	支持

### 7.1.2.6 注释

表 7-15 comment

序号	Oracle数据库	GaussDB数据库	差异
1	斜杠和星号 (/*)	支持	-
2	两个连字符 (--)	支持	-
3	COMMENT命令	支持	-
4	HINT	支持，有差异	GaussDB不支持'--+hint形式。具体信息请参见《开发者指南》中“SQL调优指南 > 使用Plan Hint进行调优”章节。

### 7.1.2.7 数据库对象

表 7-16 schema 对象

序号	Oracle数据库	GaussDB数据库	差异
1	分析视图	不支持	-
2	属性维度	不支持	-
3	集群	支持	-
4	约束	支持	-
5	数据库链接	支持	-
6	数据库触发器	支持	-
7	尺寸	支持	-

序号	Oracle数据库	GaussDB数据库	差异
8	外部过程库	不支持	-
9	分层结构	不支持	-
10	索引组织表	不支持	-
11	索引	支持	-
12	索引类型	不支持	-
13	java类	不支持	-
14	java资源	不支持	-
15	java源码	不支持	-
16	join groups	不支持	-
17	物化视图	支持	-
18	物化视图日志	不支持	-
19	挖掘模型	不支持	-
20	对象表	不支持	-
21	对象类型	不支持	-
22	对象视图	不支持	-
23	operators	支持	-
24	包	支持	-
25	序列	支持	-
26	存储函数	支持	-
27	存储过程	支持	-
28	同义词	支持，有差异	Oracle数据库的数据库对象在同一namespace内不能重名。GaussDB同一namespace内同义词可与表、视图、函数、package重名，如果重名，会优先访问本名对象，未寻找到本名对象时才会寻找同义词指向的对象；Schema名是用户名时才会搜索PUBLIC同义词。
29	表	支持	-
30	视图	支持	-
31	zone map	不支持	-



表 7-17 nonschema 对象

序号	Oracle数据库	GaussDB数据库
1	上下文	不支持
2	目录	支持
3	版本	不支持
4	闪回存档	不支持
5	锁定配置文件	不支持
6	配置文件	不支持
7	还原点	支持
8	角色	支持
9	回滚段	<ul style="list-style-type: none"> <li>• ustore支持回滚段</li> <li>• astore不支持回滚段</li> </ul>
10	表空间	支持
11	表空间集	不支持
12	统一审计策略	支持
13	用户	支持

### 7.1.2.8 数据库对象名称和限定符

表 7-18 命名规则

序号	Oracle数据库	GaussDB数据库	差异
1	数据库对象命名规则	支持，有差异。	GaussDB默认小写。
2	模式对象命名规则	支持。	-

### 7.1.2.9 SQL 语句中的引用架构对象和部件的语法

表 7-19 对象引用

序号	Oracle数据库	GaussDB数据库
1	引用对象的一般语法	支持
2	解析对象的引用	支持

序号	Oracle数据库	GaussDB数据库
3	引用外部模式对象	支持
4	引用外部数据库对象	支持
5	引用表和索引分区和子分区	支持

### 7.1.3 伪列

GaussDB数据库兼容序列伪列、rownum伪列，其余暂不支持。

#### 序列伪列

表 7-20 序列

序号	Oracle数据库	GaussDB数据库	差异
1	currval	支持，有差异。	GaussDB以函数形式实现。兼容Oracle调用方式。
2	nextval	支持，有差异。	GaussDB以函数形式实现。兼容Oracle调用方式。

#### rownum 伪列

表 7-21 rownum

序号	Oracle数据库	GaussDB数据库	差异
1	rownum	支持，有差异。	Oracle在left、right、full join的条件中使用rownum进行过滤时，不同的条件下表现不尽相同，可能存在忽略或部分忽略rownum条件的现象，而GaussDB在此情况下则表现为对left、right、full join后的结果进行过滤。

## xmldata 伪列

表 7-22 xmldata

序号	Oracle数据库	GaussDB数据库
1	xmldata	不支持

## 7.1.4 操作符

GaussDB数据库兼容除分层查询以外的运算符。

### SQL 运算符

表 7-23 SQL 运算符

序号	Oracle数据库	GaussDB数据库
1	一元运算符和二元运算符	支持
2	运算符优先级	支持

### 算术运算符

表 7-24 算术运算符

序号	Oracle数据库	GaussDB数据库
1	正负 (+-) 一元运算符	支持
2	加减 (+-) 二元运算符	支持
3	乘除 (*/) 二元运算符	支持

### COLLATE 运算符

表 7-25 COLLATE 运算符

序号	Oracle数据库	GaussDB数据库
1	COLLATE collation_name	支持

## 连接运算符

表 7-26 连接运算符

序号	Oracle数据库	GaussDB数据库
1		支持

## 集合运算符

表 7-27 集合运算符

序号	Oracle数据库	GaussDB数据库
1	union	支持
2	union all	支持
3	intersect	支持
4	minus	支持

## 多集合运算符

表 7-28 多集合运算符

序号	Oracle数据库	GaussDB数据库
1	multiset except	支持
2	multiset intersect	支持
3	multiset union	支持

## 用户自定义运算符

表 7-29 用户自定义运算符

序号	Oracle数据库	GaussDB数据库	差异
1	CREATE OPERATOR	支持	<ul style="list-style-type: none"> <li>Oracle中提供的CONTEXT_CLAUSE支持自定义功能评估函数，和GaussDB约束选择性评估函数不同。GaussDB不支持自定义功能评估函数。</li> <li>GaussDB和Oracle可选参数差异较大。GaussDB具体请参考《开发者指南》中“SQL参考 &gt; SQL语法 &gt; C &gt; CREATE OPERATOR”章节的参数说明部分。</li> </ul>

## 比较操作符

序号	Oracle数据库	GaussDB数据库
1	< =	支持
2	< >	支持
3	> =	支持
4	^ =	支持
5	! =	不支持，!=中间存在空格时，!会被识别为阶乘。

当比较操作符（<=、<>、>=、^=）中间存在空格时，也可以识别成没有空格进行正常操作。!=中间存在空格时，!会被识别为阶乘，可能会导致结果与预期不一致。

### 7.1.5 表达式

GaussDB数据库兼容大部分数据库表达式。

表 7-30 表达式

序号	Oracle数据库	GaussDB数据库	差异
1	简单表达式	支持。	-

序号	Oracle数据库	GaussDB数据库	差异
2	分析视图表达式	不支持。	-
3	复合表达式	支持。	-
4	case表达式	支持。	-
5	列表式	支持。	-
6	cursor表达式	不支持。	-
7	日期时间表达式	支持，有差异。	GaussDB的输出结果中不会带时区信息，Oracle会带有类似PM AMERICA/LOS_ANGELES的时区信息。
8	函数表达式	支持。	-
9	区间表达式	部分支持。	GaussDB支持形如SELECT INTERVAL '999999999 23:59:59.999' day(9) to second FROM DUAL;的语句，而不支持形如SELECT(SYSDATE- SYSDATE) DAY TO SECOND FROM DUAL;的语句。Oracle均支持。
10	json对象访问表达式	部分支持，有差异。	<ul style="list-style-type: none"> <li>GaussDB支持通过“-&gt;'key' “的方式从JSON对象中提取value，而Oracle支持通过“.key “方式来提取value。</li> <li>对于JSONARRAY对象而言，Oracle支持通过“.key “方式一次性提取所有key对应的value，但GaussDB目前不支持。</li> </ul>
11	模型表达式	不支持。	-
12	对象表达式	不支持。	-
13	占位符表达式	部分支持。	对于形如“.var “的一般占位符表达式，GaussDB支持，但不支持通过INDICATOR关键字将两个一般占位符表达式结合起来。
14	标量子查询表达式	支持。	-
15	类型构造器表达式	部分支持。	GaussDB不支持在类型构造器前指定New关键字，而Oracle支持。
16	表达式list	支持。	-

## 7.1.6 条件

GaussDB数据库兼容Oracle大部分条件。

表 7-31 条件

序号	Oracle数据库	GaussDB数据库	差异
1	比较条件	支持，有差异。	语句中存在ANY、SOME、ALL操作符时存在差异，Oracle支持对list对象进行操作，而GaussDB中需要将list对象转换成数组表达式的形式后再进行操作。
2	浮点条件	不支持。	-
3	逻辑条件	支持。	-
4	模型条件	不支持。	-
5	多集合条件	不支持。	-
6	模式匹配条件	支持。	-
7	NULL值条件	支持。	-
8	XML条件	不支持。	-
9	SQL/JSON条件	部分支持，有差异。	<ul style="list-style-type: none"><li>• GaussDB不支持IS JSON条件和JSON_TEXTCONTAINS条件。</li><li>• GaussDB中JSONB_EQ条件同等于Oracle中JSON_EQUAL条件，但GaussDB不支持ERROR子句。</li><li>• GaussDB中JSONB_EXISTS条件同等于Oracle中JSON_EXISTS条件，但GaussDB不支持ERROR子句、EMPTY子句和PASSING子句。</li></ul>
10	复合条件	支持。	-
11	BETWEEN条件	支持。	-
12	EXISTS条件	支持。	-
13	IN条件	支持。	-
14	IS OF TYPE条件	不支持。	-

## 7.1.7 常见的 SQL DDL 子句

GaussDB数据库兼容部分DDL子句。

表 7-32 常用 SQL DDL 子句

序号	Oracle数据库	GaussDB数据库	差异
1	分配扩展子句	不支持。	-
2	约束	支持。	-
3	取消分配未使用子句	不支持。	-
4	文件规范	不支持。	-
5	日志记录子句	部分支持，有差异。	<ul style="list-style-type: none"> <li>● GaussDB不支持LOGGING约束子句和FILESYSTEM_LIKE_LOGGING约束子句。</li> <li>● GaussDB仅支持表级的UNLOGGED约束，不支持列级的UNLOGGED约束。</li> <li>● GaussDB仅支持在CREATE TABLE、CREATE TABLE AS、SELECT INTO语句中使用日志记录子句。</li> </ul>
6	并行子句	不支持。	-
7	物理属性子句	部分支持，有差异。	<ul style="list-style-type: none"> <li>● GaussDB不支持PCTUSED。</li> <li>● GaussDB 仅支持在CREATE TABLE、CREATE INDEX语句中使用物理属性子句。</li> </ul>
8	大小子句	不支持。	-
9	存储子句	部分支持，有差异。	<ul style="list-style-type: none"> <li>● Oracle中由STORAGE子句指定存储参数，而GaussDB中由WITH子句指定存储参数。</li> <li>● GaussDB中可选的存储参数和Oracle存在很大差异。GaussDB具体可参见《开发者指南》中“SQL参考 &gt; SQL语法 &gt; C &gt; CREATE TABLE”的参数说明部分，“WITH ({storage_parameter = value} [, ...])”中描述了CREATE TABLE语句支持的存储参数。</li> </ul>

## 7.1.8 SQL 查询和子查询

GaussDB数据库兼容除分层查询以外的SQL查询和子查询。



表 7-33 SQL 查询和子查询

序号	Oracle数据库	GaussDB数据库	差异
1	创建简单查询	支持。	-
2	分层查询	不支持。	-
3	UNION [ALL], INTERSECT, 减运算符	支持。	-
4	查询结果排序	支持, 有差异。	GaussDB查询不包含分组, 且目标列同时包含聚集函数和集合返回函数时, 不忽略对集合返回函数列的排序。
5	Joins	支持, 有差异。	GaussDB只支持和Oracle相同的Join Types, 如left、right、self、natural、full outer join等。不支持In-Memory Join Groups等Join Optimizations方法。
6	使用子查询	支持。	-
7	嵌套子查询的解嵌套	支持, 有差异。	GaussDB不支持显式指定HASH_AJ或MERGE_AJ。
8	分布式查询	支持, 有差异。	GaussDB需要显式DBLINK查询。
9	聚集函数嵌套	支持。	-

## 7.1.9 PL/SQL 语言

GaussDB数据库基本兼容的PL/SQL操作符、表达式, 控制语句、集合和record等等, 不支持预定义的PL/SQL常量和类型、子类型等。

### 7.1.9.1 PL/SQL 基本语法

表 7-34 PL/SQL 操作符

序号	Oracle数据库	GaussDB数据库
1	+	支持
2	:=	支持
3	=>	支持
4	%	支持
5	'	支持
6	.	支持

序号	Oracle数据库	GaussDB数据库
7		支持
8	/	支持
9	**	不支持
10	(	支持
11	)	支持
12	:	支持
13	,	支持
14	<<	支持
15	>>	支持
16	/*	支持
17	*/	支持
18	*	支持
19	"	支持
20	..	支持
21	=	支持
22	<>	支持
23	!=	支持
24	~=	支持
25	^=	支持
26	<	支持
27	>	支持
28	<=	支持
29	>=	支持
30	@	支持
31	--	支持
32	;	支持
33	-	支持

表 7-35 逻辑运算符

序号	Oracle数据库	GaussDB数据库
1	NOT	支持
2	AND	支持
3	OR	支持

表 7-36 比较表达式

序号	Oracle数据库	GaussDB数据库
1	IS [NOT] NULL	支持
2	LIKE	支持
3	BETWEEN	支持
4	IN	支持

表 7-37 条件表达式

序号	Oracle数据库	GaussDB数据库
1	simple CASE	支持
2	searched CASE	支持

表 7-38 变量声明相关参数

序号	Oracle数据库	GaussDB数据库	差异
1	%TYPE	支持，有差异。	<ul style="list-style-type: none"> <li>GaussDB不支持record变量%type。</li> <li>GaussDB不支持pkg.record变量%type、schema.pkg.record变量%type作为出入参类型。</li> <li>GaussDB不支持表/视图.column.column %type、schema.表/视图.column.column%type嵌套1层及以上，作为变量类型或者出入参类型。</li> <li>GaussDB不支持record变量.column.column%type、pkg.record变量.column.column%type嵌套1层及以上的record的某列类型，作为变量类型或者出入参类型。</li> </ul>

序号	Oracle数据库	GaussDB数据库	差异
2	%ROWTYPE	支持，有差异。	<ul style="list-style-type: none"> <li>GaussDB在多个CN的环境下，存储过程中无法声明临时表的%ROWTYPE及%TYPE属性。因为临时表仅在当前session有效，在编译阶段其他CN无法看到当前CN的临时表。故多个CN的环境下，会提示该临时表不存在。</li> <li>GaussDB不支持view%rowtype、schema.view%rowtype作为出入参类型。</li> <li>GaussDB不支持package.cursor变量%rowtype作为出入参类型。</li> </ul>

### 7.1.9.2 数据类型兼容性

表 7-39 其他 PL/SQL 数据类型

序号	Oracle数据库	GaussDB数据库	差异
1	CHARACTER	支持，有差异。	<ul style="list-style-type: none"> <li>GaussDB中字节长度限制为：1~10485760。</li> <li>Oracle中字节长度限制为：1~32767。</li> </ul>
2	VARCHAR	支持，有差异。	<ul style="list-style-type: none"> <li>GaussDB中字节长度限制为：1~10485760。</li> <li>Oracle中字节长度限制为：1~32767。</li> </ul>
3	STRING	不支持。	-
4	PLS_INTEGER	不支持。	GaussDB中可使用INT类型替代。
5	BINARY_INTEGER	支持。	-

### 7.1.9.3 控制语句

表 7-40 条件语句

序号	Oracle数据库	GaussDB数据库
1	IF THEN	支持
2	IF THEN ELSE	支持

序号	Oracle数据库	GaussDB数据库
3	IF THEN ELSIF	支持
4	<b>simple CASE:</b> CASE selector WHEN selector_value_1 THEN statements_1 WHEN selector_value_2 THEN statements_2 ... WHEN selector_value_n THEN statements_n [ ELSE else_statements END CASE;]	支持
5	<b>searched CASE:</b> CASE WHEN condition_1 THEN statements_1 WHEN condition_2 THEN statements_2 ... WHEN condition_n THEN statements_n [ ELSE else_statements END CASE;]	支持

表 7-41 LOOP 循环语句

序号	Oracle数据库	GaussDB数据库
1	[ label ] LOOP statements END LOOP [ label ];	支持
2	EXIT;	支持
3	EXIT WHEN;	支持
4	CONTINUE;	支持
5	CONTINUE WHEN;	支持

表 7-42 FOR 循环语句

序号	Oracle数据库	GaussDB数据库	差异
1	[ label ] FOR index IN [ REVERSE ] lower_bound..upper_bound LOOP statements END LOOP [ label ];	支持，有差异。	GaussDB使用REVERSE关键字时，lower_bound必须大于等于upper_bound，否则循环体不会被执行。
2	EXIT WHEN;	支持。	-
3	CONTINUE WHEN;	支持。	-

表 7-43 WHILE LOOP 循环语句

序号	Oracle数据库	GaussDB数据库
1	[ label ] WHILE condition LOOP statements END LOOP [ label ];	支持

表 7-44 GOTO 语句

序号	Oracle数据库	GaussDB数据库
1	GOTO	支持

表 7-45 NULL 语句

序号	Oracle数据库	GaussDB数据库
1	NULL	支持

### 7.1.9.4 集合和 Record

表 7-46 类型

序号	Oracle数据库	GaussDB数据库
1	Associative array (or index-by table)	支持
2	VARRAY (variable-size array)	支持

序号	Oracle数据库	GaussDB数据库
3	Nested table	支持
4	record	支持

表 7-47 语法

序号	类型	Oracle数据库	GaussDB数据库	差异
1	Associative array (or index-by table)	TABLE OF datatype [ NOT NULL ] INDEX BY { PLS_INTEGER   BINARY_INTEGER   VARCHAR2 ( v_size )   data_type }	支持，有差异。	<ul style="list-style-type: none"><li>• GaussDB不支持 PLS_INTEGER类型，GaussDB内 data_type可以为基础数据类型、或存储过程内定义的 record类型、集合类型、数组类型，不支持ref cursor类型。</li><li>• GaussDB内NOT NULL只支持语法不支持功能，即不会校验元素是否为NULL。</li><li>• 详情可参考《开发者指南》中“存储过程 &gt; 数组、集合和record &gt; 集合”章节。</li></ul>

序号	类型	Oracle数据库	GaussDB数据库	差异
2	VARRAY (variable-size array)	{ VARRAY   [ VARYING ] ARRAY } ( size_limit ) OF datatype [ NOT NULL ]	支持，有 差异。	<ul style="list-style-type: none"> <li>• GaussDB内不支持 NOT NULL语法。</li> <li>• GaussDB内不支持 datatype为varray 类型（ varray不能嵌套 ）。</li> <li>• size_limit功能生效需要在 behavior_compat_optionsGUC参数中开启 varray_compat参数。</li> <li>• 详情可参考《开发者指南》中“存储过程 &gt; 数组、集合和record &gt; 数组”章节。</li> </ul>
3	Nested table	TABLE OF datatype [ NOT NULL ]	支持，有 差异。	<ul style="list-style-type: none"> <li>• GaussDB内NOT NULL只支持语法不支持功能。</li> <li>• 详情可参考《开发者指南》中“存储过程 &gt; 数组、集合和record &gt; 集合”章节。</li> </ul>
4	record	TYPE record_type IS RECORD ( field_definition [, field_definition]... ) ;	支持。	<ul style="list-style-type: none"> <li>• record的列可以定义为NOT NULL属性也可以指定默认值。其他类型嵌套 record类型， record类型的默认值和NOT NULL不生效；通过 package.record_type访问类型的形式来创建record变量，该record变量的默认值和NOT NULL不生效。</li> <li>• 详情可参考《开发者指南》中“存储过程 &gt; 数组、集合和record &gt; record ”章节。</li> </ul>



表 7-48 构造器

序号	Oracle数据库	GaussDB数据库
1	collection_type ( [ value [, value ]... ] )	支持

表 7-49 变量赋值

序号	Oracle数据库	GaussDB数据库	差异
1	Associative array (or index-by table)	支持。	-
2	VARRAY (variable-size array)	支持，有差异。	<ul style="list-style-type: none"> <li>• GaussDB不同VARRAY类型的数据可以相互赋值，取决于其元素之间是否能相互隐式转换。</li> <li>• 详情可参考《开发者指南》中“存储过程 &gt; 数组、集合和 record &gt; 数组”章节。</li> </ul>
3	Nested table	支持。	-
4	record	支持，有差异。	<ul style="list-style-type: none"> <li>• GaussDB不同record类型的数据可以相互赋值，取决于列与列之间是否能相互隐式转换。</li> <li>• 详情可参考《开发者指南》中“存储过程 &gt; 数组、集合和 record &gt; record ”章节。</li> </ul>

表 7-50 集合操作符

序号	Oracle数据库	GaussDB数据库	差异
1	=	支持，有差异。	<ul style="list-style-type: none"> <li>Oracle：比较时忽略集合成员先后顺序。</li> <li>GaussDB：比较时严格按照集合成员先后顺序。</li> </ul>
2	<>	支持，有差异。	<ul style="list-style-type: none"> <li>Oracle：比较时忽略集合成员先后顺序。</li> <li>GaussDB：比较时严格按照集合成员先后顺序。</li> </ul>
3	IS[NOT] NULL	支持。	-
4	^=	支持，有差异。	<ul style="list-style-type: none"> <li>Oracle：比较时忽略集合成员先后顺序。</li> <li>GaussDB：比较时严格按照集合成员先后顺序。</li> </ul>
5	~=	不支持。	-
6	IS[NOT] A SET	不支持。	-
7	IS [NOT] EMPTY	不支持。	-
8	expr [ NOT ] MEMBER [ OF ] nested_table	不支持。	-
9	nested_table1 [ NOT ] SUBMULTISET [ OF ] nested_table2	不支持。	-
10	[NOT] IN	支持。	<ul style="list-style-type: none"> <li>Oracle：比较时忽略集合成员先后顺序。</li> <li>GaussDB：比较时严格按照集合成员先后顺序。</li> </ul>

表 7-51 集合 MULTISSET 函数

序号	Oracle数据库	GaussDB数据库
1	MULTISSET UNION [ALL   DISTINCT]	支持
2	MULTISSET EXCEPT [ALL   DISTINCT]	支持
3	MULTISSET INTERSECT [ALL   DISTINCT]	支持

表 7-52 集合类型函数

序号	Oracle数据库	GaussDB数据库	差异
1	exists(idx)	支持。	-
2	extend[(count[, idx])]	支持，有差异。	GaussDB仅支持nesttable类型。
3	delete[(idx1[, idx2])]	支持。	-
4	trim[(n)]	支持，有差异。	GaussDB仅支持nesttable类型。
5	count	支持。	-
6	first	支持。	-
7	last	支持。	-
8	prior(idx)	支持。	-
9	next(idx)	支持。	-
10	limit	支持，有差异。	GaussDB用于nesttable类型，返回集合中最大可以储存的元素个数，只适用于array类型，nesttable返回空。

表 7-53 record 变量操作

序号	Oracle数据库	GaussDB数据库
1	构造器	支持
2	%ROWTYPE声明变量	支持
3	定义常量constant	不支持

表 7-54 集合相关函数

序号	Oracle数据库	GaussDB数据库
1	unnest_table(anynesttable)	支持
2	unnest_table(anyindexbyteable)	支持

### 7.1.9.5 静态 SQL

表 7-55 静态查询 SQL 语句

序号	Oracle数据库	GaussDB数据库	差异
1	SELECT	支持，有差异。	GaussDB和Oracle在某些场景下有不同。 GaussDB中FOR SHARE对检索出来的行加共享锁，不同事务的共享锁不会互相阻塞。若数据在一个事务中被FOR SHARE锁定，在另一个事务中使用SELECT FOR SHARE SKIP LOCKED时，SKIP LOCKED不会跳过锁。

表 7-56 静态 DML SQL 语句

序号	Oracle数据库	GaussDB数据库
1	INSERT	支持
2	UPDATE	支持
3	DELETE	支持
4	MERGE	支持
5	LOCK TABLE	支持

表 7-57 静态 TCL SQL 语句

序号	Oracle数据库	GaussDB数据库	差异
1	COMMIT	支持。	-
2	ROLLBACK	支持。	-
3	SAVEPOINT	支持。	-

序号	Oracle数据库	GaussDB数据库	差异
4	SET TRANSACTION	支持，有差异。	GaussDB不支持NAME string语法、USE ROLLBACK SEGMENT rollback_segment语法。

表 7-58 伪列

序号	Oracle数据库	GaussDB数据库	差异
1	CURRVAL and NEXTVAL	支持。	-
2	LEVEL	不支持。	-
3	OBJECT_VALUE	不支持。	-
4	ROWID	不支持。	-
5	ROWNUM	支持，有差异。	不推荐ROWNUM条件用于JOIN ON子句。 GaussDB中ROWNUM条件用于JOIN ON子句时在LEFT JOIN、RIGHT JOIN、FULL JOIN场景下和MERGE INTO场景下与其他数据库行为不一致，直接进行业务迁移存在风险。

表 7-59 隐式游标属性

序号	Oracle数据库	GaussDB数据库	差异
1	SQL%FOUND	支持，有差异。	GaussDB在commit \rollback之后不刷新隐式游标结果，Oracle会在commit\rollback之后刷新隐式游标结果。
2	SQL%NOTFOUND	支持，有差异。	
3	SQL%ROWCOUNT	支持，有差异。	
4	SQL%ISOPEN	支持，有差异。	
5	SQL %BULK_ROWCOU NT	不支持。	
6	SQL %BULK_EXCEPTIO NS	不支持。	

表 7-60 显式游标语法及关键字

序号	Oracle数据库	GaussDB数据库
1	CURSOR cursor_name [ parameter_list ] RETURN return_type;	支持
2	CURSOR cursor_name [ parameter_list ] [ RETURN return_type ] IS select_statement;	支持
3	OPEN	支持
4	CLOSE	支持
5	FETCH	支持
6	CURRENT OF CURSOR	支持

表 7-61 显式游标属性

序号	Oracle数据库	GaussDB数据库
1	SQL%FOUND	支持
2	SQL%NOTFOUND	支持
3	SQL%ROWCOUNT	支持
4	SQL%ISOPEN	支持

表 7-62 游标循环

序号	Oracle数据库	GaussDB数据库
1	FOR LOOP	支持

表 7-63 自治事务支持场景

序号	Oracle数据库	GaussDB数据库
1	存储过程	支持
2	匿名块	支持
3	函数	支持
4	Package	支持

### 7.1.9.6 动态 SQL

表 7-64 动态 SQL 语句执行方式

序号	Oracle数据库	GaussDB数据库	差异
1	EXECUTE IMMEDIATE	支持，有差异。	<ul style="list-style-type: none"> <li>GaussDB通过dynamic_sql_compat参数控制同名变量是否读取同一参数，并且检查调用存储过程时绑定参数出入参类型是否与语句参数类型一致。</li> <li>GaussDB不支持调用匿名块中部分绑定参数场景，例如匿名块中嵌套动态语句，使用表达式绑定参数，具体请参见《开发者指南》中“存储过程 &gt; 动态语句 &gt; 动态调用匿名块”章节。</li> <li>GaussDB不支持RETURNING、RETURN INTO。</li> </ul>
2	OPEN FOR、FETCH、CLOSE	支持。	-

### 7.1.9.7 Trigger

表 7-65 trigger 类型

序号	Oracle数据库	GaussDB数据库	差异
1	DML TRIGGER	支持，有差异。	GaussDB不支持Compound DML Triggers。
2	SYSTEM TRIGGER	不支持。	-

表 7-66 create trigger

语法类型	Oracle数据库	GaussDB数据库	差异
create语法	CREATE [ OR REPLACE ] [ EDITIONABLE   NONEDITIONABLE ] TRIGGER plsql_trigger_source	支持，有差异。	GaussDB不支持OR REPLACE、EDITIONABLE   NONEDITIONABLE，支持plsql_trigger_source部分行为。

语法类型	Oracle数据库	GaussDB数据库	差异
plsql_trigger_source ::=	[schema.] trigger_name [ sharing_clause ] [ default_collation_clause ] { simple_dml_trigger   instead_of_dml_trigger   compound_dml_trigger   system_trigger }	支持，有差异。	GaussDB不支持 schema、 sharing_clause、 default_collation_clause。
simple_dml_trigger ::=	{ BEFORE   AFTER } dml_event_clause [ referencing_clause ] [ FOR EACH ROW ] [ trigger_edition_clause ] [ trigger_ordering_clause ] [ ENABLE   DISABLE ] [ WHEN ( condition ) ] trigger_body	支持，有差异。	GaussDB不支持 referencing_clause 、 referencing_clause ( 用from referencing_table 代替 )、 trigger_edition_clause、 trigger_ordering_clause、 ENABLE   DISABLE，支持 trigger_body部分 行为。  GaussDB在没有 INSTEAD OF TRIGGER的视图上 创建语句级 BEFORE/AFTER TRIGGER时不会报 错，执行DML时报 错。
dml_event_clause ::=	{ DELETE   INSERT   UPDATE [ OF column [, column ]... ] } [ OR { DELETE   INSERT   UPDATE [ OF column [, column]... ] }... ON [ schema.] { table   view }	不支持。	-



语法类型	Oracle数据库	GaussDB数据库	差异
trigger_body ::=	{ plsql_block   CALL routine_clause }	支持，有差异。	GaussDB不支持 plsql_block，仅支持EXECUTE PROCEDURE function_name ( arguments );方式执行function，并且function需要用户定义，必须声明为不带参数并返回类型为触发器，在触发器触发时执行。
instead_of_dml_trigger ::=	INSTEAD OF { DELETE   INSERT   UPDATE } [ OR { DELETE   INSERT   UPDATE } ]... ON [ NESTED TABLE nested_table_column OF ] [ schema. ] nonconditioning_view [ referencing_clause ] [ FOR EACH ROW ] [ trigger_edition_clause ] [ trigger_ordering_clause ] [ ENABLE   DISABLE ] trigger_body	支持，有差异。	GaussDB不支持 NESTED TABLE nested_table_column OF、referencing_clause、trigger_edition_clause、trigger_ordering_clause、ENABLE   DISABLE。
compound_dml_trigger ::=	CREATE trigger FOR dml_event_clause ON view COMPOUND TRIGGER INSTEAD OF EACH ROW IS BEGIN statement; END INSTEAD OF EACH ROW;	不支持。	-

语法类型	Oracle数据库	GaussDB数据库	差异
system_trigger ::=	{ BEFORE   AFTER   INSTEAD OF } { ddl_event [OR ddl_event]...   database_event [OR database_event ]... } ON { [schema.] SCHEMA   [ PLUGGABLE ] DATABASE } [ trigger_ordering_clause ] [ ENABLE   DISABLE ] trigger_body	不支持。	-

表 7-67 alter trigger

序号	Oracle数据库	GaussDB数据库	差异
1	ALTER TRIGGER [ schema. ] trigger_name { trigger_compile_clause   { ENABLE   DISABLE }   RENAME TO new_name   { EDITIONABLE   NONEDITIONABLE } };	支持，有差异。	GaussDB不支持 schema、trigger_compile_clause、{ ENABLE   DISABLE }、{ EDITIONABLE   NONEDITIONABLE }。

表 7-68 drop trigger

序号	Oracle数据库	GaussDB数据库	差异
1	DROP TRIGGER [ schema. ] trigger ;	支持，有差异。	GaussDB不支持schema，需要在 trigger_name后面加上ON table_name。

Oracle数据库名为\*\_TRIGGERS的视图统计了trigger的相关信息，GaussDB相关视图与Oracle存在差异，GaussDB视图具体请参见《开发者指南》中“系统表和系统视图 > 系统视图 > 其他系统视图”中的DB\_TRIGGERS、ADM\_TRIGGERS、MY\_TRIGGERS章节。

表 7-69 Nested, Package, and Standalone Subprograms 兼容性

序号	Oracle数据库	GaussDB数据库	差异
1	nested subprogram (子块)	支持，有差异。	GaussDB不支持重载。 GaussDB不支持定义为自治事务。 GaussDB不支持 SETOF的使用，仅限一个限定符引用嵌套子程序或嵌套子程序的变量。
2	package subprogram	支持。	-
3	standalone subprogram (包含 Function & Procedure)	支持。	-
4	匿名块	支持。	-

表 7-70 RETURN 语句支持情况

序号	Oracle数据库	GaussDB数据库
1	Function	支持
2	Procedure	支持
3	匿名块	支持

表 7-71 Function 相关参数

序号	Oracle数据库	GaussDB数据库	差异
1	DETERMINISTIC	支持，有差异。	GaussDB中为 IMMUTABLE。
2	PARALLEL_ENABLE	不支持。	-
3	PIPELINED	不支持。	-
4	RESULT_CACHE	不支持。	-

表 7-72 参数形式支持

序号	Oracle数据库	GaussDB数据库
1	IN	支持
2	OUT	支持
3	IN OUT	支持

表 7-73 CREATE 语句

序号	Oracle数据库	GaussDB数据库	差异
1	CREATE FUNCTION	支持，有差异。	GaussDB不支持IF NOT EXISTS语法、不支持sharing_clause、仅支持部分指定function属性的子句（属性的子句仅支持invoker_rights_clause子句）、不支持关键字[ EDITIONABLE   NONEDITIONABLE ]。 GaussDB的具体语法请参见《开发者指南》中” SQL参考 > SQL语法 > C >CREATE FUNCTION ” 章节。
2	CREATE LIBRARY	不支持。	-
3	CREATE PACKAGE	支持，有差异。	GaussDB不支持IF NOT EXISTS语法、不支持sharing_clause、仅支持部分指定package属性的子句（属性的子句仅支持invoker_rights_clause子句）、不支持关键字[ EDITIONABLE   NONEDITIONABLE ]。 GaussDB的具体语法请参见《开发者指南》中” SQL参考 > SQL语法 > C >CREATE PACKAGE ” 章节。
4	CREATE PACKAGE BODY	支持，有差异。	GaussDB不支持IF NOT EXISTS语法、不支持sharing_clause、不支持关键字[ EDITIONABLE   NONEDITIONABLE ]。 GaussDB的具体语法请参见《开发者指南》中” SQL参考 > SQL语法 > C >CREATE PACKAGE ” 章节。

序号	Oracle数据库	GaussDB数据库	差异
5	CREATE PROCEDURE	支持，有差异。	GaussDB不支持IF NOT EXISTS语法、不支持sharing_clause子句、不支持关键字 [ EDITIONABLE   NONEDITIONABLE ]。 GaussDB的具体语法请参见《开发者指南》中” SQL参考 > SQL语法 > C >CREATE PROCEDURE ” 章节。
6	CREATE TRIGGER	支持，有差异。	GaussDB的具体语法请参见《开发者指南》中” SQL参考 > SQL语法 > C >CREATE TRIGGER ” 章节。
7	CREATE TYPE	支持，有差异。	GaussDB不支持varray、object类型、UNDER语法。 GaussDB的具体语法请参见《开发者指南》中” SQL参考 > SQL语法 > C >CREATE TYPE ” 章节。
8	CREATE TYPE BODY	不支持。	-

表 7-74 ALTER 语句

序号	Oracle数据库	GaussDB数据库	差异
1	ALTER FUNCTION	支持，有差异。	GaussDB不支持关键字[ EDITIONABLE   NONEDITIONABLE ]、REUSE 、 SETTINGS、DEBUG。 GaussDB的具体语法请参见《开发者指南》中” SQL参考 > SQL语法 > A > ALTER FUNCTION ” 章节。
2	ALTER LIBRARY	不支持。	-
3	ALTER PACKAGE	支持，有差异。	GaussDB不支持关键字[ EDITIONABLE   NONEDITIONABLE ]、REUSE 、 SETTINGS、DEBUG。 GaussDB的具体语法请参见《开发者指南》中” SQL参考 > SQL语法 > A > ALTER PACKAGE ” 章节。
4	ALTER PROCEDURE	支持，有差异。	GaussDB不支持关键字[ EDITIONABLE   NONEDITIONABLE ]、REUSE 、 SETTINGS、DEBUG。

序号	Oracle数据库	GaussDB数据库	差异
5	ALTER TRIGGER	支持，有差异。	GaussDB仅支持修改trigger名字。 GaussDB的具体语法请参见《开发者指南》中” SQL参考 > SQL语法 > A > ALTER TRIGGER ” 章节。
6	ALTER TYPE	支持，有差异。	GaussDB仅支持部分语句。 GaussDB的具体语法请参见《开发者指南》中” SQL参考 > SQL语法 > A > ALTER TYPE ” 章节。

表 7-75 DROP 语句

序号	Oracle数据库	GaussDB数据库	差异
1	DROP FUNCTION	支持。	-
2	DROP LIBRARY	不支持。	-
3	DROP PACKAGE	支持。	-
4	DROP PROCEDURE	支持。	-
5	DROP TRIGGER	支持，有差异。	GaussDB的语法与Oracle不同。 GaussDB的具体语法请参见《开发者指南》中” SQL参考 > SQL语法 > D > DROP TRIGGER ” 章节。
6	DROP TYPE	支持，有差异。	GaussDB不支持关键字FORCE、VALIDATE。 GaussDB的具体语法请参见《开发者指南》中” SQL参考 > SQL语法 > D > DROP TYPE ” 章节。
7	DROP TYPE BODY	不支持。	-

表 7-76 Function、Procedure、匿名块相关关键字

序号	Oracle数据库	GaussDB数据库	差异
1	ACCESSIBLE BY	不支持。	-
2	AGGREGATE	支持，有差异。	<ul style="list-style-type: none"> <li>GaussDB不支持Oracle的aggregate using [schema.] implementation_type用法。</li> <li>GaussDB的具体用法请参见《开发者指南》中“SQL参考 &gt; SQL语法 &gt; C &gt; CREATE AGGREGATE”章节。</li> </ul> 语法不同，但实现功能相同。
3	DETERMINISTIC	支持，有差异。	GaussDB仅在语法上支持关键字 DETERMINISTIC，未实现功能。
4	PIPE ROW	不支持。	-
5	PIPELINED	不支持。	-
6	SQL_MACRO	不支持。	-
7	RESTRICT_REFERENCES	不支持。	-
8	INLINE	不支持。	-

表 7-77 异常处理相关关键字

序号	Oracle数据库	GaussDB数据库	差异
1	EXCEPTION_INIT	支持，有差异。	GaussDB不支持与系统错误码进行绑定。
2	Exception	支持。	-
3	Exception Handler	支持。	-
4	SQLCODE	支持。	-
5	SQLERRM	支持。	-

表 7-78 其他 PL/SQL 关键字

序号	Oracle数据库	GaussDB数据库	差异
1	COVERAGE	不支持。	-
2	COLLATION	支持。	-

序号	Oracle数据库	GaussDB数据库	差异
3	DEPRECATE	不支持。	-
4	FORALL	支持。	-
5	NOCOPY	不支持。	-
6	RETURNING	支持。	-
7	SERIALLY_REUSABLE	不支持。	-
8	SHARING	不支持。	-
9	BULK COLLECT	支持。	-

## 7.1.10 系统函数

兼容函数分为：单行函数、用户自定义函数、AGG函数、分析函数、对象引用函数、模型函数、OLAP函数。

### 7.1.10.1 单行函数

序号	Oracle数据库函数	GaussDB数据库是否支持
1	<a href="#">数值函数</a>	支持，有差异。
2	<a href="#">返回字符值的字符函数</a>	支持，有差异。
3	<a href="#">返回数值的字符函数</a>	支持，有差异。
4	字符集函数	不支持。
5	Collation函数	不支持。
6	<a href="#">日期时间函数</a>	支持，有差异。
7	<a href="#">通用比较函数</a>	支持，有差异。
8	<a href="#">转换函数</a>	支持，有差异。
9	<a href="#">大对象函数</a>	支持，有差异。
10	集合函数	不支持。
11	<a href="#">层次函数</a>	支持。
12	数据挖掘功能函数	不支持。
13	<a href="#">XML类型函数</a>	支持，有差异。
14	JSON函数	不支持。
15	<a href="#">编码解码函数</a>	支持，有差异。
16	<a href="#">空值相关的函数</a>	支持。



序号	Oracle数据库函数	GaussDB数据库是否支持
17	<a href="#">环境和标识符函数</a>	支持，有差异。

表 7-79 数值函数

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
1	ABS	支持。	-
2	ACOS	支持。	-
3	ASIN	支持。	-
4	ATAN	支持。	-
5	ATAN2	支持。	-
6	BITAND	支持。	-
7	CEIL	支持。	-
8	COS	支持。	-
9	COSH	支持。	-
10	EXP	支持。	-
11	FLOOR	支持。	-
12	LN	支持。	-
13	LOG	支持。	-
14	MOD	支持，有差异。	<ul style="list-style-type: none"> <li>返回类型不一致，Oracle数据库类型包括BINARY_DOUBLE, BINARY_FLOAT, NUMBER; GaussDB返回类型包括int2, int4, int8, numeric。</li> <li>当两个入参中有一个是int类型时，另一个参数必须为int、numeric类型或字面值整数（无小数点的合法数字如：'12'）。</li> </ul>
15	NANVL	支持，有差异。	GaussDB不支持直接声明或浮点数除0得到NaN。
16	POWER	支持。	-
17	REMAINDER	支持，有差异。	当两个入参中有一个是int类型时，另一个参数必须为int、numeric类型或字面值整数(无小数点的合法数字如：'12')。

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
18	ROUND	支持，有差异。	<ul style="list-style-type: none"> <li>第一个参数n的float类型，GaussDB存在精度损失，比Oracle数据库精度低。</li> <li>返回类型不一致。round(n, integer)形式，Oracle数据库NUMBER类型，GaussDB返回numeric类型；round(n)形式，Oracle数据库n的数据类型，GaussDB只能返回float8和numeric类型，缺少float4返回类型。</li> <li>GaussDB判断入参有null，执行框架返回null的逻辑与Oracle数据库不一致。 SELECT round(NULL,'q'); Oracle数据库null，GaussDB报错invalid input syntax for integer: "q"。</li> </ul>
19	SIGN	支持。	-
20	SIN	支持。	-
21	SINH	支持。	-
22	SQRT	支持。	-
23	TAN	支持。	-
24	TANH	支持，有差异。	当传入整数或者加有引号的整数（如：'12'），返回结果不一致。
25	TRUNC	支持。	-
26	WIDTH_BUCKET	支持。	-

表 7-80 返回字符值的字符函数

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
1	CHR	支持，有差异。	<ul style="list-style-type: none"> <li>输入的数字不符合现有字符集时，在JDBC下GaussDB会报错，Oracle数据库会返回乱码。</li> <li>输入0、256等时Oracle数据库会返回Ascii码为0的字符，GaussDB会在'\0'处截断。</li> </ul>
2	CONCAT	支持。	-

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
3	INITCAP	支持，有差异。	返回值受限于数据库字符集，导致返回结果与Oracle数据库不一致。
4	LOWER	支持，有差异。	<ul style="list-style-type: none"> <li>返回值类型不一致，Oracle数据库和输入类型一致的数据类型。</li> <li>对时间格式上隐式转换问题，输入时间类型时，隐式转换为字符串再进行lower操作。 SELECT LOWER(TO_DATE('2012-12-10','YYYY-MM-DD')); Oracle返回10-DEC-12, GaussDB返回2012-12-10 00:00:00。</li> <li>返回值受限于数据库字符集，导致返回结果与Oracle数据库不一致。</li> </ul>
5	LPAD	支持。	-
6	LTRIM	支持，有差异。	返回值类型不一致。 当输入为字符数据类型时，Oracle返回VARCHAR2类型；输入为数据库创建时指定的国家字符集时，Oracle返回NVARCHAR2类型；输入为LOB类型时，Oracle返回LOB类型，GaussDB返回TEXT类型。
7	NCHR	支持，有差异。	<ul style="list-style-type: none"> <li>返回值字节长度与Oracle数据库不一致。</li> <li>返回值受限于数据库字符集，导致返回结果与Oracle数据库不一致。</li> <li>返回入参对应的字节数组时，单个字节在[0x80-0xFF]范围，会返回“？”，Oracle数据库返回“？”、或者不输出、或者会报错。</li> </ul>
8	NLS_LOWER	支持，有差异。	<ul style="list-style-type: none"> <li>返回值类型不同，输入为字符数据类型时，Oracle返回VARCHAR2类型；输入为LOB类型时，Oracle返回LOB类型，GaussDB返回TEXT类型。</li> <li>nlsparam参数Oracle数据库还可以传入除nls_sort外的其他参数种类而不报错，GaussDB只支持nls_sort。</li> <li>返回值受限于数据库字符集，导致返回结果与Oracle数据库不一致。</li> </ul>

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
9	NLS_UPPER	支持，有差异。	<ul style="list-style-type: none"> <li>返回值类型不同，输入为字符数据类型时，Oracle返回VARCHAR2类型；输入为LOB类型时，Oracle返回LOB类型，GaussDB返回TEXT类型。</li> <li>nlsparam参数Oracle数据库还可以传入除nls_sort外的其他参数种类而不报错，GaussDB只支持nls_sort。</li> <li>返回值受限于数据库字符集，导致返回结果与Oracle数据库不一致。</li> </ul>
10	NLSSORT	支持。	-
11	REGEXP_REPLACE	支持，有差异。	<ul style="list-style-type: none"> <li>GaussDB入参source_char不支持NCLOB类型。</li> <li>入参match_param选项'n'含义有差异：GaussDB中'n'选项与'm'选项含义相同，表示采用多行模式匹配；而Oracle表示(.)能匹配'\n'字符，没有指定该选项时默认不能匹配'\n'字符。GaussDB中(.)默认匹配'\n'选项，不需要指定选项。</li> <li>不同正则表达式匹配结果可能不一致。 SELECT REGEXP_REPLACE('abc01234xyz', '(.*?)(\d+)(.*)', '#', 'g') FROM DUAL; Oracle报错，GaussDB返回#####xyz。</li> <li>在UTF-8编码字符集下中文输入时匹配结果可能不一致。Oracle需要在GBK字符集实现中文字符串的正则表达式匹配。</li> <li>包含部分转义字符的正则表达式匹配结果可能不一致。 SELECT REGEXP_REPLACE('abcabc', '\abc', '#', 'g') FROM DUAL; Oracle报错，GaussDB返回abcabc。</li> <li>匹配规则受aformat_regexp_match参数影响，具体影响规格请参见《开发者指南》中“SQL参考 &gt; 函数和操作符 &gt; 字符处理函数和操作符”章节REGEXP_REPLACE函数部分。</li> </ul>
12	REGEXP_SUBSTR	支持，有差异。	<p>匹配规则受aformat_regexp_match参数影响，具体影响规格请参见《开发者指南》中“SQL参考 &gt; 函数和操作符 &gt; 字符处理函数和操作符”章节REGEXP_SUBSTR函数部分。</p>
13	REPLACE	支持。	-

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
14	RPAD	支持。	-
15	RTRIM	支持。	-
16	SUBSTR	支持。	-
17	TRANSLATE	支持。	-
18	TRIM	支持。	-
19	UPPER	支持，有差异。	<ul style="list-style-type: none"> <li>返回值类型不一致，Oracle数据库和输入类型一致的数据类型，GaussDB返回TEXT类型。</li> <li>对时间格式上隐式转换问题，输入时间类型时，隐式转换为字符串再进行upper操作。 SELECT UPPER(TO_DATE('2012-12-10','YYYY-MM-DD')); Oracle返回10-DEC-12，GaussDB返回2012-12-10 00:00:00。</li> <li>返回值受限于数据库字符集，导致返回结果与Oracle数据库不一致。</li> </ul>
20	INSTRB	支持。	-

表 7-81 返回数值的字符函数

序号	Oracle数据库函数	GaussDB数据库函数是否支持	差异
1	ASCII	支持，有差异。	返回值类型不同。Oracle数据库返回类型为uint4，GaussDB为int4。
2	INSTR	支持。	-
3	LENGTH	支持。	-

序号	Oracle数据库函数	GaussDB数据库函数是否支持	差异
4	REGEXP_COUNT	支持，有差异。	<ul style="list-style-type: none"> <li>GaussDB入参source_char不支持NCLOB类型。</li> <li>入参match_param选项'n'含义有差异：GaussDB中'n'选项与'm'选项含义相同，表示采用多行模式匹配；而Oracle表示(.)能匹配'\n'字符，没有指定该选项时默认不能匹配'\n'字符。GaussDB中(.)默认匹配'\n'选项，不需要指定选项。</li> <li>不同正则表达式匹配结果可能不一致。</li> <li>在UTF-8编码字符集下中文输入时匹配结果可能不一致。Oracle需要在GBK字符集实现中文字符串的正则表达式匹配。</li> <li>包含部分转义字符的正则表达式匹配结果可能不一致。</li> <li>匹配规则受aformat_regexp_match参数影响，具体影响规格请参见《开发者指南》中“SQL参考 &gt; 函数和操作符 &gt; 字符处理函数和操作符”章节REGEXP_COUNT函数部分。</li> </ul>
5	REGEXP_INSTR	支持，有差异。	匹配规则受aformat_regexp_match参数影响，具体影响规格请参见《开发者指南》中“SQL参考 > 函数和操作符 > 字符处理函数和操作符”章节REGEXP_INSTR函数部分。

表 7-82 日期时间函数

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
1	ADD_MONTHS	支持，有差异。	<ul style="list-style-type: none"> <li>公元后到公元前，GaussDB会和Oracle数据库相差1年。</li> <li>GaussDB的计算结果范围可以到-4714年，Oracle数据库只到-4713年。</li> </ul>
2	CURRENT_DATE	支持，有差异。	GaussDB不支持nls_date_format参数设置时间显示格式。

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
3	CURRENT_TIMESTAMP	支持，有差异。	Oracle数据库参数支持范围（0~9）。 GaussDB支持范围（0~6）， 微秒末位零不显示。
4	DBTIMEZONE	支持，有差异。	GaussDB不支持自带tz的 timestamp类型接口的调用。
5	EXTRACT	支持。	-
6	LAST_DAY	支持，有差异。	返回值类型不一致。
7	LOCALTIMESTAMP	支持，有差异。	Oracle数据库参数支持范围（0~9）。 GaussDB支持范围（0~6）， 微秒末位零不显示。
8	MONTHS_BETWEEN	支持，有差异。	入参类型不一致。
9	NEW_TIME	支持，有差异。	new_time函数的第一个入参为 字面量时，字面量的格式以及函 数的返回值类型均与Oracle数据 库不一致。
10	NEXT_DAY	支持。	-
11	NUMTODSINTERVAL	支持，有差异。	GaussDB不支持dsinterval类 型，暂时用interval兼容 dsinterval类型。
12	NUMTOYMININTERVAL	支持，有差异。	GaussDB不支持yminterval类 型，暂时用interval兼容 yminterval类型。
13	SESSIONTIMEZONE	支持，有差异。	<ul style="list-style-type: none"> <li>赋值语法差异。GaussDB为 set session time zone 8。 Oracle为alter session set time_zone= '+08:00'。</li> <li>默认值差异。GaussDB为时 区名称形式如:PRC。Oracle 为偏移量形式，如：+08: 00。</li> </ul>
14	SYS_EXTRACT_UTC	支持。	-
15	SYSDATE	支持，有差异。	返回值类型不一致。
16	TO_CHAR	支持，有差异。	fmt '5' 未在Oracle数据库文档 中，未适配。

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
17	TO_DSINTERVAL	支持，有差异。	GaussDB不支持dsinterval类型，暂时用interval兼容dsinterval类型。
18	TO_TIMESTAMP	支持，有差异。	GaussDB毫秒计算只支持6位，Oracle数据库支持9位。
19	TO_TIMESTAMP_TZ	支持，有差异。	GaussDB的timestamptz等价于Oracle的timestampwithlocaltimezone，缺少Oracle对应的timestamptz类型。nls_date_language只支持ENGLISH和AMERICAN两种语言。
20	TO_YMINTERVAL	支持，有差异。	GaussDB不支持yminterval类型，暂时用interval兼容yminterval类型。
21	TZ_OFFSET	支持，有差异。	接收一个时区名称为入参的时候，时区名称的类型比Oracle数据库要少。

表 7-83 通用比较函数

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
1	GREATEST	支持，有差异。	<ul style="list-style-type: none"> <li>GaussDB不支持NLS_SORT参数制定的比较方式，只支持二进制比较。</li> <li>GaussDB不支持多语种的表达式。</li> </ul>
2	LEAST	支持，有差异。	<ul style="list-style-type: none"> <li>GaussDB不支持NLS_SORT参数指定的比较方式，只支持二进制比较。</li> <li>GaussDB不支持多语种的表达式。</li> </ul>

表 7-84 转换函数

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
1	ASCIISTR	支持。	-
2	CAST	支持，有差异。	<ul style="list-style-type: none"> <li>GaussDB不支持multiset子句。</li> <li>GaussDB不支持nlsparam参数。</li> </ul>
3	HEXTORAW	支持。	-



序号	Oracle数据库函数	GaussDB数据库是否支持	差异
4	RAWTOHEX	支持。	-
5	TO_BINARY_DOUBLE	支持，有差异。	GaussDB不支持nlsparam参数。
6	TO_BINARY_FLOAT	支持，有差异。	GaussDB不支持nlsparam参数。
7	TO_BLOB	支持，有差异。	<ul style="list-style-type: none"> <li>GaussDB不支持long raw类型。</li> <li>GaussDB不支持bfile、mime_type类型。</li> </ul>
8	TO_CLOB	支持。	-
9	TO_DATE	支持，有差异。	<ul style="list-style-type: none"> <li>不支持多语种参数。</li> <li>返回类型不一致。</li> <li>缺少控制参数 NLS_DATE_FORMAT。</li> <li>部分format格式不支持。</li> <li>fmt = 'j'。1582年10月15日之前 Oracle数据库与GaussDB输出不一致。</li> <li>无分割符时，不保证与Oracle数据库完全一致。如to_date( '220725', 'yymmdd' ), yy/rr按照固定长度4解析，会解析为2207年25月，25非法月份则会报错。</li> </ul>
10	TO_MULTI_BYTE	支持。	-
11	TO_NCHAR	支持，有差异。	<ul style="list-style-type: none"> <li>GaussDB：将入参的类型转换为text。</li> <li>Oracle：将入参的类型转换为国家字符集（national character set）。</li> </ul>
12	TO_NUMBER	支持，有差异。	GaussDB不支持nlsparam参数。
13	TO_SINGLE_BYTE	支持。	-
14	TREAT	支持，有差异。	GaussDB不支持使用“.”操作符取值，不支持转化为object类型。
15	UNISTR	支持，有差异。	GaussDB只支持UTF-8编码，Oracle数据库支持UTF-8和UTF-16编码。

表 7-85 大对象函数

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
1	EMPTY_BLOB	支持。	-
2	EMPTY_CLOB	支持，有差异。	GaussDB的CLOB类型不支持Oracle数据库中的定位器概念。

表 7-86 层次函数

序号	Oracle数据库函数	GaussDB数据库是否支持
1	SYS_CONNECT_BY_PATH	不支持

表 7-87 XML 类型函数

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
1	EXISTSNODE	支持，有差异。	在入参有命名空间时，xpath和命名空间都需要定义别名。
2	EXTRACTVALUE	支持，有差异。	目前仅支持xpath1.0版本。
3	SYS_XMLAGG	支持，有差异。	xmlagg的别名，可使用xmlagg代替。
4	XMLAGG	支持。	-
5	XMLCOMMENT	支持。	-
6	XMLCONCAT	支持。	-
7	XMLEMENT	支持，有差异。	xmlement和xmlattributes的name字段赋值为NULL时，行为与Oracle不一致。 <ul style="list-style-type: none"> <li>• xmlement的name字段赋值为NULL时，结果显示name信息为空，且不显示属性信息。</li> <li>• xmlattributes的name字段赋值为NULL时，不显示属性信息。</li> </ul>
8	XMLEXISTS	支持，有差异。	GaussDB入参为xml类型。
9	XMLFOREST	支持，有差异。	GaussDB返回值为xml类型。 GaussDB不支持EVALNAME语法。
10	XMLPARSE	支持，有差异。	GaussDB返回值为xml类型。 GaussDB不支持WELLFORMED语法。

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
11	XMLROOT	支持，有差异。	GaussDB返回值为xml类型。
12	JSON_OBJECT	支持。	-
13	XMLTABLE	支持，有差异。	GaussDB从xml中选取数据使用的为XPath 1.0表达式，不支持声明默认命名空间，不支持多组输入及取别名，不支持省略传入数据的passing_clause子句，不支持RETURNING SEQUENCE BY REF子句和( SEQUENCE ) BY REF子句。
14	GETSTRINGVAL	支持	-
15	GETCLOBVAL	支持	-
16	XMLSEQUENCE	支持	-

表 7-88 编码解码函数

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
1	DECODE	支持。	-
2	DUMP	支持，有差异。	因存储格式不同，GaussDB数值和时间类型返回结果和Oracle数据库不一致。如，GaussDB中select dump(123); 返回 Typ=23 Len=4: 123,0,0,0。Oracle中select dump(123) from dual; 返回 Typ=2 Len=3: 194,2,24。
3	ORA_HASH	支持，有差异。	GaussDB中有以下行为： <ul style="list-style-type: none"> <li>时间类型的入参转换成字符串类型再进行hash。</li> <li>不支持maxbucket参数。</li> </ul>
4	VSIZE	支持，有差异。	因存储格式不同，GaussDB数值和时间类型返回结果和Oracle数据库不一致。如GaussDB中select vsize(999); 返回4。Oracle中select vsize(999) from dual; 返回3。

表 7-89 空值相关的函数

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
1	COALESCE	支持。	-
2	LNNVL	支持。	-
3	NULLIF	支持。	-
4	NVL	支持。	-
5	NVL2	支持。	-

表 7-90 环境和标识符函数

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
1	SYS_CONTEXT	支持，有差异。	<p>GaussDB对不支持的参数返回NULL。</p> <p>以下为不支持的参数列表：</p> <ul style="list-style-type: none"> <li>• 'action'</li> <li>• 'is_application_root'</li> <li>• 'is_application_pdb'</li> <li>• 'audited_cursorid'</li> <li>• 'authenticated_identity'</li> <li>• 'authentication_data'</li> <li>• 'authentication_method'</li> <li>• 'cdb_domain'</li> <li>• 'cdb_name'</li> <li>• 'client_identifier'</li> <li>• 'con_id'</li> <li>• 'con_name'</li> <li>• 'current_sql_length'</li> <li>• 'db_domain'</li> <li>• 'db_supplemental_log_level'</li> <li>• 'dblink_info'</li> <li>• 'drain_status'</li> <li>• 'entryid'</li> <li>• 'enterprise_identity'</li> <li>• 'fg_job_id'</li> <li>• 'global_uid'</li> <li>• 'identification_type'</li> <li>• 'instance'</li> <li>• 'is_dg_rolling_upgrade'</li> <li>• 'ldap_server_type'</li> <li>• 'module'</li> <li>• 'network_protocol'</li> <li>• 'nls_calendar'</li> <li>• 'nls_sort'</li> <li>• 'nls_territory'</li> <li>• 'oracle_home'</li> <li>• 'os_user'</li> </ul>

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
			<ul style="list-style-type: none"> <li>• 'platform_slash'</li> <li>• 'policy_invoker'</li> <li>• 'proxy_enterprise_identity'</li> <li>• 'proxy_user'</li> <li>• 'proxy_userid'</li> <li>• 'scheduler_job'</li> <li>• 'session_edition_id'</li> <li>• 'session_edition_name'</li> <li>• 'sessionid'</li> <li>• 'statementid'</li> <li>• 'terminal'</li> <li>• 'unified_audit_sessionid'</li> <li>• 'session_default_collation'</li> <li>• 'client_info'</li> <li>• 'bg_job_id'</li> <li>• 'client_program_name'</li> <li>• 'current_bind'</li> <li>• 'global_context_memory'</li> <li>• 'host'</li> <li>• 'current_sqln'</li> </ul>
2	SYS_GUID	支持。	-
3	USER	支持，有差异。	返回值类型不一致。

### 7.1.10.2 其它函数

序号	Oracle数据库	GaussDB数据库
1	聚合函数	支持
2	分析函数	支持
3	对象引用函数	不支持
4	模型函数	不支持
5	OLAP函数	不支持
6	数据盒功能函数	不支持

序号	Oracle数据库	GaussDB数据库
7	关于用户定义的函数	支持

表 7-91 聚合函数

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
1	AVG	支持。	-
2	CORR	支持。	-
3	COUNT	支持。	-
4	COVAR_POP	支持。	-
5	COVAR_SAMP	支持。	-
6	CUME_DIST	支持。	-
7	DENSE_RANK	支持。	-
8	FIRST	支持。	-
9	GROUPING	支持。	-
10	LAST	支持。	-
11	LISTAGG	支持。	-
12	MAX	支持。	-
13	MEDIAN	支持。	-
14	MIN	支持。	-
15	PERCENT_RANK	支持。	-
16	PERCENTILE_CONT	支持。	-
17	RANK	支持。	-
18	REGR_ (Linear Regression)	支持。	-
19	STDDEV	支持。	-
20	STDDEV_POP	支持。	-
21	STDDEV_SAMP	支持。	-
22	SUM	支持。	-
23	VAR_POP	支持。	-

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
24	VAR_SAMP	支持。	-
25	VARIANCE	支持。	-
26	WM_CONCAT	支持，有差异。	GaussDB中使用LISTAGG、STRING_AGG函数，兼容Oracle此功能。

表 7-92 分析函数

序号	Oracle数据库函数	GaussDB数据库是否支持
1	FIRST_VALUE	支持
2	LAG	支持
3	LAST_VALUE	支持
4	LEAD	支持
5	NTH_VALUE	支持
6	NTILE	支持
7	ROW_NUMBER	支持
8	RATIO_TO_REPORT	支持

## 7.1.11 系统视图

GaussDB数据库兼容了部分Oracle数据库的系统视图，兼容的详细列表如下。  
更多系统视图的字段说明信息请参考《开发者指南》中“系统视图”章节。

表 7-93 系统视图支持列表

序号	Oracle数据库	GaussDB数据库
1	ALL_ALL_TABLES	DB_ALL_TABLES
2	ALL_COL_PRIVS	DB_COL_PRIVS
3	ALL_COLL_TYPES	DB_COLL_TYPES
4	ALL_IND_COLUMNS	DB_IND_COLUMNS
5	ALL_COL_COMMENTS	DB_COL_COMMENTS
6	ALL_CONS_COLUMNS	DB_CONS_COLUMNS
7	ALL_CONSTRAINTS	DB_CONSTRAINTS



序号	Oracle数据库	GaussDB数据库
8	ALL_DEPENDENCIES	DB_DEPENDENCIES
9	ALL_DIRECTORIES	DB_DIRECTORIES
10	ALL_IND_EXPRESSIONS	DB_IND_EXPRESSIONS
11	ALL_IND_PARTITIONS	DB_IND_PARTITIONS
12	ALL_INDEXES	DB_INDEXES
13	ALL_IND_SUBPARTITIONS	DB_IND_SUBPARTITIONS
14	ALL_OBJECTS	DB_OBJECTS
15	ALL_PART_COL_STATISTICS	DB_PART_COL_STATISTICS
16	ALL_PART_KEY_COLUMNS	DB_PART_KEY_COLUMNS
17	ALL_PART_TABLES	DB_PART_TABLES
18	ALL_SCHEDULER_JOB_ARGS	DB_SCHEDULER_JOB_ARGS
19	ALL_SCHEDULER_PROGRAM_ARGS	DB_SCHEDULER_PROGRAM_ARGS
20	ALL_SEQUENCES	DB_SEQUENCES
21	ALL_SUBPART_KEY_COLUMNS	DB_SUBPART_KEY_COLUMNS
22	ALL_SYNONYMS	DB_SYNONYMS
23	ALL_TAB_COL_STATISTICS	DB_TAB_COL_STATISTICS
24	ALL_TAB_COMMENTS	DB_TAB_COMMENTS
25	ALL_TAB_HISTOGRAMS	DB_TAB_HISTOGRAMS
26	ALL_TAB_STATS_HISTORY	DB_TAB_STATS_HISTORY
27	ALL_TYPES	DB_TYPES
28	ALL_PROCEDURES	DB_PROCEDURES
29	ALL_SOURCE	DB_SOURCE
30	ALL_TAB_COLUMNS	DB_TAB_COLUMNS
31	ALL_TAB_PARTITIONS	DB_TAB_PARTITIONS
32	ALL_TAB_SUBPARTITIONS	DB_TAB_SUBPARTITIONS
33	ALL_TABLES	DB_TABLES
34	ALL_TRIGGERS	DB_TRIGGERS
35	ALL_USERS	DB_USERS
36	ALL_VIEWS	DB_VIEWS

序号	Oracle数据库	GaussDB数据库
37	DBA_AUDIT_OBJECT	ADM_AUDIT_OBJECT
38	DBA_AUDIT_SESSION	ADM_AUDIT_SESSION
39	DBA_AUDIT_STATEMENT	ADM_AUDIT_STATEMENT
40	DBA_AUDIT_TRAIL	ADM_AUDIT_TRAIL
41	DBA_COL_COMMENTS	ADM_COL_COMMENTS
42	DBA_COL_PRIVS	ADM_COL_PRIVS
43	DBA_COLL_TYPES	ADM_COLL_TYPES
44	DBA_ARGUMENTS	ADM_ARGUMENTS
45	DBA_CONSTRAINTS	ADM_CONSTRAINTS
46	DBA_DATA_FILES	ADM_DATA_FILES
47	DBA_CONS_COLUMNS	ADM_CONS_COLUMNS
48	DBA_DEPENDENCIES	ADM_DEPENDENCIES
49	DBA_DIRECTORIES	ADM_DIRECTORIES
50	DBA_PART_COL_STATISTICS	ADM_PART_COL_STATISTICS
51	DBA_PART_TABLES	ADM_PART_TABLES
52	DBA_ROLE_PRIVS	ADM_ROLE_PRIVS
53	DBA_ROLES	ADM_ROLES
54	DBA_SCHEDULER_JOB_ARGS	ADM_SCHEDULER_JOB_ARGS
55	DBA_SCHEDULER_PROGRAMS	ADM_SCHEDULER_PROGRAMS
56	DBA_SCHEDULER_PROGRAM_ARGS	ADM_SCHEDULER_PROGRAM_ARGS
57	DBA_HIST_SNAPSHOT	ADM_HIST_SNAPSHOT
58	DBA_HIST_SQL_PLAN	ADM_HIST_SQL_PLAN
59	DBA_HIST_SQLSTAT	ADM_HIST_SQLSTAT
60	DBA_HIST_SQLTEXT	ADM_HIST_SQLTEXT
61	DBA_ILMDATAMOVEMENTPOLICIES	GS_ADM_ILMDATAMOVEMENTPOLICIES
62	DBA_ILMEVALUATIONDETAILS	GS_ADM_ILMEVALUATIONDETAILS
63	DBA_ILMOBJECTS	GS_ADM_ILMOBJECTS
64	DBA_ILMPARAMETERS	GS_ADM_ILMPARAMETERS

序号	Oracle数据库	GaussDB数据库
65	DBA_ILMPOLICIES	GS_ADM_ILMPOLICIES
66	DBA_ILMRESULTS	GS_ADM_ILMRESULTS
67	DBA_ILMTASKS	GS_ADM_ILMTASKS
68	DBA_IND_COLUMNS	ADM_IND_COLUMNS
69	DBA_IND_EXPRESSIONS	ADM_IND_EXPRESSIONS
70	DBA_IND_PARTITIONS	ADM_IND_PARTITIONS
71	DBA_INDEXES	ADM_INDEXES
72	DBA_OBJECTS	ADM_OBJECTS
73	DBA_PART_INDEXES	ADM_PART_INDEXES
74	DBA_PROCEDURES	ADM_PROCEDURES
75	DBA_SCHEDULER_JOBS	ADM_SCHEDULER_JOBS
76	DBA_SCHEDULER_RUNNING_JOBS	ADM_SCHEDULER_RUNNING_JOBS
77	DBA_SEGMENTS	ADM_SEGMENTS
78	DBA_SEQUENCES	ADM_SEQUENCES
79	DBA_SOURCE	ADM_SOURCE
80	DBA_IND_SUBPARTITIONS	ADM_IND_SUBPARTITIONS
81	DBA_SUBPART_KEY_COLUMNS	ADM_SUBPART_KEY_COLUMNS
82	DBA_SYS_PRIVS	ADM_SYS_PRIVS
83	DBA_TAB_COL_STATISTICS	ADM_TAB_COL_STATISTICS
84	DBA_TAB_HISTOGRAMS	ADM_TAB_HISTOGRAMS
85	DBA_TAB_STATISTICS	ADM_TAB_STATISTICS
86	DBA_TAB_STATS_HISTORY	ADM_TAB_STATS_HISTORY
87	DBA_TABLESPACES	ADM_TABLESPACES
88	DBA_TYPES	ADM_TYPES
89	DBA_USERS	ADM_USERS
90	DBA_SYNONYMS	ADM_SYNONYMS
91	DBA_TAB_COLS	ADM_TAB_COLS
92	DBA_TAB_COLUMNS	ADM_TAB_COLUMNS
93	DBA_TAB_COMMENTS	ADM_TAB_COMMENTS

序号	Oracle数据库	GaussDB数据库
94	DBA_TABLES	ADM_TABLES
95	DBA_TAB_PARTITIONS	ADM_TAB_PARTITIONS
96	DBA_TAB_SUBPARTITIONS	ADM_TAB_SUBPARTITIONS
97	DBA_TRIGGERS	ADM_TRIGGERS
98	DBA_TYPE_ATTRS	ADM_TYPE_ATTRS
99	DBA_VIEWS	ADM_VIEWS
100	ROLE_ROLE_PRIVS	ROLE_ROLE_PRIVS
101	ROLE_SYS_PRIVS	ROLE_SYS_PRIVS
102	ROLE_TAB_PRIVS	ROLE_TAB_PRIVS
103	USER_COL_COMMENTS	MY_COL_COMMENTS
104	USER_COL_PRIVS	MY_COL_PRIVS
105	USER_COLL_TYPES	MY_COLL_TYPES
106	USER_CONSTRAINTS	MY_CONSTRAINTS
107	USER_DEPENDENCIES	MY_DEPENDENCIES
108	DICT	DICT
109	DICTIONARY	DICTIONARY
110	DUAL	DUAL
111	NLS_DATABASE_PARAMETERS	NLS_DATABASE_PARAMETERS
112	NLS_INSTANCE_PARAMETERS	NLS_INSTANCE_PARAMETERS
113	PLAN_TABLE	PLAN_TABLE
114	USER_ILMDATAMOVEMENTPOLICIES	GS_MY_ILMDATAMOVEMENTPOLICIES
115	USER_ILMEVALUATIONDETAILS	GS_MY_ILMEVALUATIONDETAILS
116	USER_ILMOBJECTS	GS_MY_ILMOBJECTS
117	USER_ILMPOLICIES	GS_MY_ILMPOLICIES
118	USER_ILMRESULTS	GS_MY_ILMRESULTS
119	USER_ILMTASKS	GS_MY_ILMTASKS
120	USER_IND_COLUMNS	MY_IND_COLUMNS
121	USER_IND_EXPRESSIONS	MY_IND_EXPRESSIONS
122	USER_IND_PARTITIONS	MY_IND_PARTITIONS

序号	Oracle数据库	GaussDB数据库
123	USER_IND_SUBPARTITIONS	MY_IND_SUBPARTITIONS
124	USER_INDEXES	MY_INDEXES
125	USER_JOBS	MY_JOBS
126	USER_OBJECTS	MY_OBJECTS
127	USER_PART_COL_STATISTICS	MY_PART_COL_STATISTICS
128	USER_PART_INDEXES	MY_PART_INDEXES
129	USER_PART_TABLES	MY_PART_TABLES
130	USER_PROCEDURES	MY_PROCEDURES
131	USER_SCHEDULER_JOB_ARGS	MY_SCHEDULER_JOB_ARGS
132	USER_SCHEDULER_PROGRAM_ARGS	MY_SCHEDULER_PROGRAM_ARGS
133	USER_SEQUENCES	MY_SEQUENCES
134	USER_SOURCE	MY_SOURCE
135	USER_SUBPART_KEY_COLUMNS	MY_SUBPART_KEY_COLUMNS
136	USER_SYNONYMS	MY_SYNONYMS
137	USER_SYS_PRIVS	MY_SYS_PRIVS
138	USER_TAB_COL_STATISTICS	MY_TAB_COL_STATISTICS
139	USER_TAB_COLUMNS	MY_TAB_COLUMNS
140	USER_TAB_COMMENTS	MY_TAB_COMMENTS
141	USER_TAB_HISTOGRAMS	MY_TAB_HISTOGRAMS
142	USER_TAB_PARTITIONS	MY_TAB_PARTITIONS
143	USER_TAB_STATISTICS	MY_TAB_STATISTICS
144	USER_TAB_STATS_HISTORY	MY_TAB_STATS_HISTORY
145	USER_TABLES	MY_TABLES
146	USER_TABLESPACES	MY_TABLESPACES
147	USER_TRIGGERS	MY_TRIGGERS
148	USER_TYPE_ATTRS	MY_TYPE_ATTRS
149	USER_TYPES	MY_TYPES
150	USER_VIEWS	MY_VIEWS
151	V\$NLS_PARAMETERS	V\$NLS_PARAMETERS

序号	Oracle数据库	GaussDB数据库
152	V\$SESSION_WAIT	V\$SESSION_WAIT
153	V\$SYSSTAT	V\$SYSSTAT
154	V\$SYSTEM_EVENT	V\$SYSTEM_EVENT
155	V\$VERSION	V\$VERSION
156	V\$INSTANCE	V_INSTANCE
157	GV\$INSTANCE	GV_INSTANCE
158	V\$MYSTAT	V_MYSTAT
159	V\$SESSION	V_SESSION
160	GV\$SESSION	GV_SESSION
161	V\$SESSION_LONGOPS	DV_SESSION_LONGOPS
162	V\$SESSION	DV_SESSIONS
163	ALL_ARGUMENTS	DB_ARGUMENTS
164	USER_CONS_COLUMNS	MY_CONS_COLUMNS
165	USER_PART_KEY_COLUMNS	MY_PART_KEY_COLUMNS
166	USER_ROLE_PRIVS	MY_ROLE_PRIVS
167	DBA_TAB_PRIVS	ADM_TAB_PRIVS
168	USER_SCHEDULER_JOBS	MY_SCHEDULER_JOBS
169	V\$LOCK	V\$LOCK
170	V\$DBLINK	V\$DBLINK
171	V\$GLOBAL_TRANSACTION	V\$GLOBAL_TRANSACTION
172	V\$OPEN_CURSOR	V\$OPEN_CURSOR
173	V\$GLOBAL_OPEN_CURSOR	V\$GLOBAL_OPEN_CURSOR
174	ALL_TAB_PRIVS	DB_TAB_PRIVS
175	ALL_TAB_MODIFICATIONS	DB_TAB_MODIFICATIONS
176	USER_TAB_MODIFICATIONS	MY_TAB_MODIFICATIONS
177	USER_AUDIT_TRAIL	MY_AUDIT_TRAIL

## 7.1.12 高级包

GaussDB数据库兼容了部分Oracle数据库的高级包，兼容的详细列表如下。

更多高级包信息请参考开发者指南中高级包章节。

表 7-94 高级包支持列表

序号	Oracle数据库	GaussDB数据库	差异
1	<b>DBMS_LOB</b>	DBE_LOB	GaussDB具体用法请参见《开发者指南》在“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_LOB”章节。
2	<b>DBMS_RANDOM</b>	DBE_RANDOM	GaussDB具体用法请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_RANDOM”章节。
3	<b>DBMS_OUTPUT</b>	DBE_OUTPUT	GaussDB具体用法请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_OUTPUT”章节。
4	<b>UTL_RAW</b>	DBE_RAW	GaussDB具体用法请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_RAW”章节。
5	<b>DBMS_SCHEDULER</b>	DBE_SCHEDULER	GaussDB具体用法请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_SCHEDULER”章节。
6	<b>DBMS_UTILITY</b>	DBE_UTILITY	GaussDB具体用法请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_UTILITY”章节。

序号	Oracle数据库	GaussDB数据库	差异
7	<b>DBMS_SQL</b>	DBE_SQL	GaussDB具体用法请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_SQL”章节。
8	<b>UTL_FILE</b>	DBE_FILE	GaussDB具体用法请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_FILE”章节。
9	<b>DBMS_SESSION</b>	DBE_SESSION	GaussDB具体用法请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_SESSION”章节。
10	<b>UTL_MATCH</b>	DBE_MATCH	GaussDB具体用法请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_MATCH”章节。
11	<b>DBMS_APPLICATION_INFO</b>	DBE_APPLICATION_INFO	GaussDB具体用法请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_APPLICATION_INFO”章节。
12	<b>DBMS_XMLDOM</b>	DBE_XMLDOM	GaussDB中具体请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_XMLDOM”章节。



序号	Oracle数据库	GaussDB数据库	差异
13	<b>DBMS_XMLPARSER</b>	DBE_XMLPARSER	GaussDB中具体请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_XMLPARSER”章节。
14	<b>DBMS_ILM</b>	DBE_ILM	GaussDB中具体请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_ILM”章节。
15	<b>DBMS_ILM_ADMIN</b>	DBE_ILM_ADMIN	GaussDB中具体请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_ILM_ADMIN”章节。
16	<b>DBMS_COMPRESSION</b>	DBE_COMPRESSION	GaussDB中具体请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_COMPRESSION”章节。
17	<b>DBMS_HEAT_MAP</b>	DBE_HEAT_MAP	GaussDB中具体请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_HEAT_MAP”章节。
18	<b>DBMS_DESCRIBE</b>	DBE_DESCRIBE	GaussDB中具体请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐）” > “DBE_DESCRIBE”章节。

序号	Oracle数据库	GaussDB数据库	差异
19	<a href="#">DBMS_XMLGEN</a>	DBE_XMLGEN	GaussDB中具体请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_XMLGEN”章节。
20	<a href="#">DBMS_STATS</a>	DBE_STATS	GaussDB中具体请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_STATS”章节。

表 7-95 DBMS\_LOB 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	APPEND Procedures	APPEND Procedures	-
2	CLOB2FILE Procedure	不支持	-
3	CLOSE Procedure	BFILECLOSE Procedure	GaussDB: 参数类型为 BFILE, 不存在函数重载。 Oracle: 该过程存在3个重载, 3个重载的参数 lob_loc、lob_loc和 file_loc的类型分别为BLOB、CLOB CHARACTER SET ANY_CS和 BFILE。

序号	Oracle数据库	GaussDB数据库	差异
4	COMPARE Functions	COMPARE Functions	<p>GaussDB: 存在3个重载函数, 对于第三个参数 ( len ) 均为BIGINT。</p> <p>Oracle: 存在3个重载函数, 对于第三个参数 ( amount ) 均为INTEGER。</p>
5	CONVERTTOBLOB Procedure	LOB_CONVERTTOBLOB Procedure	<p>GaussDB: 该过程共有5个参数, 且第3、4、5个参数类型为BIGINT。</p> <p>Oracle: 该过程共有8个参数, 在GaussDB所有参数的基础上增加了blob_csid、lang_context和warning3个参数, 类型分别为NUMBER、INTEGER和INTEGER, 且第3、4、5个参数类型为INTEGER。</p>

序号	Oracle数据库	GaussDB数据库	差异
6	CONVERTTOCLOB Procedure	LOB_CONVERTTOC LOB Procedure	GaussDB: 该过程共有5个参数。第3、4、5个参数类型为BIGINT。 Oracle: 该过程共有8个参数。第3、4、5个参数类型为INTEGER。 Oracle的该过程在GaussDB所有参数的基础上增加了3个参数:blob_csid、lang_context和warning, 参数类型分别为NUMBER、INTEGER和INTEGER。
7	COPY Procedures	LOB_COPY Functions	-
8	COPY_DBFS_LINK Procedures	不支持	-
9	COPY_FROM_DBFS_LINK	不支持	-

序号	Oracle数据库	GaussDB数据库	差异
10	CREATETEMPOracleRY Procedures	CREATE_TEMPORacleRY Procedures	<p>GaussDB: 该过程存在2个重载。第一个重载过程的第一个参数 (lob_loc) 为 BLOB, 第二个重载过程的第一个参数 (lob_loc) 为 CLOB; 两个重载过程的第三个参数 (dur) 为 INTEGER, 默认值为10。</p> <p>Oracle: 该过程存在2个重载。第一个重载过程的第一个参数 (lob_loc) 为 BLOB, 第二个重载过程的第一个参数 (lob_loc) 为 CLOB; 两个重载过程的第三个参数 (dur) 的参数类型为 PLS_INTEGER, 第一个重载过程的dur默认值为 DBMS_LOB.SESSION, 第二个重载过程的dur默认值为 10。</p>
11	DBFS_LINK_GENERATE_PATH Functions	不支持	-
12	ERASE Procedures	LOB_ERASE Procedures	-
13	FILECLOSE Procedure	不支持	-
14	FILECLOSEALL Procedure	不支持	-
15	FILEEXISTS Function	不支持	-
16	FILEGETNAME Procedure	不支持	-

序号	Oracle数据库	GaussDB数据库	差异
17	FILEISOPEN Function	不支持	-
18	FILEOPEN Procedure	不支持	-
19	FRAGMENT_DELETE Procedure	不支持	-
20	FRAGMENT_INSERT Procedures	不支持	-
21	FRAGMENT_MOVE Procedure	不支持	-
22	FRAGMENT_REPLACE Procedures	不支持	-
23	FREETEMPOracleRY Procedures	不支持	-
24	GET_DBFS_LINK Functions	不支持	-
25	GET_DBFS_LINK_STATE Procedures	不支持	-
26	GETCHUNKSIZE Functions	GETCHUNKSIZE Functions	-
27	GETCONTENTTYPE Functions	不支持	-
28	GETLENGTH Functions	不支持	-
29	GETOPTIONS Functions	不支持	-
30	GET_STOracleGE_LIMIT Function	不支持	-
31	INSTR Functions	MATCH Functions	GaussDB: 存在3个重载函数。3个重载函数的第三、四个参数均为BIGINT。 Oracle: 存在3个重载函数。3个重载函数的第三、四个参数均为INTEGER。
32	ISOPEN Functions	不支持	-
33	ISREMOTE Function	不支持	-
34	ISSECUREFILE Function	不支持	-

序号	Oracle数据库	GaussDB数据库	差异
35	ISTEMPOracleRY Functions	不支持	-
36	LOADBLOBFROMFILE Procedure	LOADBLOBFROMFI LE Procedure	-
37	LOADCLOBFROMFILE Procedure	LOADCLOBFROMFI LE Procedure	-
38	LOADFROMFILE Procedure	LOADFROMFILE Procedure	-
39	MOVE_TO_DBFS_LINK Procedures	不支持	-

序号	Oracle数据库	GaussDB数据库	差异
40	OPEN Procedures	BFILEOPEN Procedure	<p>GaussDB: 该过程不存在重载。第一个参数 ( bfile ) 类型为 DBE_LOB.BFILE, 第二个参数 ( open_mode ) 类型为 TEXT, 且只支持read模式。</p> <p>Oracle: 该过程存在3个重载。第一个重载过程的第一个参数 ( lob_loc ) 类型为NOCOPY BLOB, 第二个参数 ( openmode ) 类型为 BINARY_INTEGER; 第二个重载过程的第一个参数 ( lob_loc ) 类型为NOCOPY CLOB CHARACTER SET ANY_CS, 第二个参数 ( openmode ) 类型为 BINARY_INTEGER; 第三个重载过程的第一个参数 ( file_loc ) 类型为NOCOPY BFILE, 第二个参数 ( openmode ) 类型为 BINARY_INTEGER, 且只能为 file_readonly。</p>



序号	Oracle数据库	GaussDB数据库	差异
41	READ Procedures	READ Procedures	GaussDB: 该过程存在2个重载。 Oracle: 该过程存在3个重载。其中前两个重载与 GaussDB无差异, 第三个过程重载包括4个参数: file_loc、amount、offset和 buffer, 其类型分别为 BFILE、NOCOPY INTEGER、INTEGER和 RAW。
42	SET_DBFS_LINK Procedures	不支持	-
43	SETCONTENTTYPE Procedure	不支持	-
44	SETOPTIONS Procedures	不支持	-
45	SUBSTR Functions	LOB_SUBSTR Functions	-
46	TRIM Procedures	STRIP Functions	GaussDB: 该过程存在2个重载。两个重载过程的第二个参数 ( newlen ) 均为BIGINT。 Oracle: 该过程存在2个重载。两个重载过程的第二个参数 ( newlen ) 均为INTEGER。
47	WRITE Procedures	WRITE Functions	-
48	WRITEAPPEND Procedures	WRITEAPPEND Functions	-

表 7-96 DBMS\_RANDOM 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	INITIALIZE Procedure	不支持	-
2	NORMAL Function	不支持	-
3	RANDOM Function	不支持	-
4	SEED Procedures	DBE_RANDOM.SET_SEED Function	GaussDB: 该函数无重载, 参数类型为 INTEGER。 Oracle: 该过程存在2个重载, 2个重载过程的参数类型分别为 VARCHAR2和 BINARY_INTEGER。
5	STRING Function	不支持	-
6	TERMINATE Procedure	不支持	-
7	VALUE Functions	DBE_RANDOM.GET_VALUE Function	GaussDB: 该函数无重载。 Oracle: 存在无参数的 VALUE函数重载, 返回 NUMBER类型。

表 7-97 DBMS\_OUTPUT 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	DISABLE Procedure	DISABLE Function	-
2	ENABLE Procedure	ENABLE Function	-

序号	Oracle数据库	GaussDB数据库	差异
3	GET_LINE Procedure	GET_LINE Function	GaussDB: 该函数无重载, 首个参数 ( lines ) 数据类型为VARCHAR[]。 Oracle: 该过程存在2个重载, 2个重载过程的首个参数 ( lines ) 分别为CHARARR和DBMSOUTPUT_LI NESARRAY。
4	GET_LINES Procedure	GET_LINES Function	-
5	NEW_LINE Procedure	NEW_LINE Function	-

序号	Oracle数据库	GaussDB数据库	差异
6	PUT Procedure	PUT Function	<p>GaussDB: 当数据库服务端字符集 server_encoding 不是UTF8编码格式且入参的字符编码是合法的UTF8编码时, 该函数不会区分入参的数据类型, 都会先把该字符编码按照“UTF8 &gt; server_encoding”的转换关系进行转换后再输出。</p> <p>Oracle: 当数据库服务端字符集 server_encoding 不是UTF8编码格式且入参的字符编码是合法的UTF8编码时, 若入参类型是 NVARCHAR2, 则该过程会先把该字符编码按照“UTF8 &gt; server_encoding”的转换关系进行转换后再输出; 若入参为其他字符类型, 则会将该字符编码视作非法字符, 以占位符的形式输出。</p>

序号	Oracle数据库	GaussDB数据库	差异
7	PUT_LINE Procedure	PUT_LINE Function	<p>GaussDB: 当数据库服务端字符集 server_encoding 不是UTF8编码格式且入参的字符编码是合法的UTF8编码时, 该函数不会区分入参的数据类型, 都会先把该字符编码按照“UTF8 &gt; server_encoding”的转换关系进行转换后再输出。</p> <p>Oracle: 当数据库服务端字符集 server_encoding 不是UTF8编码格式且入参的字符编码是合法的UTF8编码时, 若入参类型是 NVARCHAR2, 则该过程会先把该字符编码按照“UTF8 &gt; server_encoding”的转换关系进行转换后再输出; 若入参为其他字符类型, 则会将该字符编码视作非法字符, 以占位符的形式输出。</p>

表 7-98 UTL\_RAW 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	BIT_AND Function	BIT_AND Function	-
2	BIT_COMPLEMENT Function	BIT_COMPLEMENT Function	-

序号	Oracle数据库	GaussDB数据库	差异
3	BIT_OR Function	BIT_OR Function	GaussDB: 两个参数类型被定义为TEXT类型并且返回TEXT类型。 Oracle: 两个参数为RAW类型并且返回RAW类型。
4	BIT_XOR Function	BIT_XOR Function	-
5	CAST_FROM_BINARY_DOUBLE Function	CAST_FROM_BINARY_DOUBLE_TO_RAW Function	-
6	CAST_FROM_BINARY_FLOAT Function	CAST_FROM_BINARY_FLOAT_TO_RAW Function	GaussDB: 参数n为FLOAT4类型。 Oracle: 参数n为FLOAT类型。
7	CAST_FROM_BINARY_INTEGER Function	CAST_FROM_BINARY_INTEGER_TO_RAW Function	GaussDB: 参数value为BIGINT类型。 Oracle: 参数value为INTEGER类型。
8	CAST_FROM_NUMBER Function	CAST_FROM_NUMBER_TO_RAW Function	GaussDB: 参数n为NUMERIC类型 Oracle: 参数n为NUMBER类型。
9	CAST_TO_BINARY_DOUBLE Function	CAST_FROM_RAW_TO_BINARY_DOUBLE Function	-
10	CAST_TO_BINARY_FLOAT Function	CAST_FROM_RAW_TO_BINARY_FLOAT Function	GaussDB: 函数返回类型为FLOAT4。 Oracle: 函数返回类型为FLOAT。
11	CAST_TO_BINARY_INTEGER Function	CAST_FROM_RAW_TO_BINARY_INTEGER Function	GaussDB: 参数endianess为INTEGER类型, 函数返回类型为INTEGER。 Oracle: 参数endianess为PLS_INTEGER类型, 函数返回类型为BINARY_INTEGER。
12	CAST_TO_NUMBER Function	CAST_FROM_RAW_TO_NUMBER Function	GaussDB: 函数返回类型为NUMERIC。 Oracle: 函数返回类型为NUMBER。
13	CAST_TO_NVARCHAR2 Function	CAST_FROM_RAW_TO_NVARCHAR2 Function	-

序号	Oracle数据库	GaussDB数据库	差异
14	CAST_TO_RAW Function	CAST_FROM_VARCHAR2_TO_RAW Function	-
15	CAST_TO_VARCHAR2 Function	CAST_TO_VARCHAR2 Function	-
16	COMPARE Function	COMPARE Function	GaussDB: 函数返回类型为INTEGER。 Oracle: 函数返回类型为NUMBER。
17	CONCAT Function	CONCAT Function	-
18	CONVERT Function	CONVERT Function	-
19	COPIES Function	COPIES Function	GaussDB: 参数n为NUMERIC类型。 Oracle: 参数n为NUMBER类型。
20	LENGTH Function	GET_LENGTH Function	-
21	OVERLAY Function	OVERLAY Function	-
22	REVERSE Function	REVERSE Function	-
23	SUBSTR Function	SUBSTR Function	GaussDB: 参数lob_loc为BLOB类型; 参数off_set为INTEGER类型, 默认值为1; 参数amount为INTEGER类型, 默认值为32767。 Oracle: 参数r为RAW类型, 参数pos为BINARY_INTEGER类型且无默认值, 参数len为BINARY_INTEGER类型, 默认值为NULL。
24	TRANSLATE Function	TRANSLATE Function	-
25	TRANSLITERATE Function	TRANSLITERATE Function	-
26	XRANGE Function	XRANGE Function	GaussDB: 参数start_byte和end_byte无默认值。 Oracle: 参数start_byte和end_byte默认为NULL。

表 7-99 DBMS\_SCHEDULER 兼容性说明

序号	Oracle数据库	GaussDB数据库
1	ADD_EVENT_QUEUE_SUBSCRIBER Procedure	不支持
2	ADD_GROUP_MEMBER Procedure	不支持
3	ADD_JOB_EMAIL_NOTIFICATION Procedure	不支持
4	ADD_TO_INCOMPATIBILITY Procedure	不支持
5	ALTER_CHAIN Procedure	不支持
6	ALTER_RUNNING_CHAIN Procedure	不支持
7	CLOSE_WINDOW Procedure	不支持
8	COPY_JOB Procedure	不支持
9	CREATE_CHAIN Procedure	不支持
10	CREATE_CREDENTIAL Procedure	CREATE_CREDENTIAL Procedure
11	CREATE_DATABASE_DESTINATION Procedure	不支持
12	CREATE_EVENT_SCHEDULE Procedure	不支持
13	CREATE_FILE_WATCHER Procedure	不支持
14	CREATE_GROUP Procedure	不支持
15	CREATE_INCOMPATIBILITY Procedure	不支持
16	CREATE_JOB Procedure	CREATE_JOB Procedure
17	CREATE_JOB_CLASS Procedure	CREATE_JOB_CLASS Procedure
18	CREATE_JOBS Procedure	不支持
19	CREATE_PROGRAM Procedure	CREATE_PROGRAM Procedure
20	CREATE_RESOURCE Procedure	不支持



序号	Oracle数据库	GaussDB数据库
21	CREATE_SCHEDULE Procedure	CREATE_SCHEDULE Procedure
22	CREATE_WINDOW Procedure	不支持
23	DEFINE_ANYDATA_ARGUMENT Procedure	不支持
24	DEFINE_CHAIN_EVENT_STEP Procedure	不支持
25	DEFINE_CHAIN_RULE Procedure	不支持
26	DEFINE_CHAIN_STEP Procedure	不支持
27	DEFINE_METADATA_ARGUMENT Procedure	不支持
28	DEFINE_PROGRAM_ARGUMENT Procedure	DEFINE_PROGRAM_ARGUMENT Procedure
29	DISABLE Procedure	DISABLE Procedure
30	DROP_AGENT_DESTINATION Procedure	不支持
31	DROP_CHAIN Procedure	不支持
32	DROP_CHAIN_RULE Procedure	不支持
33	DROP_CHAIN_STEP Procedure	不支持
34	DROP_CREDENTIAL Procedure	DROP_CREDENTIAL Procedure
35	DROP_DATABASE_DESTINATION Procedure	不支持
36	DROP_FILE_WATCHER Procedure	不支持
37	DROP_GROUP Procedure	不支持
38	DROP_INCOMPATIBILITY Procedure	不支持
39	DROP_JOB Procedure	DROP_JOB Procedure
40	DROP_JOB_CLASS Procedure	DROP_JOB_CLASS Procedure
41	DROP_PROGRAM Procedure	DROP_PROGRAM Procedure
42	DROP_PROGRAM_ARGUMENT Procedure	不支持

序号	Oracle数据库	GaussDB数据库
43	DROP_SCHEDULE Procedure	DROP_SCHEDULE Procedure
44	DROP_WINDOW Procedure	不支持
45	ENABLE Procedure	ENABLE Procedure
46	END_DETACHED_JOB_RUN Procedure	不支持
47	EVALUATE_CALENDAR_STRING Procedure	EVALUATE_CALENDAR_STRING Procedure
48	EVALUATE_RUNNING_CHAIN Procedure	不支持
49	GENERATE_JOB_NAME Function	GENERATE_JOB_NAME Function
50	GET_AGENT_INFO Function	不支持
51	GET_AGENT_VERSION Function	不支持
52	GET_ATTRIBUTE Procedure	不支持
53	GET_FILE Procedure	不支持
54	GET_SCHEDULER_ATTRIBUTE Procedure	不支持
55	OPEN_WINDOW Procedure	不支持
56	PURGE_LOG Procedure	不支持
57	PUT_FILE Procedure	不支持
58	REMOVE_EVENT_QUEUE_SUBSCRIBER Procedure	不支持
59	REMOVE_FROM_INCOMPATIBILITY Procedure	不支持
60	REMOVE_GROUP_MEMBER Procedure	不支持
61	REMOVE_JOB_EMAIL_NOTIFICATION Procedure	不支持
62	RESET_JOB_ARGUMENT_VALUE Procedure	不支持
63	RUN_CHAIN Procedure	不支持
64	RUN_JOB Procedure	RUN_JOB Procedure
65	SET_AGENT_REGISTRATION_PASSWORD Procedure	不支持

序号	Oracle数据库	GaussDB数据库
66	SET_ATTRIBUTE Procedure	SET_ATTRIBUTE Procedure
67	SET_ATTRIBUTE_NULL Procedure	不支持
68	SET_JOB_ANYDATA_VALUE Procedure	不支持
69	SET_JOB_ARGUMENT_VALUE Procedure	SET_JOB_ARGUMENT_VALUE Procedure
70	SET_JOB_ATTRIBUTES Procedure	不支持
71	SET_RESOURCE_CONSTRAINT Procedure	不支持
72	SET_SCHEDULER_ATTRIBUTE Procedure	不支持
73	STOP_JOB Procedure	STOP_JOB Procedure

表 7-100 DBMS\_UTILITY 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	ACTIVE_INSTANCES Procedure	不支持	-
2	ANALYZE_DATABASE Procedure	不支持	-
3	ANALYZE_PART_OBJECT Procedure	不支持	-
4	ANALYZE_SCHEMA Procedure	不支持	-
5	CANONICALIZE Procedure	CANONICALIZE Procedure	GaussDB: 参数 canon_len默认为1024字节。 Oracle: 参数 canon_len无默认值。

序号	Oracle数据库	GaussDB数据库	差异
6	COMMA_TO_TABLE Procedures	COMMA_TO_TABLE Procedures	GaussDB: 参数tab 为VARCHAR2数 组。  Oracle: 该过程存 在2个重载。参数 tab可以为两种类型 之一: 一种为 uncl_array, 另一种 为lname_array。
7	COMPILE_SCHEMA Procedure	不支持	-
8	CREATE_ALTER_TYPE_ER ROR_TABLE Procedure	不支持	-
9	CURRENT_INSTANCE Function	不支持	-
10	DATA_BLOCK_ADDRESS_ BLOCK Function	不支持	-
11	DATA_BLOCK_ADDRESS_ FILE Function	不支持	-
12	DB_VERSION Procedure	DB_VERSION Procedure	GaussDB: 只有参 数version, 类型为 VARCHAR2。 Oracle: 有参数 version和 compatibility, 类 型均为 VARCHAR2。
13	EXEC_DDL_STATEMENT Procedure	EXEC_DDL_STATEM ENT Function	GaussDB: 参数 parse_string为TEXT 类型。  Oracle: 参数 parse_string为 VARCHAR2类型。
14	EXPAND_SQL_TEXT Procedure	EXPAND_SQL_TEXT Function	GaussDB: 参数 output_sql_text为 CLOB。  Oracle: 参数 output_sql_text为 NOCOPY CLOB, 通 过传引用方式传递 OUT参数。

序号	Oracle数据库	GaussDB数据库	差异
15	FORMAT_CALL_STACK Function	FORMAT_CALL_STACK Function	GaussDB: 函数返回类型为TEXT。 Oracle: 函数返回类型为VARCHAR2。
16	FORMAT_ERROR_BACKTRACE Function	FORMAT_ERROR_BACKTRACE Function	GaussDB: 函数返回类型为TEXT。 Oracle: 函数返回类型为VARCHAR2。
17	FORMAT_ERROR_STACK Function	FORMAT_ERROR_STACK Function	GaussDB: 函数返回类型为TEXT。 Oracle: 函数返回类型为VARCHAR2。
18	GET_CPU_TIME Function	GET_CPU_TIME Function	GaussDB: 函数返回类型为BIGINT。 Oracle: 函数返回类型为NUMBER。
19	GET_DEPENDENCY Procedure	不支持	-
20	GET_ENDIANNESS Function	GET_ENDIANNESS Function	GaussDB: 函数返回类型为INTEGER。 Oracle: 函数返回类型为NUMBER。
21	GET_HASH_VALUE Function	GET_HASH_VALUE Function	GaussDB: 参数base、hash_size和返回类型均为INTEGER。 Oracle: 参数base、hash_size和返回类型均为NUMBER。
22	GET_PARAMETER_VALUE Function	不支持	-

序号	Oracle数据库	GaussDB数据库	差异
23	GET_SQL_HASH Function	GET_SQL_HASH Function	GaussDB: 参数 last4bytes 为 BIGINT 类型, 代表 MD5 哈希值的最后四字节, 以无符号整数形式展现, 函数返回类型为 BIGINT。 Oracle: 对应参数 pre10ihash 为 NUMBER 类型, 用于存储 MD5 计算得到的 16 字节中的 4 字节哈希值。
24	GET_TIME Function	GET_TIME Function	GaussDB: 函数返回类型为 BIGINT。 Oracle: 函数返回类型为 NUMBER。
25	GET_TZ_TRANSITIONS Procedure	不支持	-
26	INVALIDATE Procedure	不支持	-
27	IS_BIT_SET Function	IS_BIT_SET Function	GaussDB: 参数 n 和返回值类型为 INTEGER。 Oracle: 参数 n 和返回值类型为 NUMBER。
28	IS_CLUSTER_DATABASE Function	IS_CLUSTER_DATABASE Function	-
29	MAKE_DATA_BLOCK_ADDRESS Function	不支持	-
30	NAME_RESOLVE Procedure	NAME_RESOLVE Procedure	GaussDB: 参数 context 和 part1_type 为 INTEGER, 参数 object_number 为 OID; GaussDB 不支持 NUMBER 到 OID 的隐式转换。 Oracle: 参数 context、part1_type 和 object_number 均为 NUMBER。

序号	Oracle数据库	GaussDB数据库	差异
31	NAME_TOKENIZE Procedure	NAME_TOKENIZE Procedure	GaussDB: 参数 nextpos为INTEGER 类型。 Oracle: 参数 nextpos为 BINARY_INTEGER 类型。
32	OLD_CURRENT_SCHEMA Function	OLD_CURRENT_SCHEMA Function	GaussDB: 函数返回类型为 VARCHAR。 Oracle: 函数返回类型为 VARCHAR2。
33	OLD_CURRENT_USER Function	OLD_CURRENT_USER Function	GaussDB: 函数返回类型为TEXT。 Oracle: 函数返回类型为 VARCHAR2。
34	PORT_STRING Function	不支持	-
35	SQLID_TO_SQLHASH Function	不支持	-
36	TABLE_TO_COMMA Procedures	TABLE_TO_COMMA Procedures	GaussDB: 参数tab为VARCHAR2数组。 Oracle: 该存储过程存在2个重载。参数tab可以为两种类型之一: 一种为 uncl_array, 另一种为lname_array。
37	VALIDATE Procedure	不支持	-
38	WAIT_ON_PENDING_DML Function	不支持	-

表 7-101 DBMS\_SQL 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	BIND_ARRAY Procedures	SQL_BIND_ARRAY Function	-
2	BIND_VARIABLE Procedures	SQL_BIND_VARIABLE Function	-

序号	Oracle数据库	GaussDB数据库	差异
3	BIND_VARIABLE_PKG Procedure	不支持	-
4	CLOSE_CURSOR Procedure	SQL_UNREGISTER_C ONTEXT Function	-
5	COLUMN_VALUE Procedure	GET_RESULT Procedure	-
6	COLUMN_VALUE_LONG Procedure	不支持	-
7	DEFINE_ARRAY Procedure	SET_RESULTS_TYPE Procedure	-
8	DEFINE_COLUMN Procedures	SET_RESULT_TYPE Procedure	-
9	DEFINE_COLUMN_CHAR Procedure	不支持	-
10	DEFINE_COLUMN_LONG Procedure	不支持	-
11	DEFINE_COLUMN_RAW Procedure	不支持	-
12	DEFINE_COLUMN_ROWID Procedure	不支持	-
13	DESCRIBE_COLUMNS Procedure	DESCRIBE_COLUMNS Procedure	-
14	DESCRIBE_COLUMNS2 Procedure	不支持	-
15	DESCRIBE_COLUMNS3 Procedure	不支持	-
16	EXECUTE Function	SQL_RUN Function	-
17	EXECUTE_AND_FETCH Function	RUN_AND_NEXT Function	-
18	FETCH_ROWS Function	NEXT_ROW Function	-
19	GET_NEXT_RESULT Procedures	不支持	-
20	IS_OPEN Function	IS_ACTIVE Function	-
21	LAST_ERROR_POSITION Function	不支持	-
22	LAST_ROW_COUNT Function	LAST_ROW_COUNT Function	-



序号	Oracle数据库	GaussDB数据库	差异
23	LAST_ROW_ID Function	不支持	-
24	LAST_SQL_FUNCTION_CODE Function	不支持	-
25	OPEN_CURSOR Functions	REGISTER_CONTEXT Function	-
26	PARSE Procedures	支持，有差异。	GaussDB中为SQL_SET_SQL Function，不支持重载。
27	RETURN_RESULT Procedures	不支持	-
28	TO_CURSOR_NUMBER Function	不支持	-
29	TO_REFCURSOR Function	不支持	-
30	VARIABLE_VALUE Procedures	GET_VARIABLE_RESULT Procedures	-
31	VARIABLE_VALUE_PKG Procedure	不支持	-

表 7-102 DBMS\_SQL 数据类型兼容性说明

序号	Oracle数据库	GaussDB数据库
1	DBMS_SQL_DESC_REC	DBE_SQL.DESC_REC
2	DBMS_SQL_DATE_TABLE	DBE_SQL.DATE_TABLE
3	DBMS_SQL_NUMBER_TABLE	DBE_SQL.NUMBER_TABLE
4	DBMS_SQL_VARCHAR2_TABLE	DBE_SQL.VARCHAR2_TABLE
5	DBMS_SQL_BLOB_TABLE	DBE_SQL.BLOB_TABLE

表 7-103 UTL\_FILE 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	FCLOSE Procedure	CLOSE Procedure	-

序号	Oracle数据库	GaussDB数据库	差异
2	FCLOSE_ALL Procedure	CLOSE_ALL Procedure	-
3	FCOPY Procedure	COPY Procedure	-
4	FFLUSH Procedure	FLUSH Procedure	-
5	FGETATTR Procedure	GET_ATTR Procedure	-
6	FGETPOS Function	GET_POS Function	-
7	FOPEN Function	FOPEN Function	-
8	FOPEN_NCHAR Function	FOPEN_NCHAR Function	-
9	FREMOVE Procedure	REMOVE Procedure	-
10	FRENAME Procedure	RENAME Procedure	-
11	FSEEK Procedure	SEEK Procedure	-
12	GET_LINE Procedure	READ_LINE Procedure	-
13	GET_LINE_NCHAR Procedure	READ_LINE_NCHAR Procedure	-
14	GET_RAW Procedure	GET_RAW Procedure	-
15	IS_OPEN Function	IS_OPEN Function	-
16	NEW_LINE Procedure	支持，有差异，NEW_LINE Function	GaussDB因性能原因将接口定义为Function。
17	PUT Procedure	支持，有差异，WRITE Function	GaussDB因性能原因将接口定义为Function。
18	PUT_LINE Procedure	支持，有差异，WRITE_LINE Function	GaussDB因性能原因将接口定义为Function。
19	PUT_LINE_NCHAR Procedure	支持，有差异，WRITE_LINE_NCHAR Function	GaussDB因性能原因将接口定义为Function。
20	PUT_NCHAR Procedure	支持，有差异，WRITE_NCHAR Function	GaussDB因性能原因将接口定义为Function。
21	PUTF Procedure	支持，有差异，FORMAT_WRITE Function	GaussDB因性能原因将接口定义为Function。
22	PUTF_NCHAR Procedure	支持，有差异，FORMAT_WRITE_NCHAR Function	GaussDB因性能原因将接口定义为Function。
23	PUT_RAW Procedure	支持，有差异，PUT_RAW Function	GaussDB因性能原因将接口定义为Function。

表 7-104 DBMS\_SESSION 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	CLEAR_ALL_CONTEXT Procedure	不支持	-
2	CLEAR_CONTEXT Procedure	CLEAR_CONTEXT Function	-
3	CLEAR_IDENTIFIER Procedure	不支持	-
4	CLOSE_DATABASE_LINK Procedure	不支持	-
5	CURRENT_IS_ROLE_ENABLED Function	不支持	-
6	FREE_UNUSED_USER_MEMORY Procedure	不支持	-
7	GET_PACKAGE_MEMORY_UTILIZATION Procedure	不支持	-
8	IS_ROLE_ENABLED Function	不支持	-
9	IS_SESSION_ALIVE Function	不支持	-
10	LIST_CONTEXT Procedures	不支持	-
11	MODIFY_PACKAGE_STATE Procedure	MODIFY_PACKAGE_STATE Procedure	GaussDB: 仅支持入参 flags = 1的场景使用。 Oracle: 支持flags=1或 flags = 2的场景使用。
12	RESET_PACKAGE Procedure	不支持	-
13	SESSION_IS_ROLE_ENABLED Function	不支持	-
14	SESSION_TRACE_DISABLE Procedure	不支持	-
15	SESSION_TRACE_ENABLE Procedure	不支持	-

序号	Oracle数据库	GaussDB数据库	差异
16	SET_CONTEXT Procedure	SET_CONTEXT Function	GaussDB: 仅包括参数 namespace, attribute和 value, 类型均为text。 Oracle: 包括参数 namespace, attribute, value, username和 client_id, 类型均为 VARCHAR2。
17	SET_EDITION_DEFERRED Procedure	不支持	-
18	SET_IDENTIFIER Procedure	不支持	-
19	SET-NLS Procedure	不支持	-
20	SET_ROLE Procedure	不支持	-
21	SET_SQL_TRACE Procedure	不支持	-
22	SLEEP Procedure	不支持	-
23	SWITCH_CURRENT_CONSUMER_GROUP Procedure	不支持	-
24	UNIQUE_SESSION_ID Function	不支持	-

表 7-105 UTL\_MATCH 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	EDIT_DISTANCE Function	不支持	-
2	EDIT_DISTANCE_SIMILARITY Function	EDIT_DISTANCE_SIMILARITY Function	GaussDB: 参数str1和str2均为TEXT类型, 函数返回类型为INTEGER。 Oracle: 参数s1和s2为VARCHAR2类型, 函数返回类型为PLS_INTEGER。
3	JARO_WINKLER Function	不支持	-
4	JARO_WINKLER_SIMILARITY Function	不支持	-

表 7-106 DBMS\_APPLICATION\_INFO 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	READ_CLIENT_INFO Function	READ_CLIENT_INFO Procedure	GaussDB: 参数client_info为TEXT类型。 Oracle: 参数client_info为VARCHAR2类型。
2	READ_MODULE Procedure	READ_MODULE Procedure	GaussDB: 参数module_name、action_name为TEXT类型。 Oracle: 参数module_name、action_name为VARCHAR2类型。
3	SET_ACTION Procedure	SET_ACTION Procedure	GaussDB: 参数action_name为TEXT类型。 Oracle: 参数action_name为VARCHAR2类型。
4	SET_CLIENT_INFO Function	SET_CLIENT_INFO Procedure	GaussDB: 参数str为TEXT类型, 且返回类型为void。 Oracle: 参数client_info为VARCHAR2类型, 无返回值。二者均为写入客户端信息, 最大输入64字节, 超过64字节将被截断。
5	SET_MODULE Procedure	SET_MODULE Procedure	GaussDB: 参数module_name、action_name为TEXT类型。 Oracle: 参数module_name、action_name为VARCHAR2类型。
6	SET_SESSION_LONGOPS Procedure	不支持	-

表 7-107 DBMS\_XMLDOM 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	DBMS_XMLDOM.APPENDCHILD	DBE_XMLDOM.APPENDCHILD	<ul style="list-style-type: none"> <li>GaussDB: DOCUMENT类型节点下APPEND ATTR类型节点会报“operation not support”错误。 Oracle: 在此场景下不报错, 但实际并没有挂载成功。</li> <li>GaussDB: ATTR类型节点下APPEND ATTR类型节点会报“operation not support”错误。 Oracle: 在此场景下不报错, 但实际并没有挂载成功。</li> <li>GaussDB: 父节点在添加多个ATTR类型子节点时, 不允许KEY值相同的子节点同时存在于同一个父节点下。 Oracle: 允许KEY值相同的子节点同时存在于同一个父节点下。</li> </ul>
2	DBMS_XMLDOM.CREATEELEMENT	DBE_XMLDOM.CREATEELEMENT	-
3	DBMS_XMLDOM.CREATETEXTNODE	DBE_XMLDOM.CREATETEXTNODE	-
4	DBMS_XMLDOM.FREEDOCUMENT	支持, 有差异, DBE_XMLDOM.FREEDOCUMENT	<p>GaussDB: 释放时不会立刻释放对象, 累积一定数量后释放。document下全部节点失效。</p> <p>Oracle: 立即释放对象。</p>
5	DBMS_XMLDOM.FREEELEMENT	DBE_XMLDOM.FREEELEMENT	-
6	DBMS_XMLDOM.FREENODE	DBE_XMLDOM.FREENODE	-
7	DBMS_XMLDOM.FREENODELIST	支持, 有差异, DBE_XMLDOM.FREENODELIST	<p>GaussDB: nodelist会被释放。</p> <p>Oracle: 释放nodelist后, 在原始的doc中还能被查询到。</p>
8	DBMS_XMLDOM.GETATTRIBUTE	DBE_XMLDOM.GETATTRIBUTE	-

序号	Oracle数据库	GaussDB数据库	差异
9	DBMS_XMLDOM.GETATTRIBUTES	DBE_XMLDOM.GETATTRIBUTES	-
10	DBMS_XMLDOM.GETCHILDNODES	DBE_XMLDOM.GETCHILDNODES	GaussDB: 对document的node使用时会包含dtd Oracle: 不包含dtd。
11	DBMS_XMLDOM.GETCHILDRENBYTAGNAME	DBE_XMLDOM.GETCHILDRENBYTAGNAME	GaussDB: DBE_XMLDOM.GETCHILDRENBYTAGNAME接口的参数ns不支持传入参数"*", 如需获取节点下全部属性, 可使用DBE_XMLDOM.GETCHILDNODES接口。 Oracle: 支持传入参数"*"。
12	DBMS_XMLDOM.GETDOCUMENTELEMENT	DBE.XMLDOM.GETDOCUMENTELEMENT	-
13	DBMS_XMLDOM.GETFIRSTCHILD	DBE_XMLDOM.GETFIRSTCHILD	-
14	DBMS_XMLDOM.GETLASTCHILD	DBE_XMLDOM.GETLASTCHILD	-
15	DBMS_XMLDOM.GETLENGTH	DBE_XMLDOM.GETLENGTH	-
16	DBMS_XMLDOM.GETLOCALNAME	DBE_XMLDOM.GETLOCALNAME	-
17	DBMS_XMLDOM.GETNAMEDITEM	DBE_XMLDOM.GETNAMEDITEM	-
18	DBMS_XMLDOM.GETNEXTSIBLING	DBE_XMLDOM.GETNEXTSIBLING	-
19	DBMS_XMLDOM.GETNODENAME	DBE_XMLDOM.GETNODENAME	-
20	DBMS_XMLDOM.GETNODETYPE	DBE_XMLDOM.GETNODETYPE	-

序号	Oracle数据库	GaussDB数据库	差异
21	DBMS_XMLDOM.GETTAGNAME	DBE_XMLDOM.GETTAGNAME	-
22	DBMS_XMLDOM.IMPORTNODE	DBE_XMLDOM.IMPORTNODE	-
23	DBMS_XMLDOM.ISNULL	DBE_XMLDOM.ISNULL	GaussDB: 入参为DOMNODELIST类型时, 若对象在哈希表中不存在会发生报错。 Oracle: 不会报错。
24	DBMS_XMLDOM.ITEM	DBE_XMLDOM.ITEM	-
25	DBMS_XMLDOM.MAKENODE	DBE_XMLDOM.MAKENODE	GaussDB: 该函数不支持直接作为函数返回值返回。 Oracle: 支持直接作为函数返回值返回。



序号	Oracle数据库	GaussDB数据库	差异
26	DBMS_XMLDOM.NEWDOMDOCUMENT	DBE_XMLDOM.NEWDOMDOCUMENT	<ul style="list-style-type: none"> <li>GaussDB入参大小需限制1GB以内。 Oracle: 与CLOB类型大小一致。</li> <li>GaussDB 目前暂不支持外部DTD解析。 Oracle: 支持解析外部DTD。</li> <li>GaussDB newdomdocument创建的doc, 默认UTF-8字符集。 Oracle: 根据服务端字符集生成。</li> <li>GaussDB 从同一个xmltype实例中解析出的每一个doc都是独立的, 对doc的修改也不会影响到xmltype。 Oracle: 从同一个xmltype实例中解析出的每一个doc不独立, 有关联关系。</li> <li>GaussDB version字段只支持1.0, 1.0-1.9解析警告但正常执行, 1.9以上报错。 Oracle: 不报错。</li> <li>GaussDB 与Oracle数据库DTD校验差异: !ATTLIST to type (CHECK check Check) "Ch..."将报错, 因默认值"Ch..."不属于括号中枚举值, 而Oracle数据库不报错。&lt;!ENTITY baidu "www.baidu.com"&gt;.....&amp;Baidu;&amp;writer将报错, 因区分字母大小写, Baidu无法与baidu对应。 Oracle: 不报错。</li> <li>GaussDB 与Oracle数据库命名空间校验差异: 解析未声明的命名空间标签正常执行。 Oracle: 报错。</li> </ul>
27	DBMS_XMLDOM.SETATTRIBUTE	DBE_XMLDOM.SETATTRIBUTE	<p>GaussDB: 属性key不支持为null或空字符串。 Oracle: 属性key允许为null或空字符串。</p>

序号	Oracle数据库	GaussDB数据库	差异
28	DBMS_XMLDOM.SETCHARSET	DBE_XMLDOM.SETCHARSET	GaussDB目前支持的字符集有：UTF-8、UTF-16、UCS-4、UCS-2、ISO-8859-1、ISO-8859-2、ISO-8859-3、ISO-8859-4、ISO-8859-5、ISO-8859-6、ISO-8859-7、ISO-8859-8、ISO-8859-9、ISO-2022-JP、Shift_JIS、EUC-JP、ASCII。输入其他字符集会报错或者可能导致输出乱码
29	DBMS_XMLDOM.SETDOCTYPE	DBE_XMLDOM.SETDOCTYPE	GaussDB name、sysid、pubid的总长度限制在32500个字节以内。 Oracle：限制在32767字节内。
30	DBMS_XMLDOM.WRITETOBUFFER	支持，有差异，DBE_XMLDOM.WRITETOBUFFER	<ul style="list-style-type: none"> <li>GaussDB：writetobuffer输出buffer限制在1GB以内。 Oracle：限制在32767字节内。</li> <li>GaussDB：输出doc将包含XML声明version和encoding。 Oracle：用户不主动指定将不包含。</li> <li>GaussDB：入参为domnode类型时，如果节点是doc转换的，输出节点将包含XML声明version和encoding。 Oracle：用户不主动指定将不包含。</li> <li>GaussDB：默认以UTF-8字符集输出xml。 Oracle：根据数据库字符集生成。</li> </ul>

序号	Oracle数据库	GaussDB数据库	差异
31	DBMS_XMLDOM.WRITETOCLOB	DBE_XMLDOM.WRITETOCLOB	<ul style="list-style-type: none"><li>• GaussDB: writetoclob大小支持1GB以内。 Oracle: 按CLOB大小支持。</li><li>• GaussDB: 输出doc将包含XML声明version和encoding。 Oracle: 用户不主动指定将不包含。</li><li>• GaussDB: 入参为domnode类型时, 如果节点是doc转换的, 输出节点将包含XML声明version和encoding。 Oracle: 用户不主动指定将不包含。</li><li>• GaussDB 默认以UTF-8字符集输出xml。 Oracle: 根据数据库字符集生成。</li></ul>

序号	Oracle数据库	GaussDB数据库	差异
32	DBMS_XMLDOM.WRITETOFILE	DBE_XMLDOM.WRITETOFILE	<ul style="list-style-type: none"> <li>GaussDB: document入参, filename长度限制在255个字节以内, charset请参考 dbe_xmldom.setcharset接口。Oracle: filename长度限制受操作系统影响, 大于255个字节。</li> <li>GaussDB: domnode入参, filename长度限制在255个字节以内, charset请参考 dbe_xmldom.setcharset接口。Oracle: filename长度限制受操作系统影响, 大于255个字节。</li> <li>GaussDB: 该函数会添加缩进等内容, 将输出格式化。输出doc将包含XML声明version和 encoding。入参为domnode类型时, 如果节点是doc转换的, 输出节点将包含XML声明version和 encoding。Oracle: 用户不主动指定将不包含。</li> <li>GaussDB: 传入 newdomdocument()无参创建的 doc, 在不指定charset时不会报错, 默认UTF-8字符集。Oracle: 会进行报错。</li> <li>GaussDB: filename需要在 pg_directory中创建的路径下, filename中的\会被转换成/, 只允许存在一个/。文件名格式应为 pg_directory_name/file_name。Oracle: 按用户输入不进行转义。</li> </ul>
33	DBMS_XMLDOM.GETNODEVALUE	DBE_XMLDOM.GETNODEVALUE	-
34	DBMS_XMLDOM.GETPARENTNODE	DBE_XMLDOM.GETPARENTNODE	-
35	DBMS_XMLDOM.HASCHILDNODES	DBE_XMLDOM.HASCHILDNODES	-
36	DBMS_XMLDOM.MAKEELEMENT	DBE_XMLDOM.MAKEELEMENT	-

序号	Oracle数据库	GaussDB数据库	差异
37	DBMS_XMLDOM.SETNODEVALUE	DBE_XMLDOM.SETNODEVALUE	<ul style="list-style-type: none"> <li>GaussDB: nodeValue入参, 可以输入空字符串和NULL值, 不会对节点值进行修改。</li> <li>Oracle: 空字符串和NULL会将节点值修改为空字符串。</li> <li>GaussDB: nodeValue入参, 暂不支持转义字符'&amp;', 如字符串中包含该转义字符, 会清空节点值。</li> <li>Oracle: 支持转义字符。</li> </ul>
38	DBMS_XMLDOM.GETELEMENTSBYTAGNAME	DBE_XMLDOM.GETELEMENTSBYTAGNAME	-

表 7-108 DBMS\_XMLPARSER 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	DBMS_XMLPARSER.FREEPARSER	DBE_XMLPARSER.FREEPARSER	-
2	DBMS_XMLPARSER.GETDOCUMENT	DBE_XMLPARSER.GETDOCUMENT	-
3	DBMS_XMLPARSER.GETVALIDATIONMODE	DBE_XMLPARSER.GETVALIDATIONMODE	-
4	DBMS_XMLPARSER.NEWPARSER	支持, 有差异, DBE_XMLPARSER.NEWPARSER	GaussDB中parser对象的数量上限为16777215, Oracle中约为1亿。

序号	Oracle数据库	GaussDB数据库	差异
5	DBMS_XMLPARSER.PARSEBUFFER	支持，有差异，DBE_XMLPARSER.PARSEBUFFER	<ul style="list-style-type: none"> <li>● 与Oracle数据库解析字段差异：字符串encoding只支持UTF-8；version字段只支持1.0，1.0-1.9解析警告但正常执行，1.9以上报错。</li> <li>● 与Oracle数据库命名空间校验差异：解析未声明的命名空间标签正常执行，而Oracle数据库会报错。</li> <li>● 与Oracle数据库xml预定义实体解析差异：&amp;apos;&amp;quot;会被解析转译为字符' ”，而Oracle中预定义实体统一都没有转译为字符。</li> <li>● 与Oracle数据库DTD校验差异： <ul style="list-style-type: none"> <li>- !ATTLIST to type (CHECK check Check) "Ch..."将报错，因默认值"Ch..."不属于括号中枚举值，而Oracle数据库不报错。</li> <li>- &lt;!ENTITY baidu "www.baidu.com"&gt;.....&amp;Baidu;&amp;writer将报错，因区分字母大小写，Baidu无法与baidu对应，而Oracle数据库不报错。</li> </ul> </li> </ul>

序号	Oracle数据库	GaussDB数据库	差异
6	DBMS_XMLPARSER.PARSECLOB	支持，有差异，DBE_XMLPARSER.PARSECLOB	<ul style="list-style-type: none"> <li>• PARSECLOB不支持解析大于等于2GB的clob。</li> <li>• 与Oracle数据库解析字段差异：字符串encoding只支持UTF-8；version字段只支持1.0，1.0-1.9解析警告但正常执行，1.9以上报错。</li> <li>• 与Oracle数据库命名空间校验差异：解析未声明的命名空间标签正常执行，而Oracle数据库会报错。</li> <li>• 与Oracle数据库xml预定义实体解析差异：&amp;apos;&amp;quot;会被解析转译为字符' ”，而Oracle中预定义实体统一都没有转译为字符。</li> <li>• 与Oracle数据库DTD校验差异： <ul style="list-style-type: none"> <li>- !ATTLIST to type (CHECK check Check) "Ch..."将报错，因默认值"Ch..."不属于括号中枚举值，而Oracle数据库不报错。</li> <li>- &lt;!ENTITY baidu "www.baidu.com"&gt;.....&amp;Baidu;&amp;writer将报错，因区分字母大小写，Baidu无法与baidu对应，而Oracle数据库不报错。</li> </ul> </li> </ul>
7	DBMS_XMLPARSER.SETVALIDATIONMODE	DBE_XMLPARSER.SETVALIDATIONMODE	-

表 7-109 DBMS\_ILM 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	DBMS_ILM.ADD_TO_ILM	不支持	-
2	DBMS_ILM.ARCHIVESTATENAME	不支持	-

序号	Oracle数据库	GaussDB数据库	差异
3	DBMS_ILM.EXECUTE_ILM	DBE_ILM.EXECUTE_ILM	<ul style="list-style-type: none"> <li>GaussDB数据库的入参schema在Oracle数据库中对应为owner。</li> <li>GaussDB数据库不支持指定ilm_scope（一次指定多个对象）的操作。</li> </ul>
4	DBMS_ILM.EXECUTE_ILM_TASK	不支持	-
5	DBMS_ILM.PREVIEW_ILM	不支持	-
6	DBMS_ILM.REMOVE_FROM_ILM	不支持	-
7	DBMS_ILM.STOP_ILM	DBE_ILM.STOP_ILM	-

表 7-110 DBMS\_ILM\_ADMIN 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	DBMS_ILM_ADMIN.CLEAR_HEAT_MAP_ALL	不支持	-
2	DBMS_ILM_ADMIN.CLEAR_HEAT_MAP_TABLE	不支持	-
3	DBMS_ILM_ADMIN.CUSTOMIZE_ILM	DBE_ILM_ADMIN.CUSTOMIZE_ILM	<p>入参parameter取值对应的特性参数存在差异。</p> <ul style="list-style-type: none"> <li>GaussDB数据库param取值支持1、2、7、11、12、13、14和15。</li> <li>GaussDB数据库param取值为14时，对应的特性参数为WIND_DURATION，用于控制自动调度中执行窗口的持续时长，而ORACLE数据库无对应的特性参数。</li> </ul>
4	DBMS_ILM_ADMIN.DISABLE_ILM	DBE_ILM_ADMIN.DISABLE_ILM	-
5	DBMS_ILM_ADMIN.ENABLE_AUTO_OPTIMIZE	不支持	-



序号	Oracle数据库	GaussDB数据库	差异
6	DBMS_ILM_ADMIN.ENABLE_ILM	DBE_ILM_ADMIN.ENABLE_ILM	-
7	DBMS_ILM_ADMIN.IGNORE_AUTO_OPTIMIZE_CRITERIA	不支持	-
8	DBMS_ILM_ADMIN.SET_HEAT_MAP_ALL	不支持	-
9	DBMS_ILM_ADMIN.SET_HEAT_MAP_START	不支持	-
10	DBMS_ILM_ADMIN.SET_HEAT_MAP_TABLE	不支持	-

表 7-111 DBMS\_COMPRESSION 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	DBMS_COMPRESSION.GET_COMPRESSION_RATIO	DBE_COMPRESSION.GET_COMPRESSION_RATIO	<ul style="list-style-type: none"> <li>● GaussDB不支持LOBs的压缩率获取。</li> <li>● 对于单个对象的压缩率获取：                             <ul style="list-style-type: none"> <li>- GaussDB入参comptype取值仅支持1（未压缩）和2（高级压缩），Oracle还支持1024、2048等取值。</li> <li>- GaussDB入参objtype取值仅支持1（表对象），而Oracle还支持2（索引对象）。</li> <li>- Oracle数据库使用subset_numrows参数直接来决定采样的行数（即为参数的取值），而GaussDB则使用sample_ratio（采样率）来间接确定采样的行数。</li> </ul> </li> </ul>

序号	Oracle数据库	GaussDB数据库	差异
2	DBMS_COMPRESSION.GET_COMPRESSION_TYPE	DBE_COMPRESSION.GET_COMPRESSION_TYPE	<ul style="list-style-type: none"> <li>Oracle使用rowid来指定待获取压缩类型的行，而GaussDB则是使用行的ctid来指定。</li> <li>返回值为comptype，其取值差异同GET_COMPRESSION_RATIO。</li> <li>GaussDB该接口仅支持在DN调用，详见GaussDB《开发者指南》中“存储过程&gt;高级包&gt;二次封装接口（推荐）&gt;DBE_COMPRESSION”章节。</li> </ul>

表 7-112 DBMS\_HEAT\_MAP 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	DBMS_HEAT_MAP.BLOCK_HEAT_MAP	不支持	-
2	DBMS_HEAT_MAP.EXTENT_HEAT_MAP	不支持	-
3	DBMS_HEAT_MAP.OBJECT_HEAT_MAP	不支持	-
4	DBMS_HEAT_MAP.SEGMENT_HEAT_MAP	不支持	-
5	DBMS_HEAT_MAP.TABLESPACE_HEAT_MAP	不支持	-
6	不支持	DBE_HEAT_MAP.ROW_HEAT_MAP	GaussDB该接口仅支持在DN调用，详见GaussDB《开发者指南》中“存储过程>高级包>二次封装接口（推荐）>DBE_HEAT_MAP”章节。

表 7-113 DBMS\_DESCRIBE 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	DBMS_DESCRIBE.DESCRIBE_PROCEDURE	DBE_DESCRIBE.DESCRIBE_PROCEDURE	<ul style="list-style-type: none"> <li>datatype参数与Oracle存在差异，GaussDB返回数据类型的oid，Oracle数据库返回Oracle数据库内部的数据类型的编号。</li> <li>datalength、dataprecision和scale因GaussDB创建存储过程或函数时无法保留类型的约束（如number(7,2)、varchar2(20)等），该三个参数置0处理；Oracle可使用%type方法获得带约束的数据类型。</li> <li>GaussDB的具体使用请参见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_DESCRIBE”章节。</li> </ul>

表 7-114 DBMS\_STATS 兼容性说明

Oracle数据库	GaussDB数据库	差异
DBMS_STATS.ALTER_STATS_HISTORY_RETENTION	不支持	-
DBMS_STATS.CANCEL_ADVISOR_TASK	不支持	-
DBMS_STATS.CONFIGURE_ADVISOR_FILTER	不支持	-
DBMS_STATS.CONFIGURE_ADVISOR_OBJ_FILTER	不支持	-
DBMS_STATS.CONFIGURE_ADVISOR_OPR_FILTER	不支持	-
DBMS_STATS.CONFIGURE_ADVISOR_RULE_FILTER	不支持	-
DBMS_STATS.CREATE_ADVISOR_TASK	不支持	-
DBMS_STATS.CONVERT_RAW_VALUE	不支持	-
DBMS_STATS.CONVERT_RAW_VALUE_NVARCHAR	不支持	-

Oracle数据库	GaussDB数据库	差异
DBMS_STATS.CONVERT_RAW_VALUE_ROWID	不支持	-
DBMS_STATS.COPY_TABLE_STATS	不支持	-
DBMS_STATS.CREATE_EXTENDED_STATS	不支持	-
DBMS_STATS.CREATE_STAT_TABLE	DBE_STATS.CREATE_STAT_TABLE	<ul style="list-style-type: none"> <li>GaussDB中ownname应传schema名。</li> <li>GaussDB仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> </ul>
DBMS_STATS.DELETE_COLUMN_STATS	DBE_STATS.DELETE_COLUMN_STATS	<ul style="list-style-type: none"> <li>GaussDB中ownname应传schema名。</li> <li>GaussDB仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> <li>GaussDB中，使用该接口可以设置表达式统计信息，但tablename应传该表达式对应的索引名。</li> </ul>
DBMS_STATS.DELETE_DATABASE_PREFS	不支持	-
DBMS_STATS.DELETE_DATABASE_STATS	不支持	-
DEDBMS_STATS.DELETE_DICTIONARY_STATS	不支持	-
DBMS_STATS.DELETE_FIXED_OBJECTS_STATS	不支持	-

Oracle数据库	GaussDB数据库	差异
DBMS_STATS.DELETE_INDEX_STATS	DBE_STATS.DELETE_INDEX_STATS	<ul style="list-style-type: none"> <li>GaussDB中ownname应传schema名。</li> <li>GaussDB仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> </ul>
DBMS_STATS.DELETE_PENDING_STATS	不支持	-
DBMS_STATS.DELETE_PROCESSING_RATE	不支持	-
DBMS_STATS.DELETE_SCHEMA_PREFS	不支持	-
DBMS_STATS.DELETE_SCHEMA_STATS	DBE_STATS.DELETE_SCHEMA_STATS	<ul style="list-style-type: none"> <li>GaussDB中ownname应传schema名。</li> <li>GaussDB仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> </ul>
DBMS_STATS.DELETE_SYSTEM_STATS	不支持	-
DBMS_STATS.DELETE_TABLE_PREFS	不支持	-
DBMS_STATS.DELETE_TABLE_STATS	DBE_STATS.DELETE_TABLE_STATS	<ul style="list-style-type: none"> <li>GaussDB中ownname应传schema名。</li> <li>GaussDB仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> </ul>
DBMS_STATS.DIFF_TABLE_STATS_IN_HISTORY	不支持	-
DBMS_STATS.DIFF_TABLE_STATS_IN_PENDING	不支持	-
DBMS_STATS.DIFF_TABLE_STATS_IN_STATTAB	不支持	-

Oracle数据库	GaussDB数据库	差异
DBMS_STATS.DROP_ADV ISOR_TASK	不支持	-
DBMS_STATS.DROP_EXT ENDED_STATS	不支持	-
DBMS_STATS.DROP_STA T_TABLE	DBE_STATS.DROP_STAT_T ABLE	-
DBMS_STATS.EXECUTE_A DVISOR_TASK	不支持	-
DBMS_STATS.EXPORT_C OLUMN_STATS	DBE_STATS.EXPORT_COL UMN_STATS	<ul style="list-style-type: none"> <li>• GaussDB中ownname应传schema名。</li> <li>• gaussdb仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> <li>• 导出的列级统计信息与pg_statistic表保持一致，多列和pg_statistic_ext表保持一致。</li> <li>• 支持导出索引表达式统计信息。要求tablename传的是索引名称，colname传的是索引表达式名称。</li> <li>• 权限：需要具有查询表的analyze权限以及stattab表的siud权限。</li> </ul>
DBMS_STATS.EXPORT_D ATABASE_PREFS	不支持	-
DBMS_STATS.EXPORT_D ATABASE_STATS	不支持	-
DBMS_STATS.EXPORT_DI CTIONARY_STATS	不支持	-
DBMS_STATS.EXPORT_FI XED_OBJECTS_STATS	不支持	-

Oracle数据库	GaussDB数据库	差异
DBMS_STATS.EXPORT_INDEX_STATS	DBE_STATS.EXPORT_INDEX_STATS	<ul style="list-style-type: none"> <li>GaussDB中ownname应传schema名。</li> <li>gaussdb仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> <li>stattab表中，导出的表、分区级统计信息为numrows、numblocks、relallvisible，分别对应系统表pg_class、pg_partition的reltuples、relpages、relallvisible。</li> <li>权限：需要具有查询表的analyze权限以及stattab表的siud权限。</li> </ul>
DBMS_STATS.EXPORT_PENDING_STATS	不支持	-
DBMS_STATS.EXPORT_SCHEMA_PREFS	不支持	-

Oracle数据库	GaussDB数据库	差异
DBMS_STATS.EXPORT_SCHEMA_STATS	DBE_STATS.EXPORT_SCHEMA_STATS	<ul style="list-style-type: none"> <li>GaussDB中ownname应传schema名。</li> <li>gaussdb仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> <li>stattab表中，导出的表、分区级统计信息为numrows、numblocks、relallvisible，分别对应系统表pg_class、pg_partition的reltuples, relpages, relallvisible。导出表相关列级统计信息与pg_statistic表和pg_statistic_ext表保持一致。</li> <li>权限：需要具有stattab表的siud权限。</li> </ul>
DBMS_STATS.EXPORT_SYSTEM_STATS	不支持	-
DBMS_STATS.EXPORT_TABLE_PREFS	不支持	-



Oracle数据库	GaussDB数据库	差异
DBMS_STATS.EXPORT_TABLE_STATS	DBE_STATS.EXPORT_TABLE_STATS	<ul style="list-style-type: none"> <li>GaussDB中ownname应传schema名。</li> <li>GaussDB仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> <li>stattab表中，导出的表、分区级统计信息为numrows、numblocks、relallvisible，分别对应系统表pg_class、pg_partition的reltuples, relpages, relallvisible。级联导出的列级统计信息与pg_statistic表和pg_statistic_ext表保持一致。</li> <li>权限：需要具有查询表的analyze权限以及stattab表的siud权限。</li> </ul>
DBMS_STATS.FLUSH_DATABASE_MONITORING_INFO	不支持	-
DBMS_STATS.GATHER_DATABASE_STATS	不支持	-
DBMS_STATS.GATHER_DICTIONARY_STATS	不支持	-
DBMS_STATS.GATHER_FIXED_OBJECTS_STATS	不支持	-
DBMS_STATS.GATHER_INDEX_STATS	不支持	-
DBMS_STATS.GATHER_PROCESSING_RATE	不支持	-
DBMS_STATS.GATHER_SCHEMA_STATS	不支持	-
DBMS_STATS.GATHER_SYSTEM_STATS	不支持	-

Oracle数据库	GaussDB数据库	差异
DBMS_STATS.GATHER_TABLE_STATS	不支持	-
DBMS_STATS.GENERATE_STATS	不支持	-
DBMS_STATS.GET_ADVISOR_OPR_FILTER	不支持	-
DBMS_STATS.GET_ADVISOR_RECS	不支持	-
DBMS_STATS.GET_COLUMN_STATS	不支持	-
DBMS_STATS.GET_INDEX_STATS	不支持	-
DBMS_STATS.GET_PARAMETER	不支持	-
DBMS_STATS.GET_PREFS	不支持	-
DBMS_STATS.GET_STATS_HISTORY_AVAILABILITY	DBE_STATS.GET_STATS_HISTORY_AVAILABILITY	GaussDB查询到的是全库存在的最早历史统计信息的收集时间。
DBMS_STATS.GET_STATS_HISTORY_RETENTION	DBE_STATS.GET_STATS_HISTORY_RETENTION	-
DBMS_STATS.GET_SYSTEM_STATS	不支持	-
DBMS_STATS.GET_TABLE_STATS	不支持	-
DBMS_STATS.IMPLEMENT_ADVISOR_TASK	不支持	-

Oracle数据库	GaussDB数据库	差异
DBMS_STATS.IMPORT_COLUMN_STATS	DBE_STATS.IMPORT_COLUMN_STATS	<ul style="list-style-type: none"> <li>GaussDB中ownname应传schema名。</li> <li>GaussDB仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> <li>导出单列col导出的统计信息与pg_statistic表保持一致。多列ext-col导出的统计信息与pg_statistic_ext表保持一致。</li> <li>支持导入索引表达式统计信息。要求tablename传的是索引名称，colname传的是索引表达式名称。</li> <li>权限：需要具有查询表的ANALYZE权限以及stattab表的siud权限。</li> </ul>
DBMS_STATS.IMPORT_DATABASE_PREFS	不支持	-
DBMS_STATS.IMPORT_DATABASE_STATS	不支持	-
DBMS_STATS.IMPORT_DICTIONARY_STATS	不支持	-
DBMS_STATS.IMPORT_FIXED_OBJECTS_STATS	不支持	-

Oracle数据库	GaussDB数据库	差异
DBMS_STATS.IMPORT_IN DEX_STATS	DBE_STATS.IMPORT_IND EX_STATS	<ul style="list-style-type: none"> <li>• GaussDB中ownname应传schema名。</li> <li>• GaussDB仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> <li>• stattab表中，导入的表、分区级统计信息为numrows、numblocks、relallvisible，分别对应系统表pg_class、pg_partition的reltuples, relpages, relallvisible。</li> <li>• 权限：需要具有查询表的analyze权限以及stattab表的siud权限。</li> </ul>
DBMS_STATS.IMPORT_S HEMA_PREFS	不支持	-
DBMS_STATS.IMPORT_S HEMA_STATS	DBE_STATS.IMPORT_SCH EMA_STATS	<ul style="list-style-type: none"> <li>• GaussDB中ownname应传schema名。</li> <li>• GaussDB仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> <li>• stattab表中，导入的表、分区级统计信息为numrows、numblocks、relallvisible，分别对应系统表pg_class、pg_partition的reltuples, relpages, relallvisible。导入表相关列级统计信息与pg_statistic表和pg_statistic_ext表保持一致。</li> <li>• 权限：需要具有stattab表的siud权限。</li> </ul>

Oracle数据库	GaussDB数据库	差异
DBMS_STATS.IMPORT_SYSTEM_STATS	不支持	-
DBMS_STATS.IMPORT_TABLE_PREFS	不支持	-
DBMS_STATS.IMPORT_TABLE_STATS	DBE_STATS.IMPORT_TABLE_STATS	<ul style="list-style-type: none"> <li>GaussDB中ownname应传schema名。</li> <li>GaussDB仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> <li>stattab表中，导入的表、分区级统计信息为numrows、numblocks、relallvisible，分别对应系统表pg_class、pg_partition的reltuples, relpages, relallvisible。级联导入的列级统计信息与pg_statistic表和pg_statistic_ext表保持一致。</li> <li>权限：需要具有查询表的analyze权限以及stattab表的siud权限。</li> </ul>
DBMS_STATS.INTERRUPT_ADVISOR_TASK	不支持	-
DBMS_STATS.LOCK_PARTITION_STATS	DBE_STATS.LOCK_PARTITION_STATS	<ul style="list-style-type: none"> <li>GaussDB中ownname应传schema名。</li> </ul>
DBMS_STATS.LOCK_SCHEMA_STATS	DBE_STATS.LOCK_SCHEMA_STATS	<ul style="list-style-type: none"> <li>GaussDB中ownname应传schema名。</li> </ul>
DBMS_STATS.LOCK_TABLE_STATS	DBE_STATS.LOCK_TABLE_STATS	<ul style="list-style-type: none"> <li>GaussDB中ownname应传schema名。</li> </ul>
DBMS_STATS.MERGE_COLUMN_USAGE	不支持	-
DBMS_STATS.PREPARE_COLUMN_VALUES	不支持	-
DBMS_STATS.PREPARE_COLUMN_VALUES_ROWID	不支持	-

Oracle数据库	GaussDB数据库	差异
DBMS_STATS.PUBLISH_PENDING_STATS	不支持	-
DBMS_STATS.PURGE_STATS	DBE_STATS.PURGE_STATS	-
DBMS_STATS.REMAP_STAT_TABLE	不支持	-
DBMS_STATS.REPORT_ADVISOR_TASK	不支持	-
DBMS_STATS.REPORT_COLUMN_USAGE	不支持	-
DBMS_STATS.REPORT_GATHER_AUTO_STATS	不支持	-
DBMS_STATS.REPORT_GATHER_DATABASE_STATS	不支持	-
DBMS_STATS.REPORT_GATHER_DICTIONARY_STATS	不支持	-
DBMS_STATS.REPORT_GATHER_FIXED_OBJ_STATS	不支持	-
DBMS_STATS.REPORT_GATHER_SCHEMA_STATS	不支持	-
DBMS_STATS.REPORT_STATS_OPERATIONS	不支持	-
DBMS_STATS.RESET_ADVISOR_TASK	不支持	-
DBMS_STATS.RESET_COLUMN_USAGE	不支持	-
DBMS_STATS.RESET_GLOBAL_PREF_DEFAULTS	不支持	-
DBMS_STATS.RESET_PARAMETER_DEFAULTS	不支持	-
DBMS_STATS.RESTORE_DICTIONARY_STATS	不支持	-
DBMS_STATS.RESTORE_FIXED_OBJECTS_STATS	不支持	-

Oracle数据库	GaussDB数据库	差异
DBMS_STATS.RESTORE_SCHEMA_STATS	DBE_STATS.RESTORE_SCHEMA_STATS	<ul style="list-style-type: none"> <li>GaussDB中ownname应传schema名。</li> <li>GaussDB仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> </ul>
DBMS_STATS.RESTORE_SYSTEM_STATS	不支持	-
DBMS_STATS.RESTORE_TABLE_STATS	DBE_STATS.RESTORE_TABLE_STATS	<ul style="list-style-type: none"> <li>GaussDB中ownname应传schema名。</li> <li>GaussDB仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> </ul>
DBMS_STATS.RESUME_ADVISOR_TASK	不支持	-
DBMS_STATS.SCRIPT_ADVISOR_TASK	不支持	-
DBMS_STATS.SEED_COLUMN_USAGE	不支持	-
DBMS_STATS.SET_ADVISOR_TASK_PARAMETER	不支持	-
DBMS_STATS.SET_COLUMN_STATS	DBE_STATS.SET_COLUMN_STATS	<ul style="list-style-type: none"> <li>GaussDB中ownname应传schema名。</li> <li>Gaussdb仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> </ul>
DBMS_STATS.SET_DATABASE_PREFS	不支持	-
DBMS_STATS.SET_GLOBAL_PREFS	不支持	-

Oracle数据库	GaussDB数据库	差异
DBMS_STATS.SET_INDEX_STATS	DBE_STATS.SET_INDEX_STATS	<ul style="list-style-type: none"> <li>GaussDB中ownname应传schema名。GaussDB仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> <li>GaussDB中新增了relallvisible入参。</li> </ul>
DBMS_STATS.SET_PARAMETER	不支持	-
DBMS_STATS.SET_PROCESSING_RATE	不支持	-
DBMS_STATS.SET_SCHEMA_PREFS	不支持	-
DBMS_STATS.SET_SYSTEM_STATS	不支持	-
DBMS_STATS.SET_TABLE_PREFS	不支持	-
DBMS_STATS.SET_TABLE_STATS	DBE_STATS.SET_TABLE_STATS	<ul style="list-style-type: none"> <li>GaussDB中ownname应传schema名。</li> <li>GaussDB仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> <li>GaussDB中新增了relallvisible入参。</li> </ul>
DBMS_STATS.SHOW_EXTENDED_STATS_NAME	不支持	-
DBMS_STATS.TRANSFER_STATS	不支持	-
DBMS_STATS.UNLOCK_PARTITION_STATS	DBE_STATS.UNLOCK_PARTITION_STATS	GaussDB中ownname应传schema名。
DBMS_STATS.UNLOCK_SCHEMA_STATS	DBE_STATS.UNLOCK_SCHEMA_STATS	GaussDB中ownname应传schema名。
DBMS_STATS.UNLOCK_TABLE_STATS	DBE_STATS.UNLOCK_TABLE_STATS	GaussDB中ownname应传schema名。



Oracle数据库	GaussDB数据库	差异
DBMS_STATS.UPGRADE_STAT_TABLE	不支持	-

表 7-115 DBMS\_XMLGEN 兼容性说明

Oracle数据库	GaussDB数据库	差异
DBMS_XMLGEN.CONVERT	DBE_XMLGEN.CONVERT	-
DBMS_XMLGEN.NEWCONTEXT	DBE_XMLGEN.NEWCONTEXT	-
DBMS_XMLGEN.NEWCONTEXTFROMHIERARCHY	DBE_XMLGEN.NEWCONTEXTFROMHIERARCHY	<ul style="list-style-type: none"> <li>GaussDB生成的递归XML最大深度不能超过5000万层。</li> <li>Oracle的newcontextfromhierarchy方法对于connect by语句生成的xml是带xml头的，但是对于直接构造的数据不带xml头，GaussDB均带xml头。</li> </ul>
DBMS_XMLGEN.SETCONVERTSPECIALCHARS	DBE_XMLGEN.SETCONVERTSPECIALCHARS	-
DBMS_XMLGEN.SETNULLHANDLING	DBE_XMLGEN.SETNULLHANDLING	-
DBMS_XMLGEN.SETROWSETTAG	DBE_XMLGEN.SETROWSETTAG	-
DBMS_XMLGEN.SETROWTAG	DBE_XMLGEN.SETROWTAG	-
DBMS_XMLGEN.USENULLATTRIBUTEINDICATOR	DBE_XMLGEN.USENULLATTRIBUTEINDICATOR	-
DBMS_XMLGEN.USEITEMTAGSFORCOLL	DBE_XMLGEN.USEITEMTAGSFORCOLL	-
DBMS_XMLGEN.GETNUMROWSPROCESSED	DBE_XMLGEN.GETNUMROWSPROCESSED	-
DBMS_XMLGEN.SETMAXROWS	DBE_XMLGEN.SETMAXROWS	-
DBMS_XMLGEN.SETSKIPROWS	DBE_XMLGEN.SETSKIPROWS	-

Oracle数据库	GaussDB数据库	差异
DBMS_XMLGEN.RESTARTQUERY	DBE_XMLGEN.RESTARTQUERY	GaussDB分布式中不允许cursor反向移动，因此restartquery函数不可使用。
DBMS_XMLGEN.GETXMLTYPE	DBE_XMLGEN.GETXMLTYPE	-
DBMS_XMLGEN.GETXML	DBE_XMLGEN.GETXML	-
DBMS_XMLGEN.CLOSECONTEXT	DBE_XMLGEN.CLOSECONTEXT	-

## 7.2 主备版

### 7.2.1 Oracle 数据库兼容性概述

本章节主要介绍GaussDB数据库的Oracle兼容模式与Oracle数据库19C版本的兼容性对比信息。具体的相关信息和规格约束请参考《开发者指南》相应章节。

GaussDB数据库在数据类型、SQL功能和数据库对象等基本功能上与Oracle数据库兼容，PLSQL方面也基本兼容。由于GaussDB数据库与Oracle数据库底层框架实现存在差异，GaussDB数据库与Oracle数据库仍存在部分差异。GaussDB与Oracle的兼容性如下文所示。

### 7.2.2 SQL 的基本元素

GaussDB数据库基本兼容常用的SQL基本元素，但是部分存在差异，详细信息可参考下表。

### 7.2.2.1 数据类型

表 7-116 数值类型

序号	Oracle数据类型	GaussDB数据库是否支持	差异
1	NUMBER [ ( p [ , s ] ) ]	支持，有差异。	<p>精度和用法存在差异。</p> <ul style="list-style-type: none"> <li>NUMBER带参数时，GaussDB的precision与scale的最大边界值较Oracle更大。</li> <li>NUMBER不带参数时，GaussDB的precision的默认值远大于带参数时的最大边界值；而在Oracle中，precision的默认值等于带参数时的最大边界值。</li> <li>GaussDB不支持scale为负值；在Oracle中，scale为负值时会精确到相应的整数位。</li> </ul>
2	FLOAT [ ( p ) ]	支持。	-
3	BINARY_FLOAT	不支持。	-
4	BINARY_DOUBLE	支持。	-

表 7-117 日期时间类型

序号	Oracle数据类型	GaussDB数据库是否支持	差异
1	DATE	支持，有差异。	精度有差异，GaussDB支持的公元时间较Oracle范围更大。
2	TIMESTAMP [ ( fractional_seconds_precision ) ]	支持。	-

序号	Oracle数据类型	GaussDB数据库是否支持	差异
3	TIMESTAMP [ ( fractional_seconds_precision ) ] WITH TIME ZONE	支持，有差异。	GaussDB的 timestamptz等价于 Oracle的 timestampwithload timezone，缺少 Oracle对应的 timestamptz类型。  时区更新：部分国家 或地区因为政治、经 济、战争等因素经常 会更新时区信息， 数据库系统因此需要 同步修改时区文件以 确保时间内容的正确 性。  GaussDB时区类型目 前只涉及timestamp with timezone，当 新的时区文件生效 时，不会对已有的数 据进行变更，新数据 会随时区文件信息进 行同步调整。与 Oracle的同类型数据 能力存在差异。
4	TIMESTAMP [ ( fractional_seconds_precision ) ] WITH LOCAL TIME ZONE	不支持。	-
5	INTERVAL YEAR [ ( year_precision ) ] TO MONTH	支持。	-
6	INTERVAL DAY [ ( day_precision ) ] TO SECOND [ ( fractional_seconds_precision ) ]	支持。	-

表 7-118 字符类型

序号	Oracle数据类型	GaussDB数据库是否支持	差异
1	VARCHAR2 ( size [ BYTE   CHAR ] )	支持，有差异。	在GaussDB中，size单位为字节，即仅支持BYTE，不支持在BYTE和CHAR之间选择，最大10MB；而在Oracle中，size的单位可以在BYTE和CHAR之间选择，最大容量与使用的字符集有关。
2	NVARCHAR2 ( size )	支持，有差异。	在GaussDB中，NVARCHAR2(n)类型是VARCHAR2(n)类型的别名；而在Oracle中NVARCHAR2(n)与VARCHAR2(n)有一定的区别。
3	CHAR [ ( size [ BYTE   CHAR ] ) ]	支持，有差异。	在GaussDB中，size单位为字节，即仅支持BYTE，不支持在BYTE和CHAR之间选择，最大10MB；而在Oracle中，size的单位可以在BYTE和CHAR之间选择，最大容量为2000个字节，实际能容纳的字符数与使用的字符集有关。
4	NCHAR [ ( size ) ]	支持，有差异。	在GaussDB中，size单位为字节，最大10MB；而在Oracle中，size单位为字符，最大容量与使用的字符集有关。
5	CLOB	支持，有差异。	不支持定位器概念。
6	NCLOB	不支持。	-
7	LONG	不支持。	-

表 7-119 二进制类型

序号	Oracle数据类型	GaussDB数据库是否支持	差异
1	RAW ( size )	支持，有差异。	在GaussDB中，size是指字节长度建议值，不会用于校验输入raw类型的字节长度。
2	LONG RAW	不支持。	-
3	BLOB	支持。	-
4	BFILE	不支持。	-

表 7-120 ROWID 类型

序号	Oracle数据类型	GaussDB数据库是否支持
1	ROWID	不支持
2	UROWID	不支持

表 7-121 用户自定义类型

序号	Oracle数据类型	GaussDB数据库是否支持
1	对象类型	不支持
2	REF数据类型	不支持
3	可变数组	支持
4	嵌套表	支持

表 7-122 伪类型

序号	Oracle数据类型	GaussDB数据库是否支持
1	anytype	不支持
2	anydata	不支持
3	anydataset	不支持

表 7-123 xml 类型

序号	Oracle数据类型	GaussDB数据库是否支持	差异
1	XMLTYPE	支持，有差异	在GaussDB中，不支持部分操作，如不能通过使用XMLELEMENT函数将字符串转变为XMLTYPE类型，而是转变成XML类型。具体请参见《开发者指南》中“SQL参考 > 数据类型 > XMLTYPE类型”章节。
2	URIType	不支持	-

表 7-124 空间类型

序号	Oracle数据类型	GaussDB数据库是否支持
1	SDO_GEOMETRY	不支持
2	SDO_TOPO_GEOMETRY	不支持
3	SDO_GEORASTER	不支持

表 7-125 锁模式

级别	Oracle锁模式	GaussDB锁模式	备注锁模式
-	-	NULL	GaussDB的NULL表示未被赋予锁，与Oracle的0级锁对应。
0	none	INVALID	GaussDB的INVALID表示被赋予了非法锁。仅在运行过程中出现了GaussDB无法识别的锁时会被赋予INVALID锁。
1	null	AccessShare	-
2	RS	RowShare	-
3	RX	RowExclusive	-
4	S	ShareUpdateExclusive	-
5	SRX	Share	-

级别	Oracle锁模式	GaussDB锁模式	备注锁模式
6	\	ShareRowExclusive	-
7	X	Exclusive	-
8	\	AccessExclusive	-

### 7.2.2.2 数据类型比较规则

数据类型比较（排序）规则是指相同数据类型的值之间发生比较（排序）时遵循的比较（排序）规则。

表 7-126 比较规则

序号	Oracle数据库	GaussDB数据库	差异	规则描述
1	Numeric值	支持。	-	支持基于数值大小的比较，位于数轴中相对靠近正方向的数字大于位于数轴中相对靠近反方向的数字，如 $5 > 3$ 、 $1 > -1$ 、 $-1 > -2$ 。
2	日期时间值	支持。	-	支持基于日期早晚的比较，较晚的日期/时间戳大于较早的日期/时间戳，如 <code>'2000-01-01 12:00:00' &gt; '2000-01-01 11:59:59'</code> 。
3	二进制值	支持。	-	支持基于二进制值大小的比较，位于数轴中相对靠近正方向的数字大于位于数轴中相对靠近反方向的数字，如 <code>1001 0101 &gt; 1001 0011</code> 。



序号	Oracle数据库	GaussDB数据库	差异	规则描述
4	字符值	支持，有差异。	<ul style="list-style-type: none"> <li>● 有差异的内容：                             <ul style="list-style-type: none"> <li>- 在GaussDB和Oracle中，支持的比较规则不完全相同，相同比较规则的名称也可能不同。</li> <li>- GaussDB和Oracle的比较规则在可指定性上有差异，例如GaussDB不支持指定表级别的比较规则，而Oracle支持。</li> <li>- GaussDB和Oracle在指定比较规则的语法上有差异，例如在GaussDB中，使用ENCODING、LC_CTYPE和LC_COLLATE三个参数决定创建数据库时使用的字符集、字符分类和比较规则，具体请参见《开发者指南》中“SQL参考 &gt; SQL语法 &gt; C &gt; CREATE DATABASE”章节。而在Oracle中，各级别的比较规则通常由一系列带有NLS前缀的参数确定。</li> </ul> </li> <li>● 无差异的内容：                             <ul style="list-style-type: none"> <li>- 在GaussDB和Oracle中，对字符串的比较都遵循如下的基本规则：从字符串的第一个字符开始逐个比较，如果相等则继续比较下一个字符，直到出现不相等的字符或者有一个字符串已经结束为止。如果两个字符串的所有字符都相等，则认为它们相等；否则，以第一个不相等的字符的大</li> </ul> </li> </ul>	-

序号	Oracle数据库	GaussDB数据库	差异	规则描述
			<p>小关系作为比较结果进行判断。另外，部分数据类型会在比较前对较短的字符串添加空格，直到两个字符串等长后再排序，如CHAR类型。</p> <ul style="list-style-type: none"> <li>- 对于单字符的比较，常见的比较规则有按字符集中的二进制值排序、单语言排序等。在选用的字符集和比较规则都为默认的情况下GaussDB和Oracle都使用单语言排序。</li> <li>- GaussDB和Oracle中，字符类型的比较规则与所用字符集紧密相关，比如大部分非Unicode字符集不支持单语言排序，因而默认使用二进制值排序。</li> </ul>	
5	对象值	不支持。	-	-
6	Varrays和嵌套表	支持，有差异。	Oracle和GaussDB均支持Varrays的比较，与Oracle不同的是，GaussDB不仅支持比较两个Varrays中的元素个数，还支持同类型的Varrays之间的比较。	-
7	数据类型优先级	支持。	-	-
8	数据转换（显示/隐式类型转换）	支持。	-	-

### 7.2.2.3 字面量

表 7-127 字面量

序号	Oracle数据库	GaussDB数据库
1	文本字面量	支持
2	数值字面量	支持
3	日期时间字面量	支持
4	区间字面量	支持

### 7.2.2.4 格式模型

表 7-128 格式

序号	Oracle数据库	GaussDB数据库	差异
1	数字格式	支持，有差异。	GaussDB仅在参数a_format_version值为10c和a_format_dev_version值为s1的情况下，支持\$、C、TM、TM9、TME、U格式。同时在该参数下，不支持TH、PL、SG格式的fmt。 GaussDB具体支持情况请参见《开发者指南》中“SQL参考>函数和操作符>类型转换函数”章节的“number类型fmt参数表”。
2	日期时间格式	支持，有差异。	GaussDB中用于时间截断和时间四舍五入的参数仅在参数a_format_version值为10c和a_format_dev_version值为s1的情况下有效。 GaussDB具体支持情况请参见《开发者指南》中“SQL参考>函数和操作符>时间和日期处理函数和操作符”章节的“用于日期/时间格式化的模式”。
3	格式模型修饰符	支持。	-
4	字符串到日期转换规则	支持，有差异。	GaussDB中to_timestamp_tz函数在参数a_format_version值为10c和a_format_dev_version值为s1的情况下有效。 GaussDB具体支持情况请参见《开发者指南》中“SQL参考>函数和操作符>类型转换函数”章节的“to_date/to_timestamp/to_timestamp_tz”。

序号	Oracle数据库	GaussDB数据库	差异
5	xml格式模式	不支持。	-

### 7.2.2.5 空值

表 7-129 空值

序号	Oracle数据库	GaussDB数据库
1	IS NULL和IS NOT NULL	支持
2	NULLS in conditions	支持

### 7.2.2.6 注释

表 7-130 comment

序号	Oracle数据库	GaussDB数据库	差异
1	斜杠和星号 (/*)	支持	-
2	两个连字符 (--)	支持	-
3	COMMENT命令	支持	-
4	HINT	支持，有差异	GaussDB不支持'--+hint形式。具体信息请参见《开发者指南》中“SQL调优指南 > 使用Plan Hint进行调优”章节。

### 7.2.2.7 数据库对象

表 7-131 schema 对象

序号	Oracle数据库	GaussDB数据库	差异
1	分析视图	不支持	-
2	属性维度	不支持	-
3	集群	支持	-
4	约束	支持	-

序号	Oracle数据库	GaussDB数据库	差异
5	数据库链接	支持	-
6	数据库触发器	支持	-
7	尺寸	支持	-
8	外部过程库	不支持	-
9	分层结构	不支持	-
10	索引组织表	不支持	-
11	索引	支持	-
12	索引类型	不支持	-
13	java类	不支持	-
14	java资源	不支持	-
15	java源码	不支持	-
16	join groups	不支持	-
17	物化视图	支持	-
18	物化视图日志	不支持	-
19	挖掘模型	不支持	-
20	对象表	不支持	-
21	对象类型	不支持	-
22	对象视图	不支持	-
23	operators	支持	-
24	包	支持	-
25	序列	支持	-
26	存储函数	支持	-
27	存储过程	支持	-
28	同义词	支持，有差异	Oracle数据库的数据库对象在同一namespace内不能重名；GaussDB内同一namespace内同义词可与表、视图、函数、package重名，重名时，优先访问本名对象，未寻找到本名对象时才会寻找同义词指向的对象；Schema名是用户名时才会搜索PUBLIC同义词。
29	表	支持	-

序号	Oracle数据库	GaussDB数据库	差异
30	视图	支持	-
31	zone map	不支持	-

表 7-132 nonschema 对象

序号	Oracle数据库	GaussDB数据库
1	上下文	不支持
2	目录	支持
3	版本	不支持
4	闪回存档	不支持
5	锁定配置文件	不支持
6	配置文件	不支持
7	还原点	支持
8	角色	支持
9	回滚段	<ul style="list-style-type: none"><li>• ustore支持回滚段</li><li>• astore不支持回滚段</li></ul>
10	表空间	支持
11	表空间集	不支持
12	统一审计策略	支持
13	用户	支持

### 7.2.2.8 数据库对象名称和限定符

表 7-133 命名规则

序号	Oracle数据库	GaussDB数据库	差异
1	数据库对象命名规则	支持，有差异。	GaussDB默认小写。
2	模式对象命名规则	支持	-

### 7.2.2.9 SQL 语句中的引用架构对象和部件的语法

表 7-134 对象引用

序号	Oracle数据库	GaussDB数据库
1	引用对象的一般语法	支持
2	解析对象的引用	支持
3	引用外部模式对象	支持
4	引用外部数据库对象	支持
5	引用表和索引分区和子分区	支持

## 7.2.3 伪列

GaussDB数据库兼容分层查询伪列、序列伪列、rownum伪列，其余暂不支持。

### 分层查询伪列

表 7-135 分层查询伪列

序号	Oracle数据库	GaussDB数据库
1	connect_by_iscycle	支持
2	connect_by_isleaf	支持
3	level伪列	支持

### 序列伪列

表 7-136 序列

序号	Oracle数据库	GaussDB数据库	差异
1	currval	支持，有差异。	GaussDB以函数形式实现。兼容Oracle调用方式。
2	nextval	支持，有差异。	GaussDB以函数形式实现。兼容Oracle调用方式。

## rownum 伪列

表 7-137 rownum

序号	Oracle数据库	GaussDB数据库	差异
1	rownum	支持，有差异。	Oracle在left、right、full join的条件中使用rownum进行过滤时，不同的条件下表现不尽相同，可能存在忽略或部分忽略rownum条件的现象，而GaussDB在此情况下则表现为对left、right、full join后的结果进行过滤。

## 7.2.4 操作符

GaussDB数据库基本兼容Oracle数据库的运算符。

### SQL 运算符

表 7-138 SQL 运算符

序号	Oracle数据库	GaussDB数据库
1	一元运算符和二元运算符	支持
2	运算符优先级	支持

### 算术运算符

表 7-139 算术运算符

序号	Oracle数据库	GaussDB数据库
1	正负 (+-) 一元运算符	支持
2	加减 (+-) 二元运算符	支持
3	乘除 (*/) 二元运算符	支持

### COLLATE 运算符

表 7-140 COLLATE 运算符

序号	Oracle数据库	GaussDB数据库
1	COLLATE collation_name	支持



## 连接运算符

表 7-141 连接运算符

序号	Oracle数据库	GaussDB数据库
1		支持

## 分层查询运算符

表 7-142 分层查询运算符

序号	Oracle数据库	GaussDB数据库	差异
1	prior	支持，有差异。	GaussDB中仅支持对普通列调用，不支持对函数等调用。
2	connect_by_root	支持，有差异。	GaussDB中，connect_by_root调用时，用括号修饰操作值时，行为与Oracle一致；不使用括号时，仅支持对普通列调用此运算符。

## 集合运算符

表 7-143 集合运算符

序号	Oracle数据库	GaussDB数据库
1	union	支持
2	union all	支持
3	intersect	支持
4	minus	支持

## 多集合运算符

表 7-144 多集合运算符

序号	Oracle数据库	GaussDB数据库
1	multiset except	支持

序号	Oracle数据库	GaussDB数据库
2	multiset intersect	支持
3	multiset union	支持

## 用户自定义运算符

表 7-145 用户自定义运算符

序号	Oracle数据库	GaussDB数据库	差异
1	CREATE OPERATOR	支持，有差异。	<ul style="list-style-type: none"> <li>Oracle中提供的CONTEXT_CLAUSE支持自定义功能评估函数，和GaussDB约束选择性评估函数不同。GaussDB不支持自定义功能评估函数。</li> <li>Oracle和GaussDB可选参数差异较大。GaussDB具体请参考《开发者指南》中“SQL参考 &gt; SQL语法 &gt; C &gt; CREATE OPERATOR”章节的参数说明部分。</li> </ul>

## 比较操作符

序号	Oracle数据库	GaussDB数据库
1	< =	支持
2	< >	支持
3	> =	支持
4	^ =	支持
5	! =	不支持，!=中间存在空格时，!会被识别为阶乘。

当比较操作符 (<=、<>、>=、^=) 中间存在空格时，也可以识别成没有空格进行正常操作。!=中间存在空格时，!会被识别为阶乘，可能会导致结果与预期不一致。

## 7.2.5 表达式

GaussDB数据库兼容大部分Oracle数据库表达式。

表 7-146 表达式

序号	Oracle数据库	GaussDB数据库	差异
1	简单表达式	支持	-
2	分析视图表达式	不支持	-
3	复合表达式	支持	-
4	case表达式	支持	-
5	列表示式	支持	-
6	cursor表达式	不支持	-
7	日期时间表达式	支持，有差异	GaussDB的输出结果中不会带时区信息，而Oracle会带有类似” PM AMERICA/ LOS_ANGELES”的时区信息。
8	函数表达式	支持	-
9	区间表达式	部分支持。	GaussDB支持形如SELECT INTERVAL '999999999 23:59:59.999' day(9) to second FROM DUAL;的语句，而不支持形如SELECT(SYSDATE-SYSDATE) DAY TO SECOND FROM DUAL;的语句。Oracle均支持。

序号	Oracle数据库	GaussDB数据库	差异
10	json对象访问表达式	部分支持，有差异。	<p>1. GaussDB支持通过“-&gt;'key'“的方式从JSON对象中提取value，而Oracle支持通过“.key“方式来提取value。</p> <p>2. 对于JSONARRAY对象而言，Oracle支持通过“.key“方式一次性提取所有key对应的value，但GaussDB目前不支持。</p>
11	模型表达式	不支持	-
12	对象表达式	不支持	-
13	占位符表达式	部分支持。	对于形如“.var“的一般占位符表达式，GaussDB支持，但不支持通过INDICATOR关键字将两个一般占位符表达式结合起来。
14	标量子查询表达式	支持	-
15	类型构造器表达式	部分支持	GaussDB不支持在类型构造器前指定New关键字，而Oracle支持。
16	表达式list	支持	-

## 7.2.6 条件

GaussDB数据库兼容Oracle大部分条件。

表 7-147 条件

序号	Oracle数据库	GaussDB数据库	差异
1	比较条件	支持，有差异。	语句中存在 ANY、SOME、ALL 操作符时存在差异，Oracle 支持对list对象进行操作，而GaussDB中需要将list对象转换成数组表达式的形式后再进行操作。
2	浮点条件	不支持。	-
3	逻辑条件	支持。	-
4	模型条件	不支持。	-
5	多集合条件	不支持。	-
6	模式匹配条件	支持。	-
7	NULL值条件	支持。	-
8	XML条件	不支持。	-

序号	Oracle数据库	GaussDB数据库	差异
9	SQL/JSON条件	部分支持，有差异。	<ul style="list-style-type: none"><li>• GaussDB 不支持IS JSON条件和 JSON_TEXTCONTAINS条件。</li><li>• GaussDB 中 JSONB_EQ 条件同等于Oracle 中 JSON_EQUAL 条件，但 GaussDB 不支持 ERROR子句。</li><li>• GaussDB 中 JSONB_EXISTS条件同等于 Oracle中 JSON_EXISTS条件，但 GaussDB 不支持 ERROR子句、EMPTY子句和 PASSING子句。</li></ul>
10	复合条件	支持。	-
11	BETWEEN条件	支持。	-
12	EXISTS条件	支持。	-
13	IN条件	支持。	-
14	IS OF TYPE条件	不支持。	-

## 7.2.7 常见的 SQL DDL 子句

GaussDB数据库兼容部分DDL子句。

表 7-148 常用 SQL DDL 子句

序号	Oracle数据库	GaussDB数据库	差异
1	分配扩展子句	不支持。	-
2	约束	支持。	-
3	取消分配未使用子句	不支持。	-
4	文件规范	不支持。	-
5	日志记录子句	部分支持，有差异。	<ul style="list-style-type: none"><li>• GaussDB不支持 LOGGING约束子句和 FILESYSTEM_LIKE_LOGGING约束子句。</li><li>• GaussDB仅支持表级的 UNLOGGED约束，不支持列级的 UNLOGGED约束。</li><li>• GaussDB仅支持在 CREATE TABLE、CREATE TABLE AS、SELECT INTO语句中使用日志记录子句。</li></ul>
6	并行子句	不支持。	-
7	物理属性子句	部分支持，有差异。	<ul style="list-style-type: none"><li>• GaussDB不支持 PCTUSED。</li><li>• GaussDB 仅支持在 CREATE TABLE、CREATE INDEX语句中使用物理属性子句。</li></ul>
8	大小子句	不支持。	-

序号	Oracle数据库	GaussDB数据库	差异
9	存储子句	部分支持，有差异。	<ul style="list-style-type: none"> <li>Oracle中由STORAGE子句指定存储参数，而GaussDB中由WITH子句指定存储参数。</li> <li>GaussDB中可选的存储参数和Oracle存在很大差异。GaussDB具体可参见《开发者指南》中“SQL参考 &gt; SQL语法 &gt; C &gt; CREATE TABLE”的参数说明部分，“WITH ({storage_parameter = value} [, ...])”中描述了CREATE TABLE语句支持的存储参数。</li> </ul>
10	聚集函数嵌套	支持。	-

## 7.2.8 SQL 查询和子查询

GaussDB数据库兼容除分层查询以外的SQL查询和子查询。

表 7-149 SQL 查询和子查询

序号	Oracle数据库	GaussDB数据库	差异
1	创建简单查询	支持。	-
2	分层查询	支持，有差异。	GaussDB仅支持Oracle中的CONNECT_BY_FILTERING模式，不支持CONNECT_BY_NOFILTERING。
3	UNION [ALL], INTERSECT, 减运算符	支持。	-
4	查询结果排序	支持，有差异。	GaussDB查询不包含分组，且目标列同时包含聚集函数和集合返回函数时，不忽略对集合返回函数列的排序。



序号	Oracle数据库	GaussDB数据库	差异
5	Joins	支持，有差异。	GaussDB只支持和Oracle的Join Types，如 left/right、self、natural、full outer join等。不支持 In-Memory Join Group 等 Join Optimizations方法。
6	使用子查询	支持。	-
7	嵌套子查询的解嵌套	支持，有差异。	GaussDB不支持显式指定 HASH_AJ或MERGE_AJ。
8	分布式查询	支持，有差异。	GaussDB需要显式指定 DBLINK查询。
9	聚集函数嵌套	支持。	-

## 7.2.9 PL/SQL 语言

GaussDB数据库基本兼容的PL/SQL操作符、表达式，控制语句、集合和record等等，不支持预定义的PL/SQL常量和类型、子类型等。

### 7.2.9.1 PL/SQL 基本语法

表 7-150 PL/SQL 操作符

序号	Oracle数据库	GaussDB数据库
1	+	支持
2	:=	支持
3	=>	支持
4	%	支持
5	'	支持
6	.	支持
7		支持
8	/	支持
9	**	不支持
10	(	支持
11	)	支持
12	:	支持

序号	Oracle数据库	GaussDB数据库
13	,	支持
14	<<	支持
15	>>	支持
16	/*	支持
17	*/	支持
18	*	支持
19	"	支持
20	..	支持
21	=	支持
22	<>	支持
23	!=	支持
24	~=	支持
25	^=	支持
26	<	支持
27	>	支持
28	<=	支持
29	>=	支持
30	@	支持
31	--	支持
32	;	支持
33	-	支持

表 7-151 逻辑运算符

序号	Oracle数据库	GaussDB数据库
1	NOT	支持
2	AND	支持
3	OR	支持

表 7-152 比较表达式

序号	Oracle数据库	GaussDB数据库
1	IS [NOT] NULL	支持
2	LIKE	支持
3	BETWEEN	支持
4	IN	支持

表 7-153 条件表达式

序号	Oracle数据库	GaussDB数据库
1	simple CASE	支持
2	searched CASE	支持

表 7-154 变量声明相关参数

序号	Oracle数据库	GaussDB数据库	差异
1	%TYPE	支持，有差异。	<ul style="list-style-type: none"> <li>GaussDB不支持record变量%type。</li> <li>GaussDB不支持pkg.record变量%type、schema.pkg.record变量%type作为出入参类型。</li> <li>GaussDB不支持表/视图.column.column%type、schema.表/视图.column.column%type嵌套1层及以上，作为变量类型和或者出入参类型。</li> <li>GaussDB不支持record变量.column.column%type、pkg.record变量.column.column%type嵌套1层及以上的record的某列类型，作为变量类型和或者出入参类型。</li> </ul>

序号	Oracle数据库	GaussDB数据库	差异
2	%ROWTYPE	支持，有差异。	<ul style="list-style-type: none"> <li>GaussDB在多个CN的环境下，存储过程中无法声明临时表的%ROWTYPE及%TYPE属性。因为临时表仅在当前session有效，在编译阶段其他CN无法看到当前CN的临时表。故多个CN的环境下，会提示该临时表不存在。</li> <li>GaussDB不支持view %rowtype、schema.view %rowtype作为出入参类型。</li> <li>GaussDB不支持package.cursor变量%rowtype作为出入参类型。</li> </ul>

### 7.2.9.2 数据类型兼容性

表 7-155 其他 PL/SQL 数据类型

序号	Oracle数据库	GaussDB数据库	差异
1	CHARACTER	支持，有差异。	<ul style="list-style-type: none"> <li>GaussDB中字节长度限制为：1~10485760。</li> <li>Oracle中字节长度限制为：1~32767。</li> </ul>
2	VARCHAR	支持，有差异。	<ul style="list-style-type: none"> <li>GaussDB中字节长度限制为：1~10485760。</li> <li>Oracle中字节长度限制为：1~32767。</li> </ul>
3	STRING	不支持。	-
4	PLS_INTEGER	不支持。	GaussDB可使用int类型替代。
5	BINARY_INTEGER	支持。	-

表 7-156 用户自定义 PL/SQL 子类型

序号	Oracle数据库	GaussDB数据库	差异
1	SUBTYPE subtype_name IS base_type	支持。	-
2	SUBTYPE subtype_name IS base_type { precision [, scale ]   RANGE low_value .. high_value } [ NOT NULL ]	支持，有差异。	<ul style="list-style-type: none"> <li>• GaussDB 仅INT类型支持range约束。Oracle仅PLS_INTEGER、BINARY_INTEGER及其预定义子类型支持RANGE约束。</li> <li>• GaussDB 中在定义变量时，不支持指定range约束。</li> <li>• GaussDB 中，subtype基类型为字符数据类型时，不支持指定字符集信息。</li> </ul>
3	SUBTYPE subtype_name IS base_type [ NOT NULL ]	支持。	-

### 7.2.9.3 控制语句

表 7-157 条件语句

序号	Oracle数据库	GaussDB数据库
1	IF THEN	支持
2	IF THEN ELSE	支持
3	IF THEN ELSIF	支持

序号	Oracle数据库	GaussDB数据库
4	<b>simple CASE:</b> CASE selector WHEN selector_value_1 THEN statements_1 WHEN selector_value_2 THEN statements_2 ... WHEN selector_value_n THEN statements_n [ ELSE else_statements END CASE;]	支持
5	<b>searched CASE:</b> CASE WHEN condition_1 THEN statements_1 WHEN condition_2 THEN statements_2 ... WHEN condition_n THEN statements_n [ ELSE else_statements END CASE;]	支持

表 7-158 LOOP 循环语句

序号	Oracle数据库	GaussDB数据库
1	[ label ] LOOP statements END LOOP [ label ];	支持
2	EXIT;	支持
3	EXIT WHEN;	支持
4	CONTINUE;	支持
5	CONTINUE WHEN;	支持

表 7-159 FOR 循环语句

序号	Oracle数据库	GaussDB数据库	差异
1	[ label ] FOR index IN [ REVERSE ] lower_bound..upper_bound LOOP statements END LOOP [ label ];	支持，有差异。	GaussDB使用 REVERSE关键 字时， lower_bound 必须大于等于 upper_bound ，否则循环体 不会被执行。
2	EXIT WHEN;	支持。	-
3	CONTINUE WHEN;	支持。	-

表 7-160 WHILE LOOP 循环语句

序号	Oracle数据库	GaussDB数据库
1	[ label ] WHILE condition LOOP statements END LOOP [ label ];	支持

表 7-161 GOTO 语句

序号	Oracle数据库	GaussDB数据库
1	GOTO	支持

表 7-162 NULL 语句

序号	Oracle数据库	GaussDB数据库
1	NULL	支持

## 7.2.9.4 集合和 Record

表 7-163 类型

序号	Oracle数据库	GaussDB数据库
1	Associative array (or index-by table)	支持

序号	Oracle数据库	GaussDB数据库
2	VARRAY (variable-size array)	支持
3	Nested table	支持
4	record	支持

表 7-164 语法

类型	Oracle数据库	GaussDB数据库	差异
Associative array (or index-by table)	TABLE OF datatype [ NOT NULL ] INDEX BY { PLS_INTEGER   BINARY_INTEGER   VARCHAR2 ( v_size )   data_type }	支持，有差异。	<ul style="list-style-type: none"><li>• GaussDB不支持 PLS_INTEGER类型，GaussDB内 data_type可以为基础数据类型、或存储过程内定义的 record类型、集合类型、数组类型，不支持ref cursor类型。</li><li>• GaussDB内NOT NULL只支持语法不支持功能，即不会校验元素是否为NULL。</li><li>• 详情可参考《开发者指南》中“存储过程 &gt; 数组、集合和record &gt; 集合”章节。</li></ul>



类型	Oracle数据库	GaussDB数据库	差异
VARRAY (variable-size array)	{ VARRAY   [ VARYING ] ARRAY } ( size_limit ) OF datatype [ NOT NULL ]	支持，有差异。	<ul style="list-style-type: none"> <li>• GaussDB内不支持 NOT NULL语法。</li> <li>• GaussDB内不支持 datatype为varray 类型（ varray不能嵌套 ）。</li> <li>• size_limit功能生效需要在 behavior_compat_optionsGUC参数中开启 varray_compat参数。</li> <li>• 详情可参考《开发者指南》中“存储过程 &gt; 数组、集合和record &gt; 数组”章节。</li> </ul>
Nested table	TABLE OF datatype [ NOT NULL ]	支持，有差异。	<ul style="list-style-type: none"> <li>• GaussDB内NOT NULL只支持语法不支持功能，即不会校验元素是否为 NULL。</li> <li>• 详情可参考《开发者指南》中“存储过程 &gt; 数组、集合和record &gt; 集合”章节。</li> </ul>

类型	Oracle数据库	GaussDB数据库	差异
record	TYPE record_type IS RECORD ( field_definition [, field_definition]... );	支持。	<ul style="list-style-type: none"> <li>record的列可以定义为NOT NULL属性也可以指定默认值。其他类型嵌套record类型，record类型的默认值和NOT NULL不生效；通过 package.record_type访问类型的形式来创建record变量，该record变量的默认值和NOT NULL不生效。</li> <li>详情可参考《开发者指南》中“存储过程 &gt; 数组、集合和record &gt; record”章节。</li> </ul>

表 7-165 构造器

序号	Oracle数据库	GaussDB数据库
1	collection_type ( [ value [, value ]... ] )	支持

表 7-166 变量赋值

序号	Oracle数据库	GaussDB数据库	差异
1	Associative array (or index-by table)	支持。	-
2	VARRAY (variable-size array)	支持，有差异。	<ul style="list-style-type: none"> <li>GaussDB不同VARRAY类型的数据可以相互赋值，取决于其元素之间是否能相互隐式转换。</li> <li>详情可参考《开发者指南》中“存储过程 &gt; 数组、集合和record &gt; 数组”章节。</li> </ul>

序号	Oracle数据库	GaussDB数据库	差异
3	Nested table	支持。	-
4	record	支持，有差异。	<ul style="list-style-type: none"> <li>GaussDB不同record类型的数据可以相互赋值，取决于列与列之间是否能相互隐式转换。</li> <li>详情可参考《开发者指南》中“存储过程 &gt; 数组、集合和record &gt; record”章节。</li> </ul>

表 7-167 集合操作符

序号	Oracle数据库	GaussDB数据库	差异
1	=	支持，有差异。	Oracle: 比较时忽略集合成员先后顺序。 GaussDB: 比较时严格按照集合成员先后顺序。
2	<>	支持，有差异。	Oracle: 比较时忽略集合成员先后顺序。 GaussDB: 比较时严格按照集合成员先后顺序。
3	IS[NOT] NULL	支持。	-
4	^=	支持，有差异。	Oracle: 比较时忽略集合成员先后顺序。 GaussDB: 比较时严格按照集合成员先后顺序。
5	~=	不支持。	-
6	IS[NOT] A SET	不支持。	-
7	IS [NOT] EMPTY	不支持。	-
8	expr [ NOT ] MEMBER [ OF ] nested_table	不支持。	-

序号	Oracle数据库	GaussDB数据库	差异
9	nested_table1 [ NOT ] SUBMULTISET [ OF ] nested_table2	不支持。	-
10	[NOT] IN	支持，有差异。	Oracle: 比较时忽略集合成员先后顺序。 GaussDB: 比较时严格按照集合成员先后顺序。

表 7-168 集合 MULTISET 函数

序号	Oracle数据库	GaussDB数据库
1	MULTISET UNION [ALL   DISTINCT]	支持
2	MULTISET EXCEPT [ALL   DISTINCT]	支持
3	MULTISET INTERSECT [ALL   DISTINCT]	支持

表 7-169 集合类型函数

序号	Oracle数据库	GaussDB数据库	差异
1	exists(idx)	支持。	-
2	extend[(count[, idx])]	支持，有差异。	GaussDB仅支持 nesttable类型。
3	delete[(idx1[, idx2])]	支持。	-
4	trim[(n)]	支持，有差异。	GaussDB仅支持 nesttable类型。
5	count	支持。	-
6	first	支持。	-
7	last	支持。	-
8	prior(idx)	支持。	-
9	next(idx)	支持。	-

序号	Oracle数据库	GaussDB数据库	差异
10	limit	支持，有差异。	GaussDB用于nesttable类型，返回集合中最大可以储存的元素个数，只适用于array类型，nesttable返回空。

表 7-170 record 变量操作

序号	Oracle数据库	GaussDB数据库
1	构造器	支持
2	%ROWTYPE声明变量	支持
3	定义常量constant	不支持

表 7-171 集合相关函数

序号	Oracle数据库	GaussDB数据库
1	unnest_table(anynesttable)	支持
2	unnest_table(anyindexbytable)	支持

## 7.2.9.5 静态 SQL

表 7-172 静态查询 SQL 语句

序号	Oracle数据库	GaussDB数据库	差异
1	SELECT	支持，有差异。	<p>GaussDB和Oracle在某些场景下有不同。</p> <p>GaussDB中不同事务中的共享锁在如下场景中不会互相阻塞：</p> <p>SELECT FOR SHARE - SELECT FOR SHARE;</p> <p>SELECT FOR SHARE - SELECT FOR KEY SHARE;</p> <p>SELECT FOR KEY SHARE - SELECT FOR KEY SHARE;</p> <p>SELECT FOR KEY SHARE - SELECT FOR NO KEY UPDATE;</p> <p>上述场景中，由于锁与锁之间未阻塞，对在其他事务中存在非阻塞锁的数据指定SKIP LOCKED时，锁不会被跳过。</p>

表 7-173 静态 DML SQL 语句

序号	Oracle数据库	GaussDB数据库
1	INSERT	支持
2	UPDATE	支持
3	DELETE	支持
4	MERGE	支持
5	LOCK TABLE	支持

表 7-174 静态 TCL SQL 语句

序号	Oracle数据库	GaussDB数据库	差异
1	COMMIT	支持。	-
2	ROLLBACK	支持。	-
3	SAVEPOINT	支持。	-
4	SET TRANSACTION	支持，有差异。	GaussDB不支持 NAME string语法、USE ROLLBACK SEGMENT rollback_segment 语法。

表 7-175 伪列

序号	Oracle数据库	GaussDB数据库	差异
1	CURRVAL and NEXTVAL	支持。	-
2	LEVEL	支持。	-
3	OBJECT_VALUE	不支持。	-
4	ROWID	不支持。	-
5	ROWNUM	支持，有差异。	不推荐ROWNUM条件用于JOIN ON子句。GaussDB中ROWNUM条件用于JOIN ON子句时在LEFT JOIN、RIGHT JOIN、FULL JOIN场景下和MERGE INTO场景下与其他数据库行为不一致，直接进行业务迁移存在风险。

表 7-176 隐式游标属性

序号	Oracle数据库	GaussDB数据库	差异
1	SQL%FOUND	支持，有差异。	GaussDB在commit \rollback之后不刷新隐式游标结果，Oracle会在commit \rollback之后刷新隐式游标结果。
2	SQL %NOTFOUND	支持，有差异。	
3	SQL %ROWCOUNT	支持，有差异。	
4	SQL%ISOPEN	支持，有差异。	
5	SQL %BULK_ROWCOUNT	不支持。	
6	SQL %BULK_EXCEPTIONS	支持，有差异。	

表 7-177 显式游标语法及关键字

序号	Oracle数据库	GaussDB数据库
1	CURSOR cursor_name [ parameter_list ] RETURN return_type;	支持
2	CURSOR cursor_name [ parameter_list ] [ RETURN return_type ] IS select_statement;	支持
3	OPEN	支持
4	CLOSE	支持
5	FETCH	支持
6	CURRENT OF CURSOR	支持

表 7-178 显式游标属性

序号	Oracle数据库	GaussDB数据库
1	SQL%FOUND	支持
2	SQL%NOTFOUND	支持
3	SQL%ROWCOUNT	支持
4	SQL%ISOPEN	支持



表 7-179 游标循环

序号	Oracle数据库	GaussDB数据库
1	FOR LOOP	支持

表 7-180 自治事务支持场景

序号	Oracle数据库	GaussDB数据库
1	存储过程	支持
2	匿名块	支持
3	函数	支持
4	Package	支持

## 7.2.9.6 动态 SQL

表 7-181 动态 SQL 语句执行方式

序号	Oracle数据库	GaussDB数据库	差异
1	EXECUTE IMMEDIATE	支持，有差异。	<ul style="list-style-type: none"> <li>• GaussDB通过 dynamic_sql_compat 参数控制同名变量是否读取同一参数，并且检查调用存储过程时绑定参数出入参类型是否与语句参数类型一致。</li> <li>• GaussDB不支持调用匿名块中部分绑定参数场景，例如匿名块中嵌套动态语句，使用表达式绑定参数，具体请参见《开发者指南》中“存储过程 &gt; 动态语句 &gt; 动态调用匿名块”章节。</li> <li>• GaussDB不支持 RETURNING/ RETURN INTO。</li> </ul>
2	OPEN FOR、FETCH 、CLOSE	支持。	-

## 7.2.9.7 Trigger

表 7-182 trigger 类型

序号	Oracle数据库	GaussDB数据库	差异
1	DML TRIGGER	支持，有差异。	GaussDB不支持 Compound DML Triggers。
2	SYSTEM TRIGGER	不支持	-

表 7-183 create trigger

语法类型	Oracle数据库	GaussDB数据库	差异
create语法	CREATE [ OR REPLACE ] [ EDITIONABLE   NONEDITIONABLE ] TRIGGER plsql_trigger_source	支持，有差异。	GaussDB不支持 EDITIONABLE   NONEDITIONABLE，支持 plsql_trigger_source 部分行为。
plsql_trigger_source :: = =	[schema.] trigger_name [ sharing_clause ] [ default_collation_clause ] { simple_dml_trigger   instead_of_dml_trigger   compound_dml_trigger   system_trigger }	支持，有差异。	GaussDB不支持 schema、 sharing_clause、 default_collation_clause。
simple_dml_trigger ::=	{ BEFORE   AFTER } dml_event_clause [ referencing_clause ] [ FOR EACH ROW ] [ trigger_edition_clause ] [ trigger_ordering_clause ] [ ENABLE   DISABLE ] [ WHEN ( condition ) ] trigger_body	支持，有差异。	GaussDB不支持 referencing_clause 、 referencing_clause ( 用from referencing_table 代替 )、 trigger_edition_clause、 trigger_ordering_clause、 ENABLE   DISABLE；支持 trigger_body部分 行为。  GaussDB在没有 INSTEAD OF TRIGGER的视图上 创建语句级 BEFORE/AFTER TRIGGER时不会报 错，执行DML时报 错。

语法类型	Oracle数据库	GaussDB数据库	差异
dml_event_clause ::=	{ DELETE   INSERT   UPDATE [ OF column [, column ]... ] } [ OR { DELETE   INSERT   UPDATE [ OF column [, column]... ] } ]... ON [ schema. ] { table   view }	不支持。	-
trigger_body ::=	{ plsql_block   CALL routine_clause }	支持，有差异。	GaussDB的 plsql_block 不允许声明为PRAGMA AUTONOMOUS_TRANSACTION。对于第二个分支，支持类似语法，具体为EXECUTE PROCEDURE function_name ( arguments );方式执行function，并且function需要用户定义，必须声明为不带参数并返回类型为触发器，在触发器触发时执行。
instead_of_dml_trigger ::=	INSTEAD OF { DELETE   INSERT   UPDATE } [ OR { DELETE   INSERT   UPDATE } ]... ON [ NESTED TABLE nested_table_column OF ] [ schema. ] nonconditioning_view [ referencing_clause ] [ FOR EACH ROW ] [ trigger_edition_clause ] [ trigger_ordering_clause ] [ ENABLE   DISABLE ] trigger_body	支持，有差异。	GaussDB不支持 NESTED TABLE nested_table_column OF、referencing_clause、trigger_edition_clause、trigger_ordering_clause、ENABLE   DISABLE。

语法类型	Oracle数据库	GaussDB数据库	差异
compound_dml_trigger ::=	CREATE trigger FOR dml_event_clause ON view  COMPOUND TRIGGER  INSTEAD OF EACH ROW IS BEGIN statement;  END INSTEAD OF EACH ROW;	不支持。	-
system_trigger ::=	{ BEFORE   AFTER   INSTEAD OF } { ddl_event [OR ddl_event]...   database_event [OR database_event ]... } ON { [schema.] SCHEMA   [ PLUGGABLE ] DATABASE } [ trigger_ordering_clause ] [ ENABLE   DISABLE ] trigger_body	不支持。	-

表 7-184 alter trigger

序号	Oracle数据库	GaussDB数据库	差异
1	ALTER TRIGGER [ schema. ] trigger_name  { trigger_compile_clause    { ENABLE   DISABLE }   RENAME TO new_name    { EDITIONABLE   NONEDITIONABLE } };	支持，有差异。	GaussDB不支持 schema、 trigger_compile_clause、{ ENABLE   DISABLE }、 { EDITIONABLE   NONEDITIONABLE }。

表 7-185 drop trigger

序号	Oracle数据库	GaussDB数据库	差异
1	DROP TRIGGER [ schema. ] trigger ;	支持，有差异。	GaussDB不支持 schema，需要在 trigger_name后面 加上ON table_name。

Oracle数据库名为\*\_TRIGGERS的视图统计了trigger的相关信息，GaussDB相关视图与Oracle存在差异，GaussDB视图具体请参见《开发者指南》中“系统表和系统视图 > 系统视图 > 其他视图”中的DB\_TRIGGERS、ADM\_TRIGGERS、MY\_TRIGGERSDB\_TRIGGERS、ADM\_TRIGGERS、MY\_TRIGGERS章节。

表 7-186 Nested, Package, and Standalone Subprograms 兼容性

序号	Oracle数据库	GaussDB数据库	差异
1	nested subprogram (子块)	支持，有差异。	不支持重载，不 支持定义为自治事 务，不支持SETOF 的使用，仅限一个 限定符引用嵌套子 程序或嵌套子程序 的变量。
2	package subprogram	支持。	-
3	standalone subprogram (包含 Function & Procedure)	支持。	-
4	匿名块	支持。	-

表 7-187 RETURN 语句支持情况

序号	Oracle数据库	GaussDB数据库
1	Function	支持
2	Procedure	支持
3	匿名块	支持

表 7-188 Function 相关参数

序号	Oracle数据库	GaussDB数据库	差异
1	DETERMINISTIC	支持，有差异。	GaussDB中为IMMUTABLE。
2	PARALLEL_ENABLE	不支持。	-
3	PIPELINED	不支持。	-
4	RESULT_CACHE	不支持。	-

表 7-189 参数形式支持

序号	Oracle数据库	GaussDB数据库
1	IN	支持
2	OUT	支持
3	IN OUT	支持

表 7-190 CREATE 语句

序号	Oracle数据库	GaussDB数据库	差异
1	CREATE FUNCTION	支持，有差异。	GaussDB不支持IF NOT EXISTS语法、不支持 sharing_clause、仅支持部分指定function属性的子句（属性的子句仅支持 invoker_rights_clause 子句）、不支持关键字 [ EDITIONABLE   NONEDITIONABLE ]。 具体语法请参见《开发者指南》中” SQL参考 > SQL语法 > C >CREATE FUNCTION ” 章节。
2	CREATE LIBRARY	不支持。	-

序号	Oracle数据库	GaussDB数据库	差异
3	CREATE PACKAGE	支持，有差异。	GaussDB不支持IF NOT EXISTS语法、不支持 sharing_clause、仅支持部分指定package属性的子句（属性的子句仅支持 invoker_rights_clause 子句）、不支持关键字 [ EDITIONABLE   NONEDITIONABLE ] 。 具体语法请参见《开发者指南》中” SQL参考 > SQL语法 > C >CREATE PACKAGE ” 章节。
4	CREATE PACKAGE BODY	支持，有差异。	GaussDB不支持IF NOT EXISTS语法、不支持 sharing_clause、不支持关键字 [ EDITIONABLE   NONEDITIONABLE ] 。 具体语法请参见《开发者指南》中” SQL参考 > SQL语法 > C >CREATE PACKAGE ” 章节。
5	CREATE PROCEDURE	支持，有差异。	GaussDB不支持IF NOT EXISTS语法、不支持sharing_clause 以及后面的子句、不支持关键字 [ EDITIONABLE   NONEDITIONABLE ] 。 具体语法请参见《开发者指南》中” SQL参考 > SQL语法 > C >CREATE PROCEDURE ” 章节。



序号	Oracle数据库	GaussDB数据库	差异
6	CREATE TRIGGER	支持，有差异。	GaussDB的具体语法请参见《开发者指南》中”SQL参考 > SQL语法 > C >CREATE TRIGGER ” 章节。
7	CREATE TYPE	支持，有差异。	GaussDB不支持 varray、object类型、UNDER语法。 具体语法请参见《开发者指南》中”SQL参考 > SQL语法 > C >CREATE TYPE ” 章节。
8	CREATE TYPE BODY	不支持。	-

表 7-191 ALTER 语句

序号	Oracle数据库	GaussDB数据库	差异
1	ALTER FUNCTION	支持，有差异。	GaussDB不支持关键字 [ EDITIONABLE   NONEDITIONABLE ]、REUSE、SETTINGS、DEBUG。 具体语法请参见《开发者指南》中”SQL参考 > SQL语法 > A > ALTER FUNCTION ” 章节。
2	ALTER LIBRARY	不支持。	-

序号	Oracle数据库	GaussDB数据库	差异
3	ALTER PACKAGE	支持，有差异。	GaussDB不支持关键字 [ EDITIONABLE   NONEDITIONABLE ]、REUSE、SETTINGS、DEBUG。 具体语法请参见《开发者指南》中”SQL参考 > SQL语法 > A > ALTER PACKAGE ” 章节。
4	ALTER PROCEDURE	支持，有差异。	GaussDB不支持关键字 [ EDITIONABLE   NONEDITIONABLE ]、REUSE、SETTINGS、DEBUG。 具体语法请参见《开发者指南》中”SQL参考 > SQL语法 > A > ALTER PROCEDURE ” 章节。
5	ALTER TRIGGER	支持，有差异。	GaussDB仅支持修改trigger名字。 具体语法请参见《开发者指南》中”SQL参考 > SQL语法 > A > ALTER TRIGGER ” 章节。
6	ALTER TYPE	支持，有差异。	GaussDB仅支持部分语句。 具体语法请参见《开发者指南》中”SQL参考 > SQL语法 > A > ALTER TYPE ” 章节。

表 7-192 DROP 语句

序号	Oracle数据库	GaussDB数据库	差异
1	DROP FUNCTION	支持。	-
2	DROP LIBRARY	不支持。	-
3	DROP PACKAGE	支持。	-
4	DROP PROCEDURE	支持。	-
5	DROP TRIGGER	支持，有差异。	GaussDB的语法不同。 具体语法可参考： 请参见《开发者指南》中” SQL参考 > SQL语法 > D > DROP TRIGGER ” 章节。
6	DROP TYPE	支持，有差异。	GaussDB不支持关键字FORCE、VALIDATE。 具体语法请参见《开发者指南》中” SQL参考 > SQL语法 > D > DROP TYPE ” 章节。
7	DROP TYPE BODY	不支持。	-

表 7-193 Function、Procedure、匿名块相关关键字

序号	Oracle数据库	GaussDB数据库	差异
1	ACCESSIBLE BY	不支持。	-

序号	Oracle数据库	GaussDB数据库	差异
2	AGGREGATE	支持，有差异。	<ul style="list-style-type: none"> <li>GaussDB不支持Oracle的aggregate using [schema.] implementation_type用法。</li> <li>GaussDB的具体用法请参见《开发者指南》中“SQL参考 &gt; SQL语法 &gt; C &gt; CREATE AGGREGATE”章节。</li> </ul> 语法不同，但实现功能相同。
3	DETERMINISTIC	支持，有差异。	GaussDB仅在语法上支持关键字 DETERMINISTIC，未实现功能。
4	PIPE ROW	不支持。	-
5	PIPELINED	不支持。	-
6	SQL_MACRO	不支持。	-
7	RESTRICT_REFERENCES	不支持。	-
8	INLINE	不支持。	-

表 7-194 异常处理相关关键字

序号	Oracle数据库	GaussDB数据库	差异
1	EXCEPTION_INIT	支持，有差异。	GaussDB不支持与系统错误码进行绑定
2	Exception	支持。	-
3	Exception Handler	支持。	-
4	SQLCODE	支持。	-
5	SQLERRM	支持。	-

表 7-195 其他 PL/SQL 关键字

序号	Oracle数据库	GaussDB数据库	差异
1	COVERAGE	不支持。	-
2	COLLATION	支持。	-
3	DEPRECATE	不支持。	-
4	FORALL	支持。	-
5	NOCOPY	不支持。	-
6	RETURNING	支持。	-
7	SERIALLY_REUSABLE	不支持。	-
8	SHARING	不支持。	-
9	BULK COLLECT	支持。	-

## 7.2.10 系统函数

兼容函数分为：单行函数、用户自定义函数、AGG函数、分析函数、对象引用函数、模型函数、OLAP函数。

### 7.2.10.1 单行函数

序号	Oracle数据库	GaussDB数据库
1	<a href="#">数值函数</a>	支持，有差异。
2	<a href="#">返回字符值的字符函数</a>	支持，有差异。
3	<a href="#">返回数值的字符函数</a>	支持，有差异。
4	字符集函数	不支持。
5	Collation函数	不支持。
6	<a href="#">日期时间函数</a>	支持，有差异。
7	<a href="#">通用比较函数</a>	支持，有差异。
8	<a href="#">转换函数</a>	支持，有差异。
9	<a href="#">大对象函数</a>	支持，有差异。
10	集合函数	不支持。
11	<a href="#">层次函数</a>	支持。
12	数据挖掘功能函数	不支持。
13	<a href="#">XML类型函数</a>	支持，有差异。

序号	Oracle数据库	GaussDB数据库
14	JSON函数	不支持。
15	<a href="#">编码解码函数</a>	支持，有差异。
16	<a href="#">空值相关的函数</a>	支持。
17	<a href="#">环境和标识符函数</a>	支持，有差异。

表 7-196 数值函数

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
1	ABS	支持。	-
2	ACOS	支持。	-
3	ASIN	支持。	-
4	ATAN	支持。	-
5	ATAN2	支持。	-
6	BITAND	支持。	-
7	CEIL	支持。	-
8	COS	支持。	-
9	COSH	支持。	-
10	EXP	支持。	-
11	FLOOR	支持。	-
12	LN	支持。	-
13	LOG	支持。	-
14	MOD	支持，有差异。	<ul style="list-style-type: none"> <li>返回类型不一致，Oracle数据库类型包括 BINARY_DOUBLE, BINARY_FLOAT, NUMBER; GaussDB返回类型包括int2, int4, int8, numeric。</li> <li>当两个入参中有一个是int类型时，另一个参数必须为int、numeric类型或字面值整数(无小数点的合法数字如：'12')。</li> </ul>

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
15	NANVL	支持，有差异。	GaussDB不支持直接声明或浮点数除0得到NaN
16	POWER	支持。	-
17	REMAINDER	支持，有差异。	当两个入参中有一个是int类型时，另一个参数必须为int、numeric类型或字面值整数（无小数点的合法数字如：‘12’）
18	ROUND	支持，有差异。	<ul style="list-style-type: none"> <li>第一个参数n的float类型 GaussDB存在精度损失，比Oracle数据库精度低。</li> <li>返回类型不一致。round(n, integer)形式，Oracle数据库NUMBER类型，GaussDB返回numeric类型；round(n)形式，Oracle数据库n的数据类型，GaussDB只能返回float8和numeric类型，缺少float4返回类型。</li> <li>GaussDB判断入参有null，执行框架返回null的逻辑与Oracle数据库不一致 SELECT round(NULL,'q'); Oracle数据库null，GaussDB报错invalid input syntax for integer: "q"。</li> </ul>
19	SIGN	支持。	-
20	SIN	支持。	-
21	SINH	支持。	-
22	SQRT	支持。	-
23	TAN	支持。	-
24	TANH	支持，有差异。	返回值精度有差异。
25	TRUNC	支持。	-
26	WIDTH_BUCKET	支持。	-

表 7-197 返回字符值的字符函数

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
1	CHR	支持，有差异。	<ul style="list-style-type: none"> <li>输入的数字不符合现有字符集时，在JDBC下GaussDB会报错，Oracle数据库会返回乱码。</li> <li>输入0、256等时Oracle数据库会返回Ascii码为0的字符，GaussDB会在'\0'处截断。</li> </ul>
2	CONCAT	支持。	-
3	INITCAP	支持，有差异。	返回值受限于数据库字符集，导致返回结果与Oracle数据库不一致。
4	LOWER	支持，有差异。	<ul style="list-style-type: none"> <li>返回值类型不一致，Oracle数据库和输入类型一致的数据类型。</li> <li>对时间格式上隐式转换问题，输入时间类型时，隐式转换为字符串再进行lower操作。  <pre>SELECT LOWER(TO_DATE('2012-12-10','YYYY-MM-DD'));</pre>                     Oracle返回10-DEC-12，GaussDB返回2012-12-10 00:00:00。</li> <li>返回值受限于数据库字符集，导致返回结果与Oracle数据库不一致。</li> </ul>
5	LPAD	支持。	-
6	LTRIM	支持，有差异。	返回值类型不一致，输入是字符数据类型时Oracle返回VARCHAR2类型，输入是数据库创建时指定的国家字符集时Oracle返回NVARCHAR2类型，输入是LOB类型时Oracle返回LOB类型，GaussDB返回TEXT类型。



序号	Oracle数据库函数	GaussDB数据库是否支持	差异
7	NCHR	支持，有差异。	<ul style="list-style-type: none"> <li>返回值字节长度与Oracle数据库不一致。</li> <li>返回值受限于数据库字符集，导致返回结果与Oracle数据库不一致。</li> <li>返回入参对应的字节数组时，单个字节在[0x80-0xFF]范围，会返回“？”，Oracle数据库返回“？”、或者不输出、或者会报错。</li> </ul>
8	NLS_LOWER	支持，有差异。	<ul style="list-style-type: none"> <li>返回值类型不同，输入是字符数据类型时Oracle返回VARCHAR2类型，输入是LOB类型时Oracle返回LOB类型，GaussDB返回TEXT类型。</li> <li>nlsparam参数Oracle数据库还可以传入除nls_sort外的其他参数种类而不报错，GaussDB只支持nls_sort。</li> <li>返回值受限于数据库字符集，导致返回结果与Oracle数据库不一致。</li> </ul>
9	NLS_UPPER	支持，有差异。	<ul style="list-style-type: none"> <li>返回值类型不同，输入是字符数据类型时Oracle返回VARCHAR2类型，输入是LOB类型时Oracle返回LOB类型，GaussDB返回TEXT类型。</li> <li>nlsparam参数Oracle数据库还可以传入除nls_sort外的其他参数种类而不报错，GaussDB只支持nls_sort。</li> <li>返回值受限于数据库字符集，导致返回结果与Oracle数据库不一致。</li> </ul>
10	NLSSORT	支持。	-

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
11	REGEXP_REPLACE	支持，有差异。	<ul style="list-style-type: none"> <li>● GaussDB入参 source_char不支持 NCLOB类型。</li> <li>● 入参match_param选项 'n' 含义有差异： GaussDB中 'n' 选项与 'm' 选项含义相同，表示采用多行模式匹配；而 Oracle 表示 (.) 能匹配 '\n' 字符，没有指定该选项时默认不能匹配 '\n' 字符。GaussDB中 (.) 默认匹配 '\n' 选项，不需要指定选项。</li> <li>● 不同正则表达式匹配结果可能不一致。 SELECT REGEXP_REPLACE('abc01234xyz', '(.*?)(\d+)(.*)', '#', 'g') FROM DUAL; Oracle报错，GaussDB 返回#####xyz。</li> <li>● 在UTF-8编码字符集下中文输入时匹配结果可能不一致。Oracle需要在GBK字符集实现中文字符串的正则表达式匹配。</li> <li>● 包含部分转义字符的正则表达式匹配结果可能不一致。 SELECT REGEXP_REPLACE('abcabc', '\abc', '#', 'g') FROM DUAL; Oracle报错，GaussDB 返回abcabc。</li> <li>● 匹配规则受 aformat_regexp_match 参数影响，具体影响规格请参见《开发者指南》中“SQL参考 &gt; 函数和操作符 &gt; 字符处理函数和操作符”章节 REGEXP_REPLACE函数部分。</li> </ul>

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
12	REGEXP_SUBSTR	支持，有差异。	匹配规则受 aformat_regexp_match 参数影响，具体影响规格请参见《开发者指南》中“SQL 参考 > 函数和操作符 > 字符处理函数和操作符”章节 REGEXP_SUBSTR 函数部分。
13	REPLACE	支持。	-
14	RPAD	支持。	-
15	RTRIM	支持。	-
16	SUBSTR	支持。	-
17	TRANSLATE	支持。	-
18	TRIM	支持。	-
19	UPPER	支持，有差异。	<ul style="list-style-type: none"> <li>返回值类型不一致，Oracle 数据库和输入类型一致的数据类型，GaussDB 返回 TEXT 类型。</li> <li>对时间格式上隐式转换问题，输入时间类型时，隐式转换为字符串再进行 upper 操作。 SELECT UPPER(TO_DATE('2012-12-10', 'YYYY-MM-DD')); Oracle 返回 10-DEC-12, GaussDB 返回 2012-12-10 00:00:00。</li> <li>返回值受限于数据库字符集，导致返回结果与 Oracle 数据库不一致。</li> </ul>
20	INSTRB	支持。	-

表 7-198 返回数值的字符函数

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
1	ASCII	支持，有差异。	返回值类型不同。Oracle数据库返回类型为uint4，GaussDB为int4。
2	INSTR	支持。	-
3	LENGTH	支持。	-

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
4	REGEXP_COUNT	支持，有差异。	<ul style="list-style-type: none"> <li>● GaussDB入参 source_char不支持NCLOB类型。</li> <li>● 入参 match_param 选项 'n' 含义有差异：GaussDB中 'n' 选项与 'm' 选项含义相同，表示采用多行模式匹配；而Oracle表示(.)能匹配 '\n' 字符，没有指定该选项时默认不能匹配 '\n' 字符。GaussDB中(.)默认匹配 '\n' 选项，不需要指定选项。</li> <li>● 不同正则表达式匹配结果可能不一致。</li> <li>● 在UTF-8编码字符集下中文输入时匹配结果可能不一致。Oracle需要在GBK字符集实现中文字符串的正则表达式匹配。</li> <li>● 包含部分转义字符的正则表达式匹配结果可能不一致</li> <li>● 匹配规则受 aformat_regex p_match参数影响，具体影响规格请参见《开发者指南》中“SQL参考 &gt; 函数和操作符 &gt; 字符处理函数和操作符”章节</li> </ul>

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
			REGEXP_COUNT函数部分。
5	REGEXP_INSTR	支持，有差异。	匹配规则受 aformat_regexp_match 参数影响，具体影响规格请参见《开发者指南》中“SQL参考 > 函数和操作符 > 字符处理函数和操作符”章节 REGEXP_INSTR 函数部分。

表 7-199 日期时间函数

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
1	ADD_MONTHS	支持，有差异。	<ul style="list-style-type: none"> <li>• 公元后到公元前，GaussDB会和Oracle数据库相差1年。</li> <li>• GaussDB的计算结果范围可以到-4714年，Oracle数据库只到-4713年。</li> </ul>
2	CURRENT_DATE	支持，有差异。	GaussDB不支持 nls_date_format 参数设置时间显示格式。
3	CURRENT_TIMESTAMP	支持，有差异。	Oracle数据库参数支持范围（0~9）。GaussDB支持范围（0~6），微秒末位零不显示。
4	DBTIMEZONE	支持，有差异。	GaussDB不支持自带tz的 timestamp 类型接口的调用。
5	EXTRACT	支持。	-
6	LAST_DAY	支持，有差异。	返回值类型不一致。GaussDB返回值类型为 timestamp without time zone，Oracle返回值类型为 date。

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
7	LOCALTIMESTAMP	支持，有差异。	Oracle数据库参数支持范围（0~9）。 GaussDB支持范围（0~6），微秒末位零不显示。
8	MONTHS_BETWEEN	支持，有差异。	入参类型不一致。GaussDB入参均为timestamp without time zone类型，Oracle入参均为date类型。
9	NEW_TIME	支持，有差异。	new_time函数的第一个入参为字面量时，字面量的格式以及函数的返回值类型均与Oracle数据库不一致。
10	NEXT_DAY	支持。	-
11	NUMTODSINTERVAL	支持，有差异。	GaussDB不支持dsinterval类型，暂时用interval兼容dsinterval类型。
12	NUMTOYMINTERVAL	支持，有差异。	GaussDB不支持yminterval类型，暂时用interval兼容yminterval类型。
13	SESSIONTIMEZONE	支持，有差异。	<ul style="list-style-type: none"> <li>赋值语法差异： GaussDB为set session time zone 8。Oracle为alter session set time_zone= '+08:00'。</li> <li>默认值差异：GaussDB为时区名称形式如:PRC。Oracle为偏移量形式，如：+08:00。</li> </ul>
14	SYS_EXTRACT_UTC	支持。	-
15	SYSDATE	支持，有差异。	返回值类型不一致。 GaussDB返回值类型为timestamp without time zone，Oracle返回值类型为date。
16	TO_CHAR	支持，有差异。	fmt '5' 未在Oracle数据库文档中，未适配。
17	TO_DSINTERVAL	支持，有差异。	GaussDB不支持dsinterval类型，暂时用interval兼容dsinterval类型。

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
18	TO_TIMESTAMP	支持，有差异。	GaussDB毫秒计算只支持6位，Oracle数据库支持9位。
19	TO_TIMESTAMP_TZ	支持，有差异。	GaussDB的timestampz等价于Oracle的timestampwithlocaltimezone，缺少Oracle对应的timestampz类型。nls_date_language只支持ENGLISH和AMERICAN两种语言。
20	TO_YMINTERVAL	支持，有差异。	GaussDB不支持yminterval类型，暂时用interval兼容yminterval类型。
21	TZ_OFFSET	支持，有差异。	接收一个时区名称为入参的时候，时区名称的类型比Oracle数据库要少。

表 7-200 通用比较函数

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
1	GREATEST	支持，有差异。	<ul style="list-style-type: none"> <li>GaussDB不支持NLS_SORT参数指定的比较方式，只支持二进制比较。</li> <li>GaussDB不支持多语种的表达式。</li> </ul>
2	LEAST	支持，有差异。	<ul style="list-style-type: none"> <li>GaussDB不支持NLS_SORT参数指定的比较方式，只支持二进制比较。</li> <li>GaussDB不支持多语种的表达式。</li> </ul>

表 7-201 转换函数

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
1	ASCIISTR	支持。	-



序号	Oracle数据库函数	GaussDB数据库是否支持	差异
2	CAST	支持，有差异。	<ul style="list-style-type: none"> <li>GaussDB不支持multiset子句。</li> <li>GaussDB不支持nlsparm参数。</li> </ul>
3	HEXTORAW	支持。	-
4	RAWTOHEX	支持。	-
5	TO_BINARY_DOUBLE	支持，有差异。	GaussDB不支持nlsparm参数。
6	TO_BINARY_FLOAT	支持，有差异。	GaussDB不支持nlsparm参数。
7	TO_BLOB	支持，有差异。	<ul style="list-style-type: none"> <li>GaussDB不支持long raw类型。</li> <li>GaussDB不支持bfile、mime_type类型。</li> </ul>
8	TO_CLOB	支持。	-
9	TO_DATE	支持，有差异。	<ul style="list-style-type: none"> <li>不支持多语种参数。</li> <li>返回类型不一致。</li> <li>缺少控制参数NLS_DATE_FORMAT。</li> <li>部分format格式不支持。</li> <li>fmt = 'j'。1582年10月15日之前Oracle数据库与GaussDB输出不一致。</li> <li>无分割符时，不保证与Oracle数据库完全一致。如 to_date( '220725' , 'yymmdd' )，yy/rr按照固定长度4解析，会解析为2207年25月，25非法月份则会报错。</li> </ul>
10	TO_MULTI_BYTE	支持。	-
11	TO_NCHAR	支持，有差异。	<p>GaussDB: 将入参的类型转换为text。</p> <p>Oracle: 将入参的类型转换为国家字符集 ( national character set ) 。</p>

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
12	TO_NUMBER	支持，有差异。	GaussDB不支持nlsparam参数。
13	TO_SINGLE_BYTE	支持。	-
14	TREAT	支持，有差异。	GaussDB不支持使用“.”操作符取值，不支持转化为object类型。
15	UNISTR	支持，有差异。	GaussDB只支持UTF-8编码，Oracle数据库支持UTF-8和UTF-16编码。

表 7-202 大对象函数

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
1	EMPTY_BLOB	支持。	-
2	EMPTY_CLOB	支持，有差异。	GaussDB的clob类型不支持Oracle数据库中的定位器概念。

表 7-203 层次函数

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
1	SYS_CONNECT_BY_PATH	支持，有差异。	GaussDB中第一个入参指定的列的类型仅支持CHAR/VARCHAR/NVARCHAR2/TEXT类型。

表 7-204 XML 类型函数

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
1	EXISTSNODE	支持，有差异。	在入参有命名空间时，xpath和命名空间都需要定义别名。
2	EXTRACTVALUE	支持，有差异。	目前仅支持xpath1.0版本。

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
3	SYS_XMLAGG	支持，有差异。	xmlagg的别名，可使用xmlagg代替。
4	XMLAGG	支持。	-
5	XMLCOMMENT	支持。	-
6	XMLCONCAT	支持	-
7	XMLELEMENT	支持，有差异。	xmlelement和xmlattributes的name字段赋值为NULL时，行为与Oracle不一致。xmlelement的name字段赋值为NULL时，结果显示name信息为空，且不显示属性信息。xmlattributes的name字段赋值为NULL时，不显示属性信息。
8	XMLEXISTS	支持，有差异。	GaussDB入参为xml类型。
9	XMLFOREST	支持，有差异。	GaussDB返回值为xml类型。GaussDB不支持EVALNAME语法。
10	XMLPARSE	支持，有差异。	GaussDB返回值为xml类型。GaussDB不支持WELLFORMED语法。
11	XMLROOT	支持，有差异。	GaussDB返回值为xml类型。
12	JSON_OBJECT	支持。	-
13	XMLTABLE	支持，有差异。	GaussDB从xml中选取数据使用的为XPath 1.0表达式，不支持声明默认命名空间，不支持多组输入及取别名，不支持省略传入数据的passing_clause子句，不支持RETURNING SEQUENCE BY REF子句和( SEQUENCE ) BY REF子句。
14	GETSTRINGVAL	支持。	-
15	GETCLOBVAL	支持。	-

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
16	XMLSEQUENCE	支持。	-

表 7-205 编码解码函数

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
1	DECODE	支持。	-
2	DUMP	支持，有差异。	<p>因存储格式不同，GaussDB数值和时间类型返回结果和Oracle数据库不一致。如：</p> <ul style="list-style-type: none"> <li>• GaussDB中SELECT dump(123); 返回 Typ=23 Len=4: 123,0,0,0。</li> <li>• Oracle中SELECT dump(123) FROM dual; 返回Typ=2 Len=3: 194,2,24。</li> </ul>
3	ORA_HASH	支持，有差异。	<p>GaussDB中有以下行为：</p> <ul style="list-style-type: none"> <li>• 时间类型的入参转换成字符串类型再进行hash。</li> <li>• 不支持maxbucket参数。</li> </ul>
4	VSIZE	支持，有差异。	<p>因存储格式不同，GaussDB数值和时间类型返回结果和Oracle数据库不一致。如：</p> <ul style="list-style-type: none"> <li>• GaussDB中SELECT vsize(999); 返回4。</li> <li>• Oracle中SELECT vsize(999) FROM dual; 返回3。</li> </ul>

表 7-206 空值相关的函数

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
1	COALESCE	支持。	-
2	LNNVL	支持。	-
3	NULLIF	支持。	-
4	NVL	支持。	-
5	NVL2	支持。	-

表 7-207 环境和标识符函数

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
1	SYS_CONTEXT	支持，有差异。	<p>GaussDB对不支持的参数返回NULL。</p> <p>以下为不支持的参数列表：</p> <ul style="list-style-type: none"> <li>'action'</li> <li>'is_application_root'</li> <li>'is_application_pdb'</li> <li>'audited_cursorid'</li> <li>'authenticated_identity'</li> <li>'authentication_data'</li> <li>'authentication_method'</li> <li>'cdb_domain'</li> <li>'cdb_name'</li> <li>'client_identifier'</li> <li>'con_id'</li> <li>'con_name'</li> <li>'current_sql_length'</li> <li>'db_domain'</li> <li>'db_supplemental_log_level'</li> <li>'dblink_info'</li> <li>'drain_status'</li> <li>'entryid'</li> <li>'enterprise_identity'</li> <li>'fg_job_id'</li> <li>'global_uid'</li> <li>'identification_type'</li> <li>'instance'</li> <li>'is_dg_rolling_upgrade'</li> <li>'ldap_server_type'</li> <li>'module'</li> <li>'network_protocol'</li> <li>'nls_calendar'</li> <li>'nls_sort'</li> <li>'nls_territory'</li> </ul>

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
			'oracle_home' 'os_user' 'platform_slash' 'policy_invoker' 'proxy_enterprise_identity' 'proxy_user' 'proxy_userid' 'scheduler_job' 'session_edition_id' 'session_edition_name' 'sessionid' 'statementid' 'terminal' 'unified_audit_sessionid' 'session_default_collation' 'client_info' 'bg_job_id' 'client_program_name' 'current_bind' 'global_context_memory' 'host' 'current_sqln'
2	SYS_GUID	支持。	-
3	USER	支持，有差异。	返回值类型不一致，GaussDB返回值类型为name，Oracle返回值类型为varchar2。

### 7.2.10.2 其它函数

序号	Oracle数据库	GaussDB数据库
1	<b>聚合函数</b>	支持

序号	Oracle数据库	GaussDB数据库
2	分析函数	支持
3	对象引用函数	不支持
4	模型函数	不支持
5	OLAP函数	不支持
6	数据盒功能函数	不支持
7	关于用户定义的函数	支持

表 7-208 聚合函数

序号	Oracle数据库函数	GaussDB数据库是否支持	差异
1	AVG	支持。	-
2	CORR	支持。	-
3	COUNT	支持。	-
4	COVAR_POP	支持。	-
5	COVAR_SAMP	支持。	-
6	CUME_DIST	支持。	-
7	DENSE_RANK	支持。	-
8	FIRST	支持。	-
9	GROUPING	支持。	-
10	LAST	支持。	-
11	LISTAGG	支持。	-
12	MAX	支持。	-
13	MEDIAN	支持。	-
14	MIN	支持。	-
15	PERCENT_RANK	支持。	-
16	PERCENTILE_CONT	支持。	-
17	RANK	支持。	-
18	REGR_ (Linear Regression)	支持。	-
19	STDDEV	支持。	-



序号	Oracle数据库函数	GaussDB数据库是否支持	差异
20	STDDEV_POP	支持。	-
21	STDDEV_SAMP	支持。	-
22	SUM	支持。	-
23	VAR_POP	支持。	-
24	VAR_SAMP	支持。	-
25	VARIANCE	支持。	-
26	WM_CONCAT	支持，有差异。	GaussDB中使用LISTAGG、STRING_AGG函数，兼容Oracle此功能。

表 7-209 分析函数

序号	Oracle数据库函数	GaussDB数据库是否支持
1	FIRST_VALUE	支持
2	LAG	支持
3	LAST_VALUE	支持
4	LEAD	支持
5	NTH_VALUE	支持
6	NTILE	支持
7	ROW_NUMBER	支持
8	RATIO_TO_REPORT	支持

## 7.2.11 系统视图

GaussDB数据库兼容了部分Oracle数据库的系统视图，兼容的详细列表如下。

更多系统视图的字段说明信息请参考《开发者指南》中“系统视图”章节。

表 7-210 支持视图列表

序号	Oracle数据库	GaussDB数据库
1	ALL_ALL_TABLES	DB_ALL_TABLES
2	ALL_COL_PRIVS	DB_COL_PRIVS

序号	Oracle数据库	GaussDB数据库
3	ALL_COLL_TYPES	DB_COLL_TYPES
4	ALL_ERRORS	DB_ERRORS
5	ALL_IND_COLUMNS	DB_IND_COLUMNS
6	ALL_COL_COMMENTS	DB_COL_COMMENTS
7	ALL_CONS_COLUMNS	DB_CONS_COLUMNS
8	ALL_CONSTRAINTS	DB_CONSTRAINTS
9	ALL_DEPENDENCIES	DB_DEPENDENCIES
10	ALL_DIRECTORIES	DB_DIRECTORIES
11	ALL_IND_EXPRESSIONS	DB_IND_EXPRESSIONS
12	ALL_IND_PARTITIONS	DB_IND_PARTITIONS
13	ALL_INDEXES	DB_INDEXES
14	ALL_IND_SUBPARTITIONS	DB_IND_SUBPARTITIONS
15	ALL_OBJECTS	DB_OBJECTS
16	ALL_PART_COL_STATISTICS	DB_PART_COL_STATISTICS
17	ALL_PART_KEY_COLUMNS	DB_PART_KEY_COLUMNS
18	ALL_PART_TABLES	DB_PART_TABLES
19	ALL_SCHEDULER_JOB_ARGS	DB_SCHEDULER_JOB_ARGS
20	ALL_SCHEDULER_PROGRAM_ARGS	DB_SCHEDULER_PROGRAM_ARGS
21	ALL_SEQUENCES	DB_SEQUENCES
22	ALL_SUBPART_COL_STATISTICS	DB_SUBPART_COL_STATISTICS
23	ALL_SUBPART_KEY_COLUMNS	DB_SUBPART_KEY_COLUMNS
24	ALL_SYNONYMS	DB_SYNONYMS
25	ALL_TAB_COL_STATISTICS	DB_TAB_COL_STATISTICS
26	ALL_TAB_COMMENTS	DB_TAB_COMMENTS
27	ALL_TAB_HISTOGRAMS	DB_TAB_HISTOGRAMS
28	ALL_TAB_STATS_HISTORY	DB_TAB_STATS_HISTORY
29	ALL_TYPES	DB_TYPES
30	ALL_PART_INDEXES	DB_PART_INDEXES

序号	Oracle数据库	GaussDB数据库
31	ALL_PROCEDURES	DB_PROCEDURES
32	ALL_SOURCE	DB_SOURCE
33	ALL_TAB_COLUMNS	DB_TAB_COLUMNS
34	ALL_TAB_PARTITIONS	DB_TAB_PARTITIONS
35	ALL_TAB_SUBPARTITIONS	DB_TAB_SUBPARTITIONS
36	ALL_TABLES	DB_TABLES
37	ALL_TRIGGERS	DB_TRIGGERS
38	ALL_USERS	DB_USERS
39	ALL_VIEWS	DB_VIEWS
40	DBA_AUDIT_OBJECT	ADM_AUDIT_OBJECT
41	DBA_AUDIT_SESSION	ADM_AUDIT_SESSION
42	DBA_AUDIT_STATEMENT	ADM_AUDIT_STATEMENT
43	DBA_AUDIT_TRAIL	ADM_AUDIT_TRAIL
44	DBA_COL_COMMENTS	ADM_COL_COMMENTS
45	DBA_COL_PRIVS	ADM_COL_PRIVS
46	DBA_COLL_TYPES	ADM_COLL_TYPES
47	DBA_ARGUMENTS	ADM_ARGUMENTS
48	DBA_CONSTRAINTS	ADM_CONSTRAINTS
49	DBA_DATA_FILES	ADM_DATA_FILES
50	DBA_CONS_COLUMNS	ADM_CONS_COLUMNS
51	DBA_DEPENDENCIES	ADM_DEPENDENCIES
52	DBA_DIRECTORIES	ADM_DIRECTORIES
53	DBA_PART_COL_STATISTICS	ADM_PART_COL_STATISTICS
54	DBA_PART_TABLES	ADM_PART_TABLES
55	DBA_RECYCLEBIN	ADM_RECYCLEBIN
56	DBA_ROLE_PRIVS	ADM_ROLE_PRIVS
57	DBA_ROLES	ADM_ROLES
58	DBA_SCHEDULER_JOB_ARGS	ADM_SCHEDULER_JOB_ARGS
59	DBA_SCHEDULER_PROGRAMS	ADM_SCHEDULER_PROGRAMS

序号	Oracle数据库	GaussDB数据库
60	DBA_SCHEDULER_PROGRAM_ARGS	ADM_SCHEDULER_PROGRAM_ARGS
61	DBA_HIST_SNAPSHOT	ADM_HIST_SNAPSHOT
62	DBA_HIST_SQL_PLAN	ADM_HIST_SQL_PLAN
63	DBA_HIST_SQLSTAT	ADM_HIST_SQLSTAT
64	DBA_HIST_SQLTEXT	ADM_HIST_SQLTEXT
65	DBA_ILMDATAMOVEMENTPOLICIES	GS_ADM_ILMDATAMOVEMENTPOLICIES
66	DBA_ILMEVALUATIONDETAILS	GS_ADM_ILMEVALUATIONDETAILS
67	DBA_ILMOBJECTS	GS_ADM_ILMOBJECTS
68	DBA_ILMPARAMETERS	GS_ADM_ILMPARAMETERS
69	DBA_ILMPOLICIES	GS_ADM_ILMPOLICIES
70	DBA_ILMRESULTS	GS_ADM_ILMRESULTS
71	DBA_ILMTASKS	GS_ADM_ILMTASKS
72	DBA_IND_COLUMNS	ADM_IND_COLUMNS
73	DBA_IND_EXPRESSIONS	ADM_IND_EXPRESSIONS
74	DBA_IND_PARTITIONS	ADM_IND_PARTITIONS
75	DBA_INDEXES	ADM_INDEXES
76	DBA_OBJECTS	ADM_OBJECTS
77	DBA_PART_INDEXES	ADM_PART_INDEXES
78	DBA_PROCEDURES	ADM_PROCEDURES
79	DBA_SCHEDULER_JOBS	ADM_SCHEDULER_JOBS
80	DBA_SCHEDULER_RUNNING_JOBS	ADM_SCHEDULER_RUNNING_JOBS
81	DBA_SEGMENTS	ADM_SEGMENTS
82	DBA_SEQUENCES	ADM_SEQUENCES
83	DBA_SOURCE	ADM_SOURCE
84	DBA_IND_SUBPARTITIONS	ADM_IND_SUBPARTITIONS
85	DBA_SUBPART_COL_STATISTICS	ADM_SUBPART_COL_STATISTICS

序号	Oracle数据库	GaussDB数据库
86	DBA_SUBPART_KEY_COLUMNS	ADM_SUBPART_KEY_COLUMNS
87	DBA_SYS_PRIVS	ADM_SYS_PRIVS
88	DBA_TAB_COL_STATISTICS	ADM_TAB_COL_STATISTICS
89	DBA_TAB_HISTOGRAMS	ADM_TAB_HISTOGRAMS
90	DBA_TAB_STATISTICS	ADM_TAB_STATISTICS
91	DBA_TAB_STATS_HISTORY	ADM_TAB_STATS_HISTORY
92	DBA_TABLESPACES	ADM_TABLESPACES
93	DBA_TYPES	ADM_TYPES
94	DBA_USERS	ADM_USERS
95	DBA_SYNONYMS	ADM_SYNONYMS
96	DBA_TAB_COLS	ADM_TAB_COLS
97	DBA_TAB_COLUMNS	ADM_TAB_COLUMNS
98	DBA_TAB_COMMENTS	ADM_TAB_COMMENTS
99	DBA_TABLES	ADM_TABLES
100	DBA_TAB_PARTITIONS	ADM_TAB_PARTITIONS
101	DBA_TAB_SUBPARTITIONS	ADM_TAB_SUBPARTITIONS
102	DBA_TRIGGERS	ADM_TRIGGERS
103	DBA_TYPE_ATTRS	ADM_TYPE_ATTRS
104	DBA_VIEWS	ADM_VIEWS
105	ROLE_ROLE_PRIVS	ROLE_ROLE_PRIVS
106	ROLE_SYS_PRIVS	ROLE_SYS_PRIVS
107	ROLE_TAB_PRIVS	ROLE_TAB_PRIVS
108	USER_COL_COMMENTS	MY_COL_COMMENTS
109	USER_COL_PRIVS	MY_COL_PRIVS
110	USER_COLL_TYPES	MY_COLL_TYPES
111	USER_CONSTRAINTS	MY_CONSTRAINTS
112	USER_DEPENDENCIES	MY_DEPENDENCIES
113	DICTIONARY	DICTIONARY
114	DICTIONARY	DICTIONARY

序号	Oracle数据库	GaussDB数据库
115	DUAL	DUAL
116	NLS_DATABASE_PARAMETERS	NLS_DATABASE_PARAMETERS
117	NLS_INSTANCE_PARAMETERS	NLS_INSTANCE_PARAMETERS
118	PLAN_TABLE	PLAN_TABLE
119	USER_ERRORS	MY_ERRORS
120	USER_ILMDATAMOVEMENTPOLICIES	GS_MY_ILMDATAMOVEMENTPOLICIES
121	USER_ILMEVALUATIONDETAILS	GS_MY_ILMEVALUATIONDETAILS
122	USER_ILMOBJECTS	GS_MY_ILMOBJECTS
123	USER_ILMPOLICIES	GS_MY_ILMPOLICIES
124	USER_ILMRESULTS	GS_MY_ILMRESULTS
125	USER_ILMTASKS	GS_MY_ILMTASKS
126	USER_IND_COLUMNS	MY_IND_COLUMNS
127	USER_IND_EXPRESSIONS	MY_IND_EXPRESSIONS
128	USER_IND_PARTITIONS	MY_IND_PARTITIONS
129	USER_IND_SUBPARTITIONS	MY_IND_SUBPARTITIONS
130	USER_INDEXES	MY_INDEXES
131	USER_JOBS	MY_JOBS
132	USER_OBJECTS	MY_OBJECTS
133	USER_PART_COL_STATISTICS	MY_PART_COL_STATISTICS
134	USER_PART_INDEXES	MY_PART_INDEXES
135	USER_PART_TABLES	MY_PART_TABLES
136	USER_PROCEDURES	MY_PROCEDURES
137	USER_RECYCLEBIN	MY_RECYCLEBIN
138	USER_SCHEDULER_JOB_ARGS	MY_SCHEDULER_JOB_ARGS
139	USER_SCHEDULER_PROGRAM_ARGS	MY_SCHEDULER_PROGRAM_ARGS
140	USER_SEQUENCES	MY_SEQUENCES
141	USER_SOURCE	MY_SOURCE

序号	Oracle数据库	GaussDB数据库
142	USER_SUBPART_KEY_COLUMNS	MY_SUBPART_KEY_COLUMNS
143	USER_SYNONYMS	MY_SYNONYMS
144	USER_SYS_PRIVS	MY_SYS_PRIVS
145	USER_TAB_COL_STATISTICS	MY_TAB_COL_STATISTICS
146	USER_TAB_COLUMNS	MY_TAB_COLUMNS
147	USER_TAB_COMMENTS	MY_TAB_COMMENTS
148	USER_TAB_HISTOGRAMS	MY_TAB_HISTOGRAMS
149	USER_TAB_PARTITIONS	MY_TAB_PARTITIONS
150	USER_TAB_STATISTICS	MY_TAB_STATISTICS
151	USER_TAB_STATS_HISTORY	MY_TAB_STATS_HISTORY
152	USER_TAB_SUBPARTITIONS	MY_TAB_SUBPARTITIONS
153	USER_TABLES	MY_TABLES
154	USER_TABLESPACES	MY_TABLESPACES
155	USER_TRIGGERS	MY_TRIGGERS
156	USER_TYPE_ATTRS	MY_TYPE_ATTRS
157	USER_TYPES	MY_TYPES
158	USER_VIEWS	MY_VIEWS
159	V\$GLOBAL_TRANSACTION	V\$GLOBAL_TRANSACTION
160	V\$NLS_PARAMETERS	V\$NLS_PARAMETERS
161	V\$SESSION_WAIT	V\$SESSION_WAIT
162	V\$SYSSTAT	V\$SYSSTAT
163	V\$SYSTEM_EVENT	V\$SYSTEM_EVENT
164	V\$VERSION	V\$VERSION
165	V\$INSTANCE	V_INSTANCE
166	GV\$INSTANCE	GV_INSTANCE
167	V\$MYSTAT	V_MYSTAT
168	V\$SESSION	V_SESSION
169	GV\$SESSION	GV_SESSION
170	V\$SESSION_LONGOPS	DV_SESSION_LONGOPS

序号	Oracle数据库	GaussDB数据库
171	V\$SESSION	DV_SESSIONS
172	ALL_ARGUMENTS	DB_ARGUMENTS
173	USER_CONS_COLUMNS	MY_CONS_COLUMNS
174	USER_PART_KEY_COLUMNS	MY_PART_KEY_COLUMNS
175	USER_SUBPART_COL_STATISTICS	MY_SUBPART_COL_STATISTICS
176	USER_ROLE_PRIVS	MY_ROLE_PRIVS
177	DBA_TAB_PRIVS	ADM_TAB_PRIVS
178	USER_SCHEDULER_JOBS	MY_SCHEDULER_JOBS
179	V\$LOCK	V\$LOCK
180	V\$DBLINK	V\$DBLINK
181	V\$OPEN_CURSOR	V\$OPEN_CURSOR
182	ALL_TAB_PRIVS	DB_TAB_PRIVS
183	ALL_TAB_MODIFICATIONS	DB_TAB_MODIFICATIONS
184	USER_TAB_MODIFICATIONS	MY_TAB_MODIFICATIONS
185	USER_AUDIT_TRAIL	MY_AUDIT_TRAIL

## 7.2.12 高级包

GaussDB数据库兼容的高级包如表1所示。

表 7-211 支持高级包列表

序号	Oracle数据库	GaussDB数据库	差异
1	<b>DBMS_LOB</b>	DBE_LOB	GaussDB具体用法请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_LOB”章节。
2	<b>DBMS_RANDOM</b>	DBE_RANDOM	GaussDB具体用法请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_RANDOM”章节。



序号	Oracle数据库	GaussDB数据库	差异
3	<b>DBMS_OUTPUT</b>	DBE_OUTPUT	GaussDB具体用法请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_OUTPUT”章节。
4	<b>UTL_RAW</b>	DBE_RAW	GaussDB具体用法请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_RAW”章节。
5	<b>DBMS_SCHEDULER</b>	DBE_SCHEDULER	GaussDB具体用法请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_SCHEDULER”章节。
6	<b>DBMS_UTILITY</b>	DBE_UTILITY	GaussDB具体用法请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_UTILITY”章节。
7	<b>DBMS_SQL</b>	DBE_SQL	GaussDB具体用法请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_SQL”章节。
8	<b>UTL_FILE</b>	DBE_FILE	GaussDB具体用法请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_FILE”章节。

序号	Oracle数据库	GaussDB数据库	差异
9	<b>DBMS_SESSION</b>	DBE_SESSION	GaussDB具体用法请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_SESSION”章节。
10	<b>UTL_MATCH</b>	DBE_MATCH	GaussDB具体用法请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_MATCH”章节。
11	<b>DBMS_APPLICATION_INFO</b>	DBE_APPLICATION_INFO	GaussDB具体用法请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_APPLICATION_INFO”章节。
12	<b>DBMS_XMLDOM</b>	DBE_XMLDOM	GaussDB中具体信息请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_XMLDOM”章节。
13	<b>DBMS_XMLPARSER</b>	DBE_XMLPARSER	GaussDB中具体信息请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_XMLPARSER”章节。
14	<b>DBMS_ILM</b>	DBE_ILM	GaussDB中具体信息请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_ILM”章节。

序号	Oracle数据库	GaussDB数据库	差异
15	<b>DBMS_ILM_ADMIN</b>	DBE_ILM_ADMIN	GaussDB中具体信息请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_ILM_ADMIN”章节。
16	<b>DBMS_COMPRESSION</b>	DBE_COMPRESSION	GaussDB中具体信息请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_COMPRESSION”章节。
17	<b>DBMS_HEAT_MAP</b>	DBE_HEAT_MAP	GaussDB中具体信息请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_HEAT_MAP”章节。
18	<b>DBMS_DESCRIBE</b>	DBE_DESCRIBE	GaussDB中具体信息请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_DESCRIBE”章节。
19	<b>DBMS_XMLGEN</b>	DBE_XMLGEN	GaussDB中具体信息请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_XMLGEN”章节。
20	<b>DBMS_STATS</b>	DBE_STATS	GaussDB中具体信息请参见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐） > DBE_STATS”章节。

表 7-212 DBMS\_LOB 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	APPEND Procedures	APPEND Procedures	-
2	CLOB2FILE Procedure	不支持	-
3	CLOSE Procedure	BFILECLOSE Procedure	GaussDB: 参数类型为BFILE, 不存在函数重载。 Oracle: 该过程存在3个重载, 3个重载的参数lob_loc、lob_loc和file_loc的类型分别为BLOB、CLOB CHARACTER SET ANY_CS和BFILE。
4	COMPARE Functions	COMPARE Functions	GaussDB: 存在3个重载函数, 对于第三个参数 ( len ) 均为BIGINT。 Oracle: 存在3个重载函数, 对于第三个参数 ( amount ) 均为INTEGER。
5	CONVERTTOBLOB Procedure	LOB_CONVERTTO BLOB Procedure	GaussDB: 该过程共有5个参数, 且第3、4、5个参数类型为BIGINT。 Oracle: 该过程共有8个参数, 在GaussDB所有参数的基础上增加了blob_csid、lang_context和warning3个参数, 类型分别为NUMBER、INTEGER和INTEGER, 且第3、4、5个参数类型为INTEGER。

序号	Oracle数据库	GaussDB数据库	差异
6	CONVERTTOCLOB Procedure	LOB_CONVERTTOCLOB Procedure	GaussDB: 该过程共有5个参数。第3、4、5个参数类型为BIGINT。 Oracle: 该过程共有8个参数。第3、4、5个参数类型为INTEGER。Oracle的该过程在GaussDB所有参数的基础上增加了3个参数:blob_csid、lang_context和warning, 参数类型分别为NUMBER、INTEGER和INTEGER。
7	COPY Procedures	LOB_COPY Functions	-
8	COPY_DBFS_LINK Procedures	不支持	-
9	COPY_FROM_DBFS_LINK	不支持	-

序号	Oracle数据库	GaussDB数据库	差异
10	CREATETEMPORARY Procedures	CREATE_TEMPORARY Procedures	<p>GaussDB: 该过程存在2个重载。第一个重载过程的第一个参数 (lob_loc) 为 BLOB, 第二个重载过程的第一个参数 (lob_loc) 为 CLOB; 两个重载过程的第三个参数 (dur) 为 INTEGER, 默认值为10。</p> <p>Oracle: 该过程存在2个重载。第一个重载过程的第一个参数 (lob_loc) 为 BLOB, 第二个重载过程的第一个参数 (lob_loc) 为 CLOB; 两个重载过程的第三个参数 (dur) 的参数类型为 PLS_INTEGER, 第一个重载过程的 dur 默认值为 DBMS_LOB.SESSION, 第二个重载过程的 dur 默认值为10。</p>
11	DBFS_LINK_GENERATE_PATH Functions	不支持	-
12	ERASE Procedures	LOB_ERASE Procedures	-
13	FILECLOSE Procedure	不支持	-
14	FILECLOSEALL Procedure	不支持	-
15	FILEEXISTS Function	不支持	-
16	FILEGETNAME Procedure	不支持	-

序号	Oracle数据库	GaussDB数据库	差异
17	FILEISOPEN Function	不支持	-
18	FILEOPEN Procedure	不支持	-
19	FRAGMENT_DELE TE Procedure	不支持	-
20	FRAGMENT_INSE RT Procedures	不支持	-
21	FRAGMENT_MOV E Procedure	不支持	-
22	FRAGMENT_REPL ACE Procedures	不支持	-
23	FREETEMPORARY Procedures	不支持	-
24	GET_DBFS_LINK Functions	不支持	-
25	GET_DBFS_LINK_S TATE Procedures	不支持	-
26	GETCHUNKSIZE Functions	GETCHUNKSIZE Functions	-
27	GETCONTENTTYP E Functions	不支持	-
28	GETLENGTH Functions	不支持	-
29	GETOPTIONS Functions	不支持	-
30	GET_STORAGE_LI MIT Function	不支持	-
31	INSTR Functions	MATCH Functions	GaussDB: 存在3个重载函数。3个重载函数的第三、四个参数均为BIGINT。 Oracle: 存在3个重载函数。3个重载函数的第三、四个参数均为INTEGER。
32	ISOPEN Functions	不支持	-

序号	Oracle数据库	GaussDB数据库	差异
33	ISREMOTE Function	不支持	-
34	ISSECUREFILE Function	不支持	-
35	ISTEMPORARY Functions	不支持	-
36	LOADBLOBFROM FILE Procedure	LOADBLOBFROM BFILE Procedure	-
37	LOADCLOBFROM FILE Procedure	LOADCLOBFROM BFILE Procedure	-
38	LOADFROMFILE Procedure	LOADFROMBFILE Procedure	-
39	MOVE_TO_DBFS_ LINK Procedures	不支持	-



序号	Oracle数据库	GaussDB数据库	差异
40	OPEN Procedures	BFILEOPEN Procedure	<p>GaussDB: 该过程不存在重载。第一个参数 ( bfile ) 类型为 DBE_LOB.BFILE, 第二个参数 ( open_mode ) 类型为TEXT, 且只支持read模式。</p> <p>Oracle: 该过程存在3个重载。第一个重载过程的第一个参数 ( lob_loc ) 类型为NOCOPY BLOB, 第二个参数 ( openmode ) 类型为 BINARY_INTEGER ; 第二个重载过程的第一个参数 ( lob_loc ) 类型为 NOCOPY CLOB CHARACTER SET ANY_CS, 第二个参数 ( openmode ) 类型为 BINARY_INTEGER ; 第三个重载过程的第一个参数 ( file_loc ) 类型为 NOCOPY BFILE, 第二个参数 ( openmode ) 类型为 BINARY_INTEGER , 且只能为 file_readonly。</p>

序号	Oracle数据库	GaussDB数据库	差异
41	READ Procedures	READ Procedures	GaussDB: 该过程存在2个重载。 Oracle: 该过程存在3个重载。其中前两个重载与GaussDB无差异, 第三个过程重载包括4个参数: file_loc、amount、offset和buffer, 其类型分别为BFILE、NOCOPY INTEGER、INTEGER和RAW。
42	SET_DBFS_LINK Procedures	不支持	-
43	SETCONTENTTYPE Procedure	不支持	-
44	SETOPTIONS Procedures	不支持	-
45	SUBSTR Functions	LOB_SUBSTR Functions	-
46	TRIM Procedures	STRIP Functions	GaussDB: 该过程存在2个重载。两个重载过程的第二个参数 (newlen) 均为BIGINT。 Oracle: 该过程存在2个重载。两个重载过程的第二个参数 (newlen) 均为INTEGER。
47	WRITE Procedures	WRITE Functions	-
48	WRITEAPPEND Procedures	WRITEAPPEND Functions	-

表 7-213 DBMS\_RANDOM 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	INITIALIZE Procedure	不支持	-

序号	Oracle数据库	GaussDB数据库	差异
2	NORMAL Function	不支持	-
3	RANDOM Function	不支持	-
4	SEED Procedures	DBE_RANDOM.SE T_SEED Function	GaussDB: 该函数无重载, 参数类型为INTEGER。 Oracle: 该过程存在2个重载, 2个重载过程的参数类型分别为VARCHAR2和 BINARY_INTEGER。
5	STRING Function	不支持	-
6	TERMINATE Procedure	不支持	-
7	VALUE Functions	DBE_RANDOM.GE T_VALUE Function	GaussDB: 该函数无重载。 Oracle: 存在无参数的VALUE函数重载, 返回NUMBER类型。

表 7-214 DBMS\_OUTPUT 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	DISABLE Procedure	DISABLE Function	-
2	ENABLE Procedure	ENABLE Function	-
3	GET_LINE Procedure	GET_LINE Function	-

序号	Oracle数据库	GaussDB数据库	差异
4	GET_LINES Procedure	GET_LINES Function	<p>GaussDB: 该函数无重载, 首个参数 ( lines ) 数据类型为VARCHAR[]。</p> <p>Oracle: 该过程存在2个重载, 2个重载过程的首个参数 ( lines ) 分别为CHARARR和DBMSOUTPUT_LINERARRAY。</p>
5	NEW_LINE Procedure	NEW_LINE Function	-
6	PUT Procedure	PUT Function	<p>GaussDB: 当数据库服务端字符集 server_encoding不是UTF8编码格式且入参的字符编码是合法的UTF8编码时, 该函数不会区分入参的数据类型, 都会先把该字符编码按照“UTF8 &gt; server_encoding”的转换关系进行转换后再输出。</p> <p>Oracle: 当数据库服务端字符集 server_encoding不是UTF8编码格式且入参的字符编码是合法的UTF8编码时, 若入参类型是NVARCHAR2, 则该过程会先把该字符编码按照“UTF8 &gt; server_encoding”的转换关系进行转换后再输出; 若入参为其他字符类型, 则会将该字符编码视作非法字符, 以占位符的形式输出。</p>

序号	Oracle数据库	GaussDB数据库	差异
7	PUT_LINE Procedure	PUT_LINE Function	<p>GaussDB: 当数据库服务端字符集 server_encoding 不是 UTF8 编码格式且入参的字符编码是合法的 UTF8 编码时, 该函数不会区分入参的数据类型, 都会先把该字符编码按照 “UTF8 &gt; server_encoding” 的转换关系进行转换后再输出。</p> <p>Oracle: 当数据库服务端字符集 server_encoding 不是 UTF8 编码格式且入参的字符编码是合法的 UTF8 编码时, 若入参类型是 NVARCHAR2, 则该过程会先把该字符编码按照 “UTF8 &gt; server_encoding” 的转换关系进行转换后再输出; 若入参为其他字符类型, 则会将该字符编码视作非法字符, 以占位符的形式输出。</p>

表 7-215 UTL\_RAW 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	BIT_AND Function	BIT_AND Function	-
2	BIT_COMPLEMENT Function	BIT_COMPLEMENT Function	-

序号	Oracle数据库	GaussDB数据库	差异
3	BIT_OR Function	BIT_OR Function	GaussDB: 两个参数类型被定义为TEXT类型并且返回TEXT类型。 Oracle: 两个参数为RAW类型并且返回RAW类型。
4	BIT_XOR Function	BIT_XOR Function	-
5	CAST_FROM_BINARY_DOUBLE Function	CAST_FROM_BINARY_DOUBLE_TO_RAW Function	-
6	CAST_FROM_BINARY_FLOAT Function	CAST_FROM_BINARY_FLOAT_TO_RAW Function	GaussDB: 参数n为FLOAT4类型。 Oracle: 参数n为FLOAT类型。
7	CAST_FROM_BINARY_INTEGER Function	CAST_FROM_BINARY_INTEGER_TO_RAW Function	GaussDB: 参数value为BIGINT类型。 Oracle: 参数value为INTEGER类型。
8	CAST_FROM_NUMBER Function	CAST_FROM_NUMBER_TO_RAW Function	GaussDB: 参数n为NUMERIC类型 Oracle: 参数n为NUMBER类型。
9	CAST_TO_BINARY_DOUBLE Function	CAST_FROM_RAW_TO_BINARY_DOUBLE Function	-
10	CAST_TO_BINARY_FLOAT Function	CAST_FROM_RAW_TO_BINARY_FLOAT Function	GaussDB: 函数返回类型为FLOAT4。 Oracle: 函数返回类型为FLOAT。

序号	Oracle数据库	GaussDB数据库	差异
11	CAST_TO_BINARY_INTEGER Function	CAST_FROM_RAW_TO_BINARY_INTEGER Function	GaussDB: 参数 endianness为 INTEGER类型, 函数返回类型为 INTEGER。 Oracle: 参数 endianness为 PLS_INTEGER类型, 函数返回类型为 BINARY_INTEGER。
12	CAST_TO_NUMBER Function	CAST_FROM_RAW_TO_NUMBER Function	GaussDB: 函数返回类型为 NUMERIC。 Oracle: 函数返回类型为 NUMBER。
13	CAST_TO_NVARCHAR2 Function	CAST_FROM_RAW_TO_NVARCHAR2 Function	-
14	CAST_TO_RAW Function	CAST_FROM_VARCHAR2_TO_RAW Function	-
15	CAST_TO_VARCHAR2 Function	CAST_TO_VARCHAR2 Function	-
16	COMPARE Function	COMPARE Function	GaussDB: 函数返回类型为 INTEGER。 Oracle: 函数返回类型为 NUMBER。
17	CONCAT Function	CONCAT Function	-
18	CONVERT Function	CONVERT Function	-
19	COPIES Function	COPIES Function	GaussDB: 参数n为 NUMERIC类型。 Oracle: 参数n为 NUMBER类型。
20	LENGTH Function	GET_LENGTH Function	GaussDB: 函数返回类型为 INTEGER。 Oracle: 函数返回类型为 NUMBER。

序号	Oracle数据库	GaussDB数据库	差异
21	OVERLAY Function	OVERLAY Function	-
22	REVERSE Function	REVERSE Function	-
23	SUBSTR Function	SUBSTR Function	GaussDB: 参数 lob_loc为BLOB类型; 参数off_set为INTEGER类型, 默认值为1; 参数 amount为INTEGER类型, 默认值为32767。 Oracle: 参数r为RAW类型, 参数 pos为BINARY_INTEGER类型且无默认值, 参数len为BINARY_INTEGER类型, 默认值为NULL。
24	TRANSLATE Function	TRANSLATE Function	-
25	TRANSLITERATE Function	TRANSLITERATE Function	-
26	XRANGE Function	XRANGE Function	GaussDB: 参数 start_byte和 end_byte无默认值。 Oracle: 参数 start_byte和 end_byte默认为NULL。

表 7-216 DBMS\_SCHEDULER 兼容性说明

序号	Oracle数据库	GaussDB数据库
1	ADD_EVENT_QUEUE_SU BSCRIBER Procedure	不支持
2	ADD_GROUP_MEMBER Procedure	不支持
3	ADD_JOB_EMAIL_NOTIFI CATION Procedure	不支持



序号	Oracle数据库	GaussDB数据库
4	ADD_TO_INCOMPATIBILITY Procedure	不支持
5	ALTER_CHAIN Procedure	不支持
6	ALTER_RUNNING_CHAIN Procedure	不支持
7	CLOSE_WINDOW Procedure	不支持
8	COPY_JOB Procedure	不支持
9	CREATE_CHAIN Procedure	不支持
10	CREATE_CREDENTIAL Procedure	CREATE_CREDENTIAL Procedure
11	CREATE_DATABASE_DESTINATION Procedure	不支持
12	CREATE_EVENT_SCHEDULE Procedure	不支持
13	CREATE_FILE_WATCHER Procedure	不支持
14	CREATE_GROUP Procedure	不支持
15	CREATE_INCOMPATIBILITY Procedure	不支持
16	CREATE_JOB Procedure	CREATE_JOB Procedure
17	CREATE_JOB_CLASS Procedure	CREATE_JOB_CLASS Procedure
18	CREATE_JOBS Procedure	不支持
19	CREATE_PROGRAM Procedure	CREATE_PROGRAM Procedure
20	CREATE_RESOURCE Procedure	不支持
21	CREATE_SCHEDULE Procedure	CREATE_SCHEDULE Procedure
22	CREATE_WINDOW Procedure	不支持
23	DEFINE_ANYDATA_ARGUMENT Procedure	不支持

序号	Oracle数据库	GaussDB数据库
24	DEFINE_CHAIN_EVENT_STEP Procedure	不支持
25	DEFINE_CHAIN_RULE Procedure	不支持
26	DEFINE_CHAIN_STEP Procedure	不支持
27	DEFINE_METADATA_ARGUMENT Procedure	不支持
28	DEFINE_PROGRAM_ARGUMENT Procedure	DEFINE_PROGRAM_ARGUMENT Procedure
29	DISABLE Procedure	DISABLE Procedure
30	DROP_AGENT_DESTINATION Procedure	不支持
31	DROP_CHAIN Procedure	不支持
32	DROP_CHAIN_RULE Procedure	不支持
33	DROP_CHAIN_STEP Procedure	不支持
34	DROP_CREDENTIAL Procedure	DROP_CREDENTIAL Procedure
35	DROP_DATABASE_DESTINATION Procedure	不支持
36	DROP_FILE_WATCHER Procedure	不支持
37	DROP_GROUP Procedure	不支持
38	DROP_INCOMPATIBILITY Procedure	不支持
39	DROP_JOB Procedure	DROP_JOB Procedure
40	DROP_JOB_CLASS Procedure	DROP_JOB_CLASS Procedure
41	DROP_PROGRAM Procedure	DROP_PROGRAM Procedure
42	DROP_PROGRAM_ARGUMENT Procedure	不支持
43	DROP_SCHEDULE Procedure	DROP_SCHEDULE Procedure

序号	Oracle数据库	GaussDB数据库
44	DROP_WINDOW Procedure	不支持
45	ENABLE Procedure	ENABLE Procedure
46	END_DETACHED_JOB_RUN Procedure	不支持
47	EVALUATE_CALENDAR_STRING Procedure	EVALUATE_CALENDAR_STRING Procedure
48	EVALUATE_RUNNING_CHAIN Procedure	不支持
49	GENERATE_JOB_NAME Function	GENERATE_JOB_NAME Function
50	GET_AGENT_INFO Function	不支持
51	GET_AGENT_VERSION Function	不支持
52	GET_ATTRIBUTE Procedure	不支持
53	GET_FILE Procedure	不支持
54	GET_SCHEDULER_ATTRIBUTE Procedure	不支持
55	OPEN_WINDOW Procedure	不支持
56	PURGE_LOG Procedure	不支持
57	PUT_FILE Procedure	不支持
58	REMOVE_EVENT_QUEUE_SUBSCRIBER Procedure	不支持
59	REMOVE_FROM_INCOMPATIBILITY Procedure	不支持
60	REMOVE_GROUP_MEMBER Procedure	不支持
61	REMOVE_JOB_EMAIL_NOTIFICATION Procedure	不支持
62	RESET_JOB_ARGUMENT_VALUE Procedure	不支持
63	RUN_CHAIN Procedure	不支持
64	RUN_JOB Procedure	RUN_JOB Procedure

序号	Oracle数据库	GaussDB数据库
65	SET_AGENT_REGISTRATION_PASS Procedure	不支持
66	SET_ATTRIBUTE Procedure	SET_ATTRIBUTE Procedure
67	SET_ATTRIBUTE_NULL Procedure	不支持
68	SET_JOB_ANYDATA_VALUE Procedure	不支持
69	SET_JOB_ARGUMENT_VALUE Procedure	SET_JOB_ARGUMENT_VALUE Procedure
70	SET_JOB_ATTRIBUTES Procedure	不支持
71	SET_RESOURCE_CONSTRAINT Procedure	不支持
72	SET_SCHEDULER_ATTRIBUTE Procedure	不支持
73	STOP_JOB Procedure	STOP_JOB Procedure

表 7-217 DBMS\_UTILITY 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	ACTIVE_INSTANCES Procedure	不支持	-
2	ANALYZE_DATABASE Procedure	不支持	-
3	ANALYZE_PART_OBJECT Procedure	不支持	-
4	ANALYZE_SCHEMA Procedure	不支持	-
5	CANONICALIZE Procedure	CANONICALIZE Procedure	GaussDB: 参数 canon_len默认为1024字节。 Oracle: 参数 canon_len无默认值。

序号	Oracle数据库	GaussDB数据库	差异
6	COMMA_TO_TABLE Procedures	COMMA_TO_TABLE Procedure	GaussDB: 参数 tab 为 VARCHAR2 数组。 Oracle: 该过程存在 2 个重载。参数 tab 可以为两种类型之一: 一种为 uncl_array, 另一种为 lname_array。
7	COMPILE_SCHEMA Procedure	不支持	-
8	CREATE_ALTER_TYPE_ERROR_TABLE Procedure	不支持	-
9	CURRENT_INSTANCE Function	不支持	-
10	DATA_BLOCK_ADDRESS_BLOCK Function	不支持	-
11	DATA_BLOCK_ADDRESS_FILE Function	不支持	-
12	DB_VERSION Procedure	DB_VERSION Procedure	GaussDB: 只有参数 version, 类型为 VARCHAR2。 Oracle: 有参数 version 和 compatibility, 类型均为 VARCHAR2。
13	EXEC_DDL_STATEMENT Procedure	EXEC_DDL_STATEMENT Function	GaussDB: 参数 parse_string 为 TEXT 类型。 Oracle: 参数 parse_string 为 VARCHAR2 类型。

序号	Oracle数据库	GaussDB数据库	差异
14	EXPAND_SQL_TEXT Procedure	EXPAND_SQL_TEXT Function	GaussDB: 参数 output_sql_text 为 CLOB。 Oracle: 参数 output_sql_text 为 NOCOPY CLOB, 通过传引用方式传递 OUT 参数。
15	FORMAT_CALL_STACK Function	FORMAT_CALL_STACK Function	GaussDB: 函数返回类型为 TEXT。 Oracle: 函数返回类型为 VARCHAR2。
16	FORMAT_ERROR_BACKTRACE Function	FORMAT_ERROR_BACKTRACE Function	GaussDB: 函数返回类型为 TEXT。 Oracle: 函数返回类型为 VARCHAR2。
17	FORMAT_ERROR_STACK Function	FORMAT_ERROR_STACK Function	GaussDB: 函数返回类型为 TEXT。 Oracle: 函数返回类型为 VARCHAR2。
18	GET_CPU_TIME Function	GET_CPU_TIME Function	GaussDB: 函数返回类型为 BIGINT。 Oracle: 函数返回类型为 NUMBER。
19	GET_DEPENDENCY Procedure	不支持	-
20	GET_ENDIANNESS Function	GET_ENDIANNESS Function	GaussDB: 函数返回类型为 INTEGER。 Oracle: 函数返回类型为 NUMBER。
21	GET_HASH_VALUE Function	GET_HASH_VALUE Function	GaussDB: 参数 base、hash_size 和返回类型均为 INTEGER。 Oracle: 参数 base、hash_size 和返回类型均为 NUMBER。

序号	Oracle数据库	GaussDB数据库	差异
22	GET_PARAMETER_VALUE Function	不支持	-
23	GET_SQL_HASH Function	GET_SQL_HASH Function	GaussDB: 参数 last4bytes 为 BIGINT类型, 代表 MD5哈希值的最后四字节, 以无符号整数形式展现, 函数返回类型为 BIGINT。 Oracle: 对应参数 pre10ihash为 NUMBER类型, 用于存储MD5计算得到的16字节中的4字节哈希值。
24	GET_TIME Function	GET_TIME Function	GaussDB: 函数返回类型为BIGINT。 Oracle: 函数返回类型为NUMBER。
25	GET_TZ_TRANSITIONS Procedure	不支持	-
26	INVALIDATE Procedure	不支持	-
27	IS_BIT_SET Function	IS_BIT_SET Function	GaussDB: 参数n和返回值类型为 INTEGER。 Oracle: 参数n和返回值类型为 NUMBER。
28	IS_CLUSTER_DATA_BASE Function	IS_CLUSTER_DATA_BASE Function	-
29	MAKE_DATA_BLOCK_ADDRESS Function	不支持	-

序号	Oracle数据库	GaussDB数据库	差异
30	NAME_RESOLVE Procedure	NAME_RESOLVE Procedure	GaussDB: 参数 context和 part1_type为 INTEGER, 参数 object_number为 OID; GaussDB不支持NUMBER到OID的隐式转换。 Oracle: 参数 context、 part1_type和 object_number均为NUMBER。
31	NAME_TOKENIZE Procedure	NAME_TOKENIZE Procedure	GaussDB: 参数 nextpos为 INTEGER类型。 Oracle: 参数 nextpos为 BINARY_INTEGER类型。
32	OLD_CURRENT_SCHEMA Function	OLD_CURRENT_SCHEMA Function	GaussDB: 函数返回类型为 VARCHAR。 Oracle: 函数返回类型为 VARCHAR2。
33	OLD_CURRENT_USER Function	OLD_CURRENT_USER Function	GaussDB: 函数返回类型为TEXT。 Oracle: 函数返回类型为 VARCHAR2。
34	PORT_STRING Function	不支持	-
35	SQLID_TO_SQLHASH Function	不支持	-
36	TABLE_TO_COMMA Procedures	TABLE_TO_COMMA Procedure	GaussDB: 参数 tab为VARCHAR2数组。 Oracle: 该存储过程存在2个重载。参数tab可以为两种类型之一: 一种为uncl_array, 另一种为 lname_array。



序号	Oracle数据库	GaussDB数据库	差异
37	VALIDATE Procedure	不支持	-
38	WAIT_ON_PENDING_DML Function	不支持	-

表 7-218 DBMS\_SQL 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	BIND_ARRAY Procedures	SQL_BIND_ARRAY Function	-
2	BIND_VARIABLE Procedures	SQL_BIND_VARIABLE Function	-
3	BIND_VARIABLE_PACKAGE Procedure	不支持	-
4	CLOSE_CURSOR Procedure	SQL_UNREGISTER_CONTEXT Function	-
5	COLUMN_VALUE Procedure	GET_RESULT Procedure	-
6	COLUMN_VALUE_LONG Procedure	不支持	-
7	DEFINE_ARRAY Procedure	SET_RESULTS_TYPE Procedure	-
8	DEFINE_COLUMN Procedures	SET_RESULT_TYPE Procedure	-
9	DEFINE_COLUMN_CHAR Procedure	不支持	-
10	DEFINE_COLUMN_LONG Procedure	不支持	-
11	DEFINE_COLUMN_RAW Procedure	不支持	-
12	DEFINE_COLUMN_ROWID Procedure	不支持	-
13	DESCRIBE_COLUMNS Procedure	DESCRIBE_COLUMNS Procedure	-
14	DESCRIBE_COLUMNS2 Procedure	不支持	-

序号	Oracle数据库	GaussDB数据库	差异
15	DESCRIBE_COLU MNS3 Procedure	不支持	-
16	EXECUTE Function	SQL_RUN Function	-
17	EXECUTE_AND_FE TCH Function	RUN_AND_NEXT Function	-
18	FETCH_ROWS Function	NEXT_ROW Function	-
19	GET_NEXT_RESUL T Procedures	不支持	-
20	IS_OPEN Function	IS_ACTIVE Function	-
21	LAST_ERROR_POS ITION Function	不支持	-
22	LAST_ROW_COU NT Function	LAST_ROW_COU NT Function	-
23	LAST_ROW_ID Function	不支持	-
24	LAST_SQL_FUNCT ION_CODE Function	不支持	-
25	OPEN_CURSOR Functions	REGISTER_CONTE XT Function	-
26	PARSE Procedures	支持，有差异。	GaussDB中为 SQL_SET_SQL Function，不支持 重载。
27	RETURN_RESULT Procedures	不支持	-
28	TO_CURSOR_NU MBER Function	不支持	-
29	TO_REFCURSOR Function	不支持	-
30	VARIABLE_VALUE Procedures	GET_VARIABLE_RE SULT Procedures	-
31	VARIABLE_VALUE_ PKG Procedure	不支持	-

表 7-219 DBMS\_SQL 数据类型兼容性说明

序号	Oracle数据库	GaussDB数据库
1	DBMS_SQL DESC_REC	DBE_SQL.DESC_REC
2	DBMS_SQL DATE_TABLE	DBE_SQL.DATE_TABLE
3	DBMS_SQL NUMBER_TABLE	DBE_SQL.NUMBER_TABLE
4	DBMS_SQL VARCHAR2_TABLE	DBE_SQL.VARCHAR2_TABLE
5	DBMS_SQL BLOB_TABLE	DBE_SQL.BLOB_TABLE

表 7-220 UTL\_FILE 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	FCLOSE Procedure	CLOSE Procedure	-
2	FCLOSE_ALL Procedure	CLOSE_ALL Procedure	-
3	FCOPY Procedure	COPY Procedure	-
4	FFLUSH Procedure	FLUSH Procedure	-
5	FGETATTR Procedure	GET_ATTR Procedure	-
6	FGETPOS Function	GET_POS Function	-
7	FOPEN Function	FOPEN Function	-
8	FOPEN_NCHAR Function	FOPEN_NCHAR Function	-
9	FREMOVE Procedure	REMOVE Procedure	-
10	FRENAME Procedure	RENAME Procedure	-
11	FSEEK Procedure	SEEK Procedure	-
12	GET_LINE Procedure	READ_LINE Procedure	-
13	GET_LINE_NCHAR Procedure	READ_LINE_NCHAR Procedure	-
14	GET_RAW Procedure	GET_RAW Procedure	-

序号	Oracle数据库	GaussDB数据库	差异
15	IS_OPEN Function	IS_OPEN Function	-
16	NEW_LINE Procedure	支持，有差异，NEW_LINE Function	GaussDB因性能原因将接口定义为Function。
17	PUT Procedure	支持，有差异，WRITE Function	GaussDB因性能原因将接口定义为Function。
18	PUT_LINE Procedure	支持，有差异，WRITE_LINE Function	GaussDB因性能原因将接口定义为Function。
19	PUT_LINE_NCHAR Procedure	支持，有差异，WRITE_LINE_NCHAR Function	GaussDB因性能原因将接口定义为Function。
20	PUT_NCHAR Procedure	支持，有差异，WRITE_NCHAR Function	GaussDB因性能原因将接口定义为Function。
21	PUTF Procedure	支持，有差异，FORMAT_WRITE Function	GaussDB因性能原因将接口定义为Function。
22	PUTF_NCHAR Procedure	支持，有差异，FORMAT_WRITE_NCHAR Function	GaussDB因性能原因将接口定义为Function。
23	PUT_RAW Procedure	支持，有差异，PUT_RAW Function	GaussDB因性能原因将接口定义为Function。

表 7-221 DBMS\_SESSION 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	CLEAR_ALL_CONTAINER Procedure	不支持	-
2	CLEAR_CONTEXT Procedure	CLEAR_CONTEXT Function	-
3	CLEAR_IDENTIFIER Procedure	不支持	-
4	CLOSE_DATABASE_LINK Procedure	不支持	-

序号	Oracle数据库	GaussDB数据库	差异
5	CURRENT_IS_ROLE_ENABLED Function	不支持	-
6	FREE_UNUSED_USER_MEMORY Procedure	不支持	-
7	GET_PACKAGE_MEMORY_UTILIZATION Procedure	不支持	-
8	IS_ROLE_ENABLED Function	不支持	-
9	IS_SESSION_ALIVE Function	不支持	-
10	LIST_CONTEXT Procedures	不支持	-
11	MODIFY_PACKAGE_STATE Procedure	MODIFY_PACKAGE_STATE Procedure	GaussDB: 仅支持 入参flags = 1的场景使用。 Oracle: 支持 flags=1或flags= 2 的场景使用。
12	RESET_PACKAGE Procedure	不支持	-
13	SESSION_IS_ROLE_ENABLED Function	不支持	-
14	SESSION_TRACE_DISABLE Procedure	不支持	-
15	SESSION_TRACE_ENABLE Procedure	不支持	-
16	SET_CONTEXT Procedure	SET_CONTEXT Function	GaussDB: 仅包括 参数namespace, attribute和value, 类型均为text。 Oracle: 包括参数 namespace, attribute, value, username和 client_id, 类型均 为VARCHAR2。

序号	Oracle数据库	GaussDB数据库	差异
17	SET_EDITION_DEFERRED Procedure	不支持	-
18	SET_IDENTIFIER Procedure	不支持	-
19	SET-NLS Procedure	不支持	-
20	SET_ROLE Procedure	不支持	-
21	SET_SQL_TRACE Procedure	不支持	-
22	SLEEP Procedure	不支持	-
23	SWITCH_CURRENT_CONSUMER_GROUP Procedure	不支持	--
24	UNIQUE_SESSION_ID Function	不支持	-

表 7-222 UTL\_MATCH 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	EDIT_DISTANCE Function	不支持	-
2	EDIT_DISTANCE_SIMILARITY Function	EDIT_DISTANCE_SIMILARITY Function	GaussDB: 参数 str1和str2均为 TEXT类型, 函数返回类型为 INTEGER。 Oracle: 参数s1和s2为VARCHAR2类型, 函数返回类型为 PLS_INTEGER。
3	JARO_WINKLER Function	不支持	-
4	JARO_WINKLER_SIMILARITY Function	不支持	-

表 7-223 DBMS\_APPLICATION\_INFO 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	READ_CLIENT_INFO Procedure	READ_CLIENT_INFO Function	GaussDB: 参数 client_info为TEXT类型。 Oracle: 参数 client_info为VARCHAR2类型。
2	READ_MODULE Procedure	READ_MODULE Procedure	GaussDB: 参数 module_name、action_name为TEXT类型。 Oracle: 参数 module_name、action_name为VARCHAR2类型。
3	SET_ACTION Procedure	SET_ACTION Procedure	GaussDB: 参数 action_name为TEXT类型。 Oracle: 参数 action_name为VARCHAR2类型。
4	SET_CLIENT_INFO Procedure	SET_CLIENT_INFO Function	GaussDB: 参数str为TEXT类型, 且返回类型为void。 Oracle: 参数 client_info为VARCHAR2类型, 无返回值。二者均为写入客户端信息, 最大输入64字节, 超过64字节将被截断。
5	SET_MODULE Procedure	SET_MODULE Procedure	GaussDB: 参数 module_name、action_name为TEXT类型。 Oracle: 参数 module_name、action_name为VARCHAR2类型。
6	SET_SESSION_LONGOPS Procedure	不支持	-

表 7-224 DBMS\_XMLDOM 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	DBMS_XMLDOM.APP ENDCHILD	DBE_XMLDOM.A PPENDCHILD	<ul style="list-style-type: none"> <li>GaussDB: DOCUMENT类型节点下APPEND ATTR类型节点会报“operation not support”错误。 Oracle: 在此场景下不报错, 但实际并没有挂载成功。</li> <li>GaussDB: ATTR类型节点下APPEND ATTR类型节点会报“operation not support”错误。 Oracle: 在此场景下不报错, 但实际并没有挂载成功。</li> <li>GaussDB: 父节点在添加多个ATTR类型子节点时, 不允许KEY值相同的子节点同时存在于同一个父节点下。 Oracle: 允许KEY值相同的子节点同时存在于同一个父节点下。</li> </ul>
2	DBMS_XMLDOM.CRE ATEELEMENT	DBE_XMLDOM.C REATEELEMENT	-
3	DBMS_XMLDOM.CRE ATETEXTNODE	DBE_XMLDOM.C REATETEXTNOD E	-
4	DBMS_XMLDOM.FRE EDOCUMENT	DBE_XMLDOM.F REEDOCUMENT	GaussDB: 释放时不会立刻释放对象, 累积一定数量后释放。 document下全部节点失效。 Oracle: 立即释放对象。
5	DBMS_XMLDOM.FRE EELEMENT	DBE_XMLDOM.F REELEMENT	-
6	DBMS_XMLDOM.FRE ENODE	DBE_XMLDOM.F REENODE	-
7	DBMS_XMLDOM.FRE ENODELIST	DBE_XMLDOM.F REENODELIST	GaussDB: nodelist会被释放。 Oracle: 释放nodelist后, 在原始的doc中还能被查询到。
8	DBMS_XMLDOM.GET ATTRIBUTE	DBE_XMLDOM.G ETATTRIBUTE	-
9	DBMS_XMLDOM.GET ATTRIBUTES	DBE_XMLDOM.G ETATTRIBUTES	-



序号	Oracle数据库	GaussDB数据库	差异
10	DBMS_XMLDOM.GETCHILDNODES	DBE_XMLDOM.GETCHILDNODES	GaussDB: 对document的node使用时会包含dtd。 Oracle: 不包含dtd。
11	DBMS_XMLDOM.GETCHILDRENBYTAGNAME	DBE_XMLDOM.GETCHILDRENBYTAGNAME	GaussDB: DBE_XMLDOM.GETCHILDRENBYTAGNAME接口的参数ns不支持传入参数" * "，如需获取节点下全部属性，可使用DBE_XMLDOM.GETCHILDNODES接口。 Oracle: 支持传入参数" * "。
12	DBMS_XMLDOM.GETDOCUMENTELEMENT	DBE.XMLDOM.GETDOCUMENTELEMENT	-
13	DBMS_XMLDOM.GETFIRSTCHILD	DBE_XMLDOM.GETFIRSTCHILD	-
14	DBMS_XMLDOM.GETLASTCHILD	DBE_XMLDOM.GETLASTCHILD	-
15	DBMS_XMLDOM.GETLENGTH	DBE_XMLDOM.GETLENGTH	-
16	DBMS_XMLDOM.GETLOCALNAME	DBE_XMLDOM.GETLOCALNAME	-
17	DBMS_XMLDOM.GETNAMEDITEM	DBE_XMLDOM.GETNAMEDITEM	-
18	DBMS_XMLDOM.GETNEXTSIBLING	DBE_XMLDOM.GETNEXTSIBLING	-
19	DBMS_XMLDOM.GETNODENAME	DBE_XMLDOM.GETNODENAME	-
20	DBMS_XMLDOM.GETNODETYPE	DBE_XMLDOM.GETNODETYPE	-
21	DBMS_XMLDOM.GETTAGNAME	DBE_XMLDOM.GETTAGNAME	-
22	DBMS_XMLDOM.IMPORTNODE	DBE_XMLDOM.IMPORTNODE	-
23	DBMS_XMLDOM.ISNULL	DBE_XMLDOM.ISNULL	GaussDB: 入参为DOMNODELIST类型时，若对象在哈希表中不存在会发生报错。 Oracle: 不会报错。

序号	Oracle数据库	GaussDB数据库	差异
24	DBMS_XMLDOM.ITEM	DBE_XMLDOM.ITEM	-
25	DBMS_XMLDOM.MAKENODE	DBE_XMLDOM.MAKENODE	GaussDB: 该函数不支持直接作为函数返回值返回。 Oracle: 支持直接作为函数返回值返回。

序号	Oracle数据库	GaussDB数据库	差异
26	DBMS_XMLDOM.NEWDOMDOCUMENT	DBE_XMLDOM.NEWDOMDOCUMENT	<ul style="list-style-type: none"> <li>GaussDB集中式入参大小需限制在2GB以内。 Oracle: 与CLOB类型大小一致。</li> <li>GaussDB目前暂不支持外部DTD解析。 Oracle: 支持解析外部DTD。</li> <li>GaussDB newdomdocument创建的doc, 默认UTF-8字符集。 Oracle: 根据服务端字符集生成。</li> <li>GaussDB从同一个xmltype实例中解析出的每一个doc都是独立的, 对doc的修改也不会影响到xmltype。 Oracle: 从同一个xmltype实例中解析出的每一个doc不独立, 有关联关系。</li> <li>GaussDB version字段只支持1.0, 1.0-1.9解析警告但正常执行, 1.9以上报错。 Oracle: 不报错。</li> <li>GaussDB与Oracle数据库DTD校验差异: !ATTLIST to type (CHECK check Check) "Ch..."将报错, 因默认值"Ch..."不属于括号中枚举值, 而Oracle数据库不报错。&lt;!ENTITY baidu "www.baidu.com"&gt;..... &amp;Baidu;&amp;writer将报错, 因区分字母大小写, Baidu无法与baidu对应。 Oracle: 不报错。</li> <li>GaussDB 与Oracle数据库命名空间校验差异: 解析未声明的命名空间标签正常执行。 Oracle: 报错。</li> </ul>
27	DBMS_XMLDOM.SETATTRIBUTE	DBE_XMLDOM.SETATTRIBUTE	<p>GaussDB: 属性key不支持为null或空字符串。 Oracle: 属性key允许为null或空字符串。</p>

序号	Oracle数据库	GaussDB数据库	差异
28	DBMS_XMLDOM.SET CHARSET	DBE_XMLDOM.S ETCHARSET	GaussDB目前支持的字符集有： UTF-8、UTF-16、UCS-4、 UCS-2、ISO-8859-1、 ISO-8859-2、ISO-8859-3、 ISO-8859-4、ISO-8859-5、 ISO-8859-6、ISO-8859-7、 ISO-8859-8、ISO-8859-9、 ISO-2022-JP、Shift_JIS、EUC- JP、ASCII。输入其他字符集会 报错或者可能导致输出乱码。
29	DBMS_XMLDOM.SET DOCTYPE	DBE_XMLDOM.S ETDOCTYPE	GaussDB name、sysid、pubid 的总长度限制在32500个字节以 内。 Oracle：限制在32767字节内。
30	DBMS_XMLDOM.WRI TETOBUFFER	DBE_XMLDOM. WRITETOBUFFER	<ul style="list-style-type: none"> <li>• GaussDB writetobuffer输出 buffer限制在1GB以内。 Oracle：限制在32767字节 内。</li> <li>• GaussDB输出doc将包含XML 声明version和encoding。 Oracle：用户不主动指定将 不包含。</li> <li>• GaussDB入参为domnode类 型时，如果节点是doc转换 的，输出节点将包含XML声 明version和encoding。 Oracle：用户不主动指定将 不包含。</li> <li>• GaussDB默认以UTF-8字符 集输出xml。 Oracle：根据数据库字符集 生成。</li> </ul>

序号	Oracle数据库	GaussDB数据库	差异
31	DBMS_XMLDOM.WRITETOCLOB	DBE_XMLDOM.WRITETOCLOB	<ul style="list-style-type: none"> <li>• GaussDB writetoclob大小支持1GB以内。 Oracle: 按CLOB大小支持。</li> <li>• GaussDB输出doc将包含XML声明version和encoding。 Oracle: 用户不主动指定将不包含。</li> <li>• GaussDB入参为domnode类型时, 如果节点是doc转换的, 输出节点将包含XML声明version和encoding。 Oracle: 用户不主动指定将不包含。</li> <li>• GaussDB 默认以UTF-8字符集输出xml。 Oracle: 根据数据库字符集生成。</li> </ul>

序号	Oracle数据库	GaussDB数据库	差异
32	DBMS_XMLDOM.WRITETOFILE	DBE_XMLDOM.WRITETOFILE	<ul style="list-style-type: none"> <li>• GaussDB document入参，filename长度限制在255个字节以内，charset请参考dbe_xmldom.setcharset接口。 Oracle: filename长度限制受操作系统影响，大于255个字节。</li> <li>• GaussDB domnode入参，filename长度限制在255个字节以内，charset请参考dbe_xmldom.setcharset接口。 Oracle: filename长度限制受操作系统影响，大于255个字节。</li> <li>• GaussDB该函数会添加缩进等内容，将输出格式化。输出doc将包含XML声明version和encoding。入参为domnode类型时，如果节点是doc转换的，输出节点将包含XML声明version和encoding。 Oracle: 用户不主动指定将不包含。</li> <li>• GaussDB传入newdomdocument()无参创建的doc，在不指定charset时不会报错，默认UTF-8字符集。 Oracle: 会进行报错。</li> <li>• GaussDB filename需要在pg_directory中创建的路径下，filename中的\会被转换成/，只允许存在一个/。文件名格式应为pg_directory_name/file_name。 Oracle: 按用户输入不进行转义。</li> </ul>
33	DBMS_XMLDOM.GETNODEVALUE	DBE_XMLDOM.GETNODEVALUE	-
34	DBMS_XMLDOM.GETPARENTNODE	DBE_XMLDOM.GETPARENTNODE	-

序号	Oracle数据库	GaussDB数据库	差异
35	DBMS_XMLDOM.HAS CHILDNODES	DBE_XMLDOM.H ASCHILDNODES	-
36	DBMS_XMLDOM.MA KKELEMENT	DBE_XMLDOM. MAKEELEMENT	-
37	DBMS_XMLDOM.SET NODEVALUE	DBE_XMLDOM.S ETNODEVALUE	<ul style="list-style-type: none"> <li>GaussDB nodeValue入参，可以输入空字符串和NULL值，但不会对节点值进行修改。 Oracle: 空字符串和NULL会将节点值修改为空字符串。</li> <li>GaussDB nodeValue入参，暂不支持转义字符'&amp;'，如字符串中包含该转义字符，会清空节点值。 Oracle: 支持转义字符。</li> </ul>
38	DBMS_XMLDOM.GET ELEMENTSBYTAGNA ME	DBE_XMLDOM.G ETELEMENTSBY AGNAME	-

表 7-225 DBMS\_XMLPARSER 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	DBMS_XMLPARSE R.FREEPARSER	DBE_XMLPARSER. FREEPARSER	-
2	DBMS_XMLPARSE R.GETDOCUMENT	DBE_XMLPARSER. GETDOCUMENT	-
3	DBMS_XMLPARSE R.GETVALIDATIO NMODE	DBE_XMLPARSER. GETVALIDATION MODE	-
4	DBMS_XMLPARSE R.NEWPARSER	支持，有差异， DBE_XMLPARSER. NEWPARSER	GaussDB中parser 对象的数量上限为 16777215， Oracle数据库中约 为1亿。

序号	Oracle数据库	GaussDB数据库	差异
5	DBMS_XMLPARSE R.PARSEBUFFER	支持，有差异， DBE_XMLPARSER. PARSEBUFFER	<p>1. 与Oracle数据库解析字段差异： 字符串encoding只支持UTF-8； version字段只支持1.0，1.0-1.9解析警告但正常执行，1.9以上报错。</p> <p>2. 与Oracle数据库命名空间校验差异：解析未声明的命名空间标签正常执行，而Oracle数据库会报错。</p> <p>3. 与Oracle数据库xml预定义实体解析差异： '&amp;quot;'会被解析转译为字符' ”'，而Oracle数据库中预定义实体统一都没有转译为字符。</p> <p>4. 与Oracle数据库DTD校验差异：</p> <ul style="list-style-type: none"> <li>!ATTLIST to type (CHECK check Check) "Ch..."将报错，因默认值"Ch..."不属于括号中枚举值，而Oracle数据库不报错。</li> <li>&lt;!ENTITY baidu "www.baidu.com"&gt;..... &amp;Baidu;&amp;writer将报错，因区分字母大小写，Baidu无法与baidu对应，而Oracle数据库不报错。</li> </ul>



序号	Oracle数据库	GaussDB数据库	差异
6	DBMS_XMLPARSER.PARSECLOB	支持，有差异，DBE_XMLPARSER.PARSECLOB	<p>1. PARSECLOB不支持解析大于等于2GB的clob。</p> <p>2. 与Oracle数据库解析字段差异： 字符串encoding只支持UTF-8； version字段只支持1.0，1.0-1.9解析警告但正常执行，1.9以上报错。</p> <p>3. 与Oracle数据库命名空间校验差异：解析未声明的命名空间标签正常执行，而Oracle数据库会报错。</p> <p>4. 与Oracle数据库xml预定义实体解析差异： '&amp;quot;'会被解析转译为字符' ” ，而Oracle数据库预定义实体统一都没有转译为字符。</p> <p>5. 与Oracle数据库DTD校验差异：</p> <ul style="list-style-type: none"> <li>!ATTLIST to type (CHECK check Check) "Ch..."将报错，因默认值"Ch..."不属于括号中枚举值，而Oracle数据库不报错。</li> <li>&lt;!ENTITY baidu "www.baidu.com"&gt;..... &amp;Baidu;&amp;writer将报错，因区分字母大小写，Baidu无法与baidu对应，而Oracle数据库不报错。</li> </ul>

序号	Oracle数据库	GaussDB数据库	差异
7	DBMS_XMLPARSER.SETVALIDATIONMODE	DBE_XMLPARSER.SETVALIDATIONMODE	-

表 7-226 DBMS\_ILM 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	DBMS_ILM.ADD_T O_ILM	不支持	-
2	DBMS_ILM.ARCHI VESTATENAME	不支持	-
3	DBMS_ILM.EXECU TE_ILM	支持，有差异， DBE_ILM.EXECUTE _ILM	<ul style="list-style-type: none"> <li>• GaussDB数据库的入参 schema 在 Oracle 数据库 中对应为 owner。</li> <li>• GaussDB数据库不支持指定 ilm_scope（一次指定多个对象）的操作。</li> </ul>
4	DBMS_ILM.EXECU TE_ILM_TASK	不支持	-
5	DBMS_ILM.PREVI EW_ILM	不支持	-
6	DBMS_ILM.REMO VE_FROM_ILM	不支持	-
7	DBMS_ILM.STOP_ ILM	DBE_ILM.STOP_IL M	-

表 7-227 DBMS\_ILM\_ADMIN 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	DBMS_ILM_ADMIN.CLEAR_HEAT_M AP_ALL	不支持	-
2	DBMS_ILM_ADMIN.CLEAR_HEAT_M AP_TABLE	不支持	-

序号	Oracle数据库	GaussDB数据库	差异
3	DBMS_ILM_ADMIN.CUSTOMIZE_ILM	支持，有差异，DBE_ILM_ADMIN.CUSTOMIZE_ILM	入参parameter取值对应的特性参数存在差异。GaussDB数据库param取值支持1、2、7、11、12、13、14和15。GaussDB数据库param取值为14时，对应的特性参数为WIND_DURATION，用于控制自动调度中执行窗口的持续时长，而ORACLE数据库对应的特性参数则为AUTO_OPTIMIZE_INACTIVITY_THRESHOLD，其表示ado的不活动时间长度。
4	DBMS_ILM_ADMIN.DISABLE_ILM	DBE_ILM_ADMIN.DISABLE_ILM	-
5	DBMS_ILM_ADMIN.ENABLE_AUTO_OPTIMIZE	不支持	-
6	DBMS_ILM_ADMIN.ENABLE_ILM	DBE_ILM_ADMIN.ENABLE_ILM	-
7	DBMS_ILM_ADMIN.IGNORE_AUTO_OPTIMIZE_CRITERIA	不支持	-
8	DBMS_ILM_ADMIN.SET_HEAT_MAP_ALL	不支持	-
9	DBMS_ILM_ADMIN.SET_HEAT_MAP_START	不支持	-
10	DBMS_ILM_ADMIN.SET_HEAT_MAP_TABLE	不支持	-

表 7-228 DBMS\_COMPRESSION 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	DBMS_COMPRESSION.GET_COMPRESSION_RATIO	支持，有差异，DBE_COMPRESSION.GET_COMPRESSION_RATIO	<ul style="list-style-type: none"> <li>• GaussDB不支持LOBs的压缩率获取。</li> <li>• 对于单个对象的压缩率获取： <ul style="list-style-type: none"> <li>- GaussDB入参comptype取值仅支持1（未压缩）和2（高级压缩），Oracle还支持1024、2048等取值。</li> <li>- GaussDB入参objtype取值仅支持1（表对象），而Oracle还支持2（索引对象）。</li> <li>- Oracle数据库使用subset_numrows参数直接来决定采样的行数（即为参数的取值），而GaussDB则使用sample_ratio（采样率）来间接确定采样的行数。</li> </ul> </li> </ul>

序号	Oracle数据库	GaussDB数据库	差异
2	DBMS_COMPRESSION.GET_COMPRESSION_TYPE	支持，有差异，DBE_COMPRESSION.GET_COMPRESSION_TYPE	<ul style="list-style-type: none"> <li>Oracle使用rowid来指定待获取压缩类型的行，而GaussDB则是使用行的ctid来指定。</li> <li>返回值为comptype，其取值差异同GET_COMPRESSION_RATIO。</li> </ul>

表 7-229 DBMS\_HEAT\_MAP 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	DBMS_HEAT_MAP.BLOCK_HEAT_MAP	不支持	-
2	DBMS_HEAT_MAP.EXTENT_HEAT_MAP	不支持	-
3	DBMS_HEAT_MAP.OBJECT_HEAT_MAP	不支持	-
4	DBMS_HEAT_MAP.SEGMENT_HEAT_MAP	不支持	-
5	DBMS_HEAT_MAP.TABLESPACE_HEAT_MAP	不支持	-
6	不支持	DBE_HEAT_MAP.ROW_HEAT_MAP	详见《开发者指南》中“存储过程 > 高级包 > 二次封装接口（推荐）> DBE_HEAT_MAP”章节。

表 7-230 DBMS\_DESCRIBE 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	DBMS_DESCRIBE. DESCRIBE_PROCE DURE	支持，有差异， DBE_DESCRIBE.DE SCRIBE_PROCE DURE	<ul style="list-style-type: none"> <li>datatype参数与O存在差异，GaussDB返回数据类型的oid，O数据库返回O数据库内部的数据类型的编号。</li> <li>datalength、dataprecision和scale因GaussDB创建存储过程或函数时无法保留类型的约束（如number(7,2)、varchar2(20)等），该三个参数置0处理；Oracle可使用%type方法获得带约束的数据类型。</li> <li>具体的行为差异详见《开发者指南》&gt;“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐）&gt; DBE_DESCRIBE”章节。</li> </ul>

表 7-231 DBMS\_STATS 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	DBMS_STATS.ALTE R_STATS_HISTO RY_RETENTION	不支持	-
2	DBMS_STATS.CAN CEL_ADVISOR_TA SK	不支持	-

序号	Oracle数据库	GaussDB数据库	差异
3	DBMS_STATS.CONFIGURE_ADVISOR_FILTER	不支持	-
4	DBMS_STATS.CONFIGURE_ADVISOR_OBJ_FILTER	不支持	-
5	DBMS_STATS.CONFIGURE_ADVISOR_OPR_FILTER	不支持	-
6	DBMS_STATS.CONFIGURE_ADVISOR_RULE_FILTER	不支持	-
7	DBMS_STATS.CREATE_ADVISOR_TASK	不支持	-
8	DBMS_STATS.CONVERT_RAW_VALUE	不支持	-
9	DBMS_STATS.CONVERT_RAW_VALUE_NVARCHAR	不支持	-
10	DBMS_STATS.CONVERT_RAW_VALUE_ROWID	不支持	-
11	DBMS_STATS.COPY_TABLE_STATS	不支持	-
12	DBMS_STATS.CREATE_EXTENDED_STATS	不支持	-
13	DBMS_STATS.CREATE_STAT_TABLE	DBE_STATS.CREATE_STAT_TABLE	<ul style="list-style-type: none"> <li>GaussDB中ownname应传schema名。</li> <li>GaussDB仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> </ul>

序号	Oracle数据库	GaussDB数据库	差异
14	DBMS_STATS.DELETE_COLUMN_STATS	DBE_STATS.DELETE_COLUMN_STATS	<ul style="list-style-type: none"> <li>GaussDB中 ownname应传 schema名。</li> <li>GaussDB仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> <li>GaussDB中，使用该接口可以设置表达式统计信息，但 tablename应传该表达式对应的索引名。</li> </ul>
15	DBMS_STATS.DELETE_DATABASE_PREFS	不支持	-
16	DBMS_STATS.DELETE_DATABASE_STATS	不支持	-
17	DEDBMS_STATS.DELETE_DICTIONARY_STATS	不支持	-
18	DBMS_STATS.DELETE_FIXED_OBJECTS_STATS	不支持	-
19	DBMS_STATS.DELETE_INDEX_STATS	DBE_STATS.DELETE_INDEX_STATS	<ul style="list-style-type: none"> <li>GaussDB中 ownname应传 schema名。</li> <li>GaussDB仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> </ul>



序号	Oracle数据库	GaussDB数据库	差异
20	DBMS_STATS.DEL ETE_PENDING_ST ATS	不支持	-
21	DBMS_STATS.DEL ETE_PROCESSING _RATE	不支持	-
22	DBMS_STATS.DEL ETE_SCHEMA_PRE FS	不支持	-
23	DBMS_STATS.DEL ETE_SCHEMA_STA TS	DBE_STATS.DELET E_SCHEMA_STATS	<ul style="list-style-type: none"> <li>● GaussDB中 ownname应传 schema名。</li> <li>● GaussDB仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> </ul>
24	DBMS_STATS.DEL ETE_SYSTEM_STA TS	不支持	-
25	DBMS_STATS.DEL ETE_TABLE_PREFS	不支持	-
26	DBMS_STATS.DEL ETE_TABLE_STATS	DBE_STATS.DELET E_TABLE_STATS	<ul style="list-style-type: none"> <li>● GaussDB中 ownname应传 schema名。</li> <li>● GaussDB仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> </ul>
27	DBMS_STATS.DIFF _TABLE_STATS_IN _HISTORY	不支持	-

序号	Oracle数据库	GaussDB数据库	差异
28	DBMS_STATS.DIFF_TABLE_STATS_IN_PENDING	不支持	-
29	DBMS_STATS.DIFF_TABLE_STATS_IN_STATTAB	不支持	-
30	DBMS_STATS.DROP_ADVISOR_TASK	不支持	-
31	DBMS_STATS.DROP_EXTENDED_STATS	不支持	-
32	DBMS_STATS.DROP_STAT_TABLE	DBE_STATS.DROP_STAT_TABLE	-
33	DBMS_STATS.EXECUTE_ADVISOR_TASK	不支持	-

序号	Oracle数据库	GaussDB数据库	差异
34	DBMS_STATS.EXPORT_COLUMN_STATS	DBE_STATS.EXPORT_COLUMN_STATS	<ul style="list-style-type: none"> <li>• GaussDB中 ownname应传 schema名。</li> <li>• GaussDB仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> <li>• 导出的列级统计信息与 pg_statistic表保持一致，多列和 pg_statistic_ext表保持一致。</li> <li>• 支持导出索引表达式统计信息。要求 tabname传的是索引名称，colname传的是索引表达式名称。</li> <li>• 权限：需要具有查询表的 analyze权限以及stattab表的 siud权限。</li> </ul>
35	DBMS_STATS.EXPORT_DATABASE_PREFS	不支持	-
36	DBMS_STATS.EXPORT_DATABASE_STATS	不支持	-
37	DBMS_STATS.EXPORT_DICTIONARY_STATS	不支持	-
38	DBMS_STATS.EXPORT_FIXED_OBJECTS_STATS	不支持	-

序号	Oracle数据库	GaussDB数据库	差异
39	DBMS_STATS.EXPORT_INDEX_STAT S	DBE_STATS.EXPOR T_INDEX_STATS	<ul style="list-style-type: none"> <li>• GaussDB中 ownname应传 schema名。</li> <li>• GaussDB仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> <li>• stattab表中，导出的表、分区级统计信息为numrows、numblocks、relallvisible，分别对应系统表pg_class、pg_partition的reltuples、relpages、relallvisible。</li> <li>• 权限：需要具有查询表的analyze权限以及stattab表的siud权限。</li> </ul>
40	DBMS_STATS.EXPORT_PENDING_STAT S	不支持	-
41	DBMS_STATS.EXPORT_SCHEMA_P REFS	不支持	-

序号	Oracle数据库	GaussDB数据库	差异
42	DBMS_STATS.EXPORT_SCHEMA_STATS	DBE_STATS.EXPORT_SCHEMA_STATS	<ul style="list-style-type: none"> <li>• GaussDB中 ownname应传 schema名。</li> <li>• GaussDB仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> <li>• stattab表中，导出的表、分区级统计信息为numrows、numblocks、relallvisible，分别对应系统表pg_class、pg_partition的reltuples，relpages，relallvisible。导出表相关列级统计信息与pg_statistic表和pg_statistic_ext表保持一致。</li> <li>• 权限：需要具有stattab表的siud权限。</li> </ul>
43	DBMS_STATS.EXPORT_SYSTEM_STATS	不支持	-
44	DBMS_STATS.EXPORT_TABLE_PREFS	不支持	-

序号	Oracle数据库	GaussDB数据库	差异
45	DBMS_STATS.EXP ORT_TABLE_STATS	DBE_STATS.EXPOR T_TABLE_STATS	<ul style="list-style-type: none"> <li>• GaussDB中 ownname应传 schema名。</li> <li>• GaussDB仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> <li>• stattab表中，导出的表、分区级统计信息为numrows、numblocks、relallvisible，分别对应系统表pg_class、pg_partition的reltuples，relpages，relallvisible。级联导出的列级统计信息与pg_statistic表和pg_statistic_ext表保持一致。</li> <li>• 权限：需要具有查询表的analyze权限以及stattab表的siud权限。</li> </ul>
46	DBMS_STATS.FLU SH_DATABASE_M ONITORING_INFO	不支持	-
47	DBMS_STATS.GAT HER_DATABASE_S TATS	不支持	-
48	DBMS_STATS.GAT HER_DICTIONARY _STATS	不支持	-

序号	Oracle数据库	GaussDB数据库	差异
49	DBMS_STATS.GATHER_FIXED_OBJECTS_STATS	不支持	-
50	DBMS_STATS.GATHER_INDEX_STATS	不支持	-
51	DBMS_STATS.GATHER_PROCESSING_RATE	不支持	-
52	DBMS_STATS.GATHER_SCHEMA_STATS	不支持	-
53	DBMS_STATS.GATHER_SYSTEM_STATS	不支持	-
54	DBMS_STATS.GATHER_TABLE_STATS	不支持	-
55	DBMS_STATS.GENERATE_STATS	不支持	-
56	DBMS_STATS.GET_ADVISOR_OPR_FILTER	不支持	-
57	DBMS_STATS.GET_ADVISOR_RECS	不支持	-
58	DBMS_STATS.GET_COLUMN_STATS	不支持	-
59	DBMS_STATS.GET_INDEX_STATS	不支持	-
60	DBMS_STATS.GET_PARAM	不支持	-
61	DBMS_STATS.GET_PREFS	不支持	-
62	DBMS_STATS.GET_STATS_HISTORY_AVAILABILITY	DBE_STATS.GET_STATS_HISTORY_AVAILABILITY	GaussDB查询到的是全库存在的最早历史统计信息的收集时间。
63	DBMS_STATS.GET_STATS_HISTORY_RETENTION	DBE_STATS.GET_STATS_HISTORY_RETENTION	-

序号	Oracle数据库	GaussDB数据库	差异
64	DBMS_STATS.GET_SYSTEM_STATS	不支持	-
65	DBMS_STATS.GET_TABLE_STATS	不支持	-
66	DBMS_STATS.IMPLEMENT_ADVISOR_TASK	不支持	-
67	DBMS_STATS.IMPORT_COLUMN_STATS	DBE_STATS.IMPORT_COLUMN_STATS	<ul style="list-style-type: none"> <li>• GaussDB中ownname应传schema名。</li> <li>• GaussDB仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> <li>• 导出单列col导出的统计信息与pg_statistic表保持一致。多列ext-col导出的统计信息与pg_statistic_ext表保持一致。</li> <li>• 支持导入索引表达式统计信息。要求tablename传的是索引名称，colname传的是索引表达式名称。</li> <li>• 权限：需要具有查询表的analyze权限以及stattab表的siud权限。</li> </ul>
68	DBMS_STATS.IMPORT_DATABASE_PREFS	不支持	-



序号	Oracle数据库	GaussDB数据库	差异
69	DBMS_STATS.IMP ORT_DATABASE_S TATS	不支持	-
70	DBMS_STATS.IMP ORT_DICTIONARY _STATS	不支持	-
71	DBMS_STATS.IMP ORT_FIXED_OBJEC TS_STATS	不支持	-
72	DBMS_STATS.IMP ORT_INDEX_STAT S	DBE_STATS.IMPOR T_INDEX_STATS	<ul style="list-style-type: none"> <li>• GaussDB中 ownname应传 schema名。</li> <li>• GaussDB仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> <li>• statab表中，导入的表、分区级统计信息为numrows、numblocks、relallvisible，分别对应系统表pg_class、pg_partition的reltuples，relpages，relallvisible。</li> <li>• 权限：需要具有查询表的analyze权限以及statab表的siud权限。</li> </ul>
73	DBMS_STATS.IMP ORT_SCHEMA_PR EFS	不支持	-

序号	Oracle数据库	GaussDB数据库	差异
74	DBMS_STATS.IMP ORT_SCHEMA_ST ATS	DBE_STATS.IMPOR T_SCHEMA_STATS	<ul style="list-style-type: none"> <li>• GaussDB中 ownname应传 schema名。</li> <li>• GaussDB仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> <li>• stattab表中，导入的表、分区级统计信息为numrows、numblocks、relallvisible，分别对应系统表pg_class、pg_partition的reltuples，relpages，relallvisible。导入表相关列级统计信息与pg_statistic表和pg_statistic_ext表保持一致。</li> <li>• 权限：需要具有stattab表的siud权限。</li> </ul>
75	DBMS_STATS.IMP ORT_SYSTEM_ST ATS	不支持	-
76	DBMS_STATS.IMP ORT_TABLE_PREF S	不支持	-

序号	Oracle数据库	GaussDB数据库	差异
77	DBMS_STATS.IMP ORT_TABLE_STATS	DBE_STATS.IMPOR T_TABLE_STATS	<ul style="list-style-type: none"> <li>• GaussDB中 ownname应传 schema名。</li> <li>• GaussDB仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> <li>• statab表中，导入的表、分区级统计信息为numrows、numblocks、relallvisible，分别对应系统表pg_class、pg_partition的reltuples，relpages，relallvisible。级联导入的列级统计信息与pg_statistic表和pg_statistic_ext表保持一致。</li> <li>• 权限：需要具有查询表的analyze权限以及statab表的siud权限。</li> </ul>
78	DBMS_STATS.INTE RRUPT_ADVISOR_ TASK	不支持	-
79	DBMS_STATS.LOC K_PARTITION_STA TS	DBE_STATS.LOCK_ PARTITION_STATS	GaussDB中 ownname应传 schema名。
80	DBMS_STATS.LOC K_SCHEMA_STATS	DBE_STATS.LOCK_ SCHEMA_STATS	GaussDB中 ownname应传 schema名。

序号	Oracle数据库	GaussDB数据库	差异
81	DBMS_STATS.LOC K_TABLE_STATS	DBE_STATS.LOCK_ TABLE_STATS	GaussDB中 ownname应传 schema名。
82	DBMS_STATS.MER GE_COL_USAGE	不支持	-
83	DBMS_STATS.PRE PARE_COLUMN_V ALUES	不支持	-
84	DBMS_STATS.PRE PARE_COLUMN_V ALUES_ROWID	不支持	-
85	DBMS_STATS.PUB LISH_PENDING_S TATS	不支持	-
86	DBMS_STATS.PUR GE_STATS	DBE_STATS.PURGE _STATS	-
87	DBMS_STATS.REM AP_STAT_TABLE	不支持	-
88	DBMS_STATS.REP ORT_ADVISOR_TA SK	不支持	-
89	DBMS_STATS.REP ORT_COL_USAGE	不支持	-
90	DBMS_STATS.REP ORT_GATHER_AU TO_STATS	不支持	-
91	DBMS_STATS.REP ORT_GATHER_DA TABASE_STATS	不支持	-
92	DBMS_STATS.REP ORT_GATHER_DIC TIONARY_STATS	不支持	-
93	DBMS_STATS.REP ORT_GATHER_FIX ED_OBJ_STATS	不支持	-
94	DBMS_STATS.REP ORT_GATHER_SC HEMA_STATS	不支持	-
95	DBMS_STATS.REP ORT_STATS_OPER ATIONS	不支持	-

序号	Oracle数据库	GaussDB数据库	差异
96	DBMS_STATS.RES ET_ADVISOR_TAS K	不支持	-
97	DBMS_STATS.RES ET_COL_USAGE	不支持	-
98	DBMS_STATS.RES ET_GLOBAL_PREF _DEFAULTS	不支持	-
99	DBMS_STATS.RES ET_PARAM_DEFA ULTS	不支持	-
100	DBMS_STATS.RES TORE_DICTIONAR Y_STATS	不支持	-
101	DBMS_STATS.RES TORE_FIXED_OBJE CTS_STATS	不支持	-
102	DBMS_STATS.RES TORE_SCHEMA_S TATS	DBE_STATS.RESTO RE_SCHEMA_STAT S	<ul style="list-style-type: none"> <li>• GaussDB中 ownname应传 schema名。</li> <li>• GaussDB仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> </ul>
103	DBMS_STATS.RES TORE_SYSTEM_ST ATS	不支持	-

序号	Oracle数据库	GaussDB数据库	差异
104	DBMS_STATS.RES TORE_TABLE_STAT S	DBE_STATS.RESTO RE_TABLE_STATS	<ul style="list-style-type: none"> <li>GaussDB中 ownname应传 schema名。</li> <li>GaussDB仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> </ul>
105	DBMS_STATS.RES UME_ADVISOR_T ASK	不支持	-
106	DBMS_STATS.SCRI PT_ADVISOR_TAS K	不支持	-
107	DBMS_STATS.SEE D_COL_USAGE	不支持	-
108	DBMS_STATS.SET_ ADVISOR_TASK_P ARAMETER	不支持	-
109	DBMS_STATS.SET_ COLUMN_STATS	DBE_STATS.SET_C OLUMN_STATS	<ul style="list-style-type: none"> <li>GaussDB中 ownname应传 schema名。</li> <li>GaussDB仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> </ul>
110	DBMS_STATS.SET_ DATABASE_PREFS	不支持	-
111	DBMS_STATS.SET_ GLOBAL_PREFS	不支持	-

序号	Oracle数据库	GaussDB数据库	差异
112	DBMS_STATS.SET_INDEX_STATS	DBE_STATS.SET_INDEX_STATS	<ul style="list-style-type: none"> <li>GaussDB中 ownname应传 schema名。</li> <li>GaussDB仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> <li>GaussDB中新增了 relallvisible入参。</li> </ul>
113	DBMS_STATS.SET_PARAM	不支持	-
114	DBMS_STATS.SET_PROCESSING_RATE	不支持	-
115	DBMS_STATS.SET_SCHEMA_PREFS	不支持	-
116	DBMS_STATS.SET_SYSTEM_STATS	不支持	-
117	DBMS_STATS.SET_TABLE_PREFS	不支持	-
118	DBMS_STATS.SET_TABLE_STATS	DBE_STATS.SET_TABLE_STATS	<ul style="list-style-type: none"> <li>GaussDB中 ownname应传 schema名。</li> <li>GaussDB仅支持部分入参功能，详见《开发者指南》中“存储过程 &gt; 高级包 &gt; 二次封装接口（推荐） &gt; DBE_STATS”章节。</li> <li>GaussDB中新增了 relallvisible入参。</li> </ul>

序号	Oracle数据库	GaussDB数据库	差异
119	DBMS_STATS.SHOW_EXTENDED_STATS_NAME	不支持	-
120	DBMS_STATS.TRANSFER_STATS	不支持	-
121	DBMS_STATS.UNLOCK_PARTITION_STATS	DBE_STATS.UNLOCK_PARTITION_STATS	GaussDB中ownname应传schema名。
122	DBMS_STATS.UNLOCK_SCHEMA_STATS	DBE_STATS.UNLOCK_SCHEMA_STATS	GaussDB中ownname应传schema名。
123	DBMS_STATS.UNLOCK_TABLE_STATS	DBE_STATS.UNLOCK_TABLE_STATS	GaussDB中ownname应传schema名。
124	DBMS_STATS.UPGRADE_STAT_TABLE	不支持	-

表 7-232 DBMS\_XMLGEN 兼容性说明

序号	Oracle数据库	GaussDB数据库	差异
1	DBMS_XMLGEN.CONVERT	DBE_XMLGEN.CONVERT	-
2	DBMS_XMLGEN.NEWCONTEXT	DBE_XMLGEN.NEWCONTEXT	-
3	DBMS_XMLGEN.NEWCONTEXTFROMHIERARCHY	DBE_XMLGEN.NEWCONTEXTFROMHIERARCHY	<ul style="list-style-type: none"> <li>● GaussDB生成的递归XML最大深度不能超过5000万层。</li> <li>● Oracle的newcontextfromhierarchy方法对于connect by语句生成的xml是带xml头的，但是对于直接构造的数据不带xml头，GaussDB均带xml头。</li> </ul>



序号	Oracle数据库	GaussDB数据库	差异
4	DBMS_XMLGEN.SETCONVERTSPECIALCHARS	DBE_XMLGEN.SETCONVERTSPECIALCHARS	-
5	DBMS_XMLGEN.SETNULLHANDLING	DBE_XMLGEN.SETNULLHANDLING	-
6	DBMS_XMLGEN.SETROWSETTAG	DBE_XMLGEN.SETROWSETTAG	-
7	DBMS_XMLGEN.SETROWTAG	DBE_XMLGEN.SETROWTAG	-
8	DBMS_XMLGEN.USENULLATTRIBUTEINDICATOR	DBE_XMLGEN.USENULLATTRIBUTEINDICATOR	-
9	DBMS_XMLGEN.USEITEMTAGSFORCOLL	DBE_XMLGEN.USEITEMTAGSFORCOLL	-
10	DBMS_XMLGEN.GETNUMROWSPROCESSED	DBE_XMLGEN.GETNUMROWSPROCESSED	-
11	DBMS_XMLGEN.SETMAXROWS	DBE_XMLGEN.SETMAXROWS	-
12	DBMS_XMLGEN.ETSKIPROWS	DBE_XMLGEN.ETSKIPROWS	-
13	DBMS_XMLGEN.RESTARTQUERY	DBE_XMLGEN.RESTARTQUERY	<ul style="list-style-type: none"> <li>• GaussDB: 调用 RESTARTQUERY方法后对更新的数据不可见。</li> <li>• Oracle: 调用 RESTARTQUERY方法后对更新的数据可见。</li> </ul>
14	DBMS_XMLGEN.GETXMLTYPE	DBE_XMLGEN.GETXMLTYPE	-
15	DBMS_XMLGEN.GETXML	DBE_XMLGEN.GETXML	-
16	DBMS_XMLGEN.CLOSECONTEXT	DBE_XMLGEN.CLOSECONTEXT	-

# 8 与 MySQL 兼容性说明

---

## 8.1 分布式

### 8.1.1 MySQL 数据库兼容 MySQL 模式

#### 8.1.1.1 MySQL 数据库兼容性概述

本文档主要介绍GaussDB数据库的MySQL 5.7兼容模式（即`sql_compatibility = 'MYSQL'`、且设置参数`b_format_version='5.7'`、`b_format_dev_version='s1'`时）与MySQL5.7数据库的兼容性对比信息。仅介绍503.0.0版本后新增的兼容性特性，特性的相关规格和约束建议在开发者指南中查看。

GaussDB数据库在数据类型、SQL功能和数据库对象等基本功能上与MySQL数据库兼容。

由于GaussDB数据库与MySQL数据库底层框架实现存在差异，GaussDB数据库与MySQL数据库仍存在部分差异。

#### 8.1.1.2 数据类型

GaussDB数据库的数据类型大部分功能场景与MySQL一致，但存在部分差异。

- 除特别说明，部分数据类型精度、标度、位数大小等不支持用浮点型数值定义，建议使用合法的整型数值定义。

### 8.1.1.2.1 数值数据类型

表 8-1 整数类型

序号	MySQL数据库	GaussDB数据库	差异
1	BOOL	不完全兼容	MySQL: BOOL/BOOLEAN类型实际映射为TINYINT类型。
2	BOOLEAN	不完全兼容	<p>GaussDB: 支持BOOL, 其中:</p> <ul style="list-style-type: none"> <li>“真”值的有效文本值是: TRUE、't'、'true'、'y'、'yes'、'1'、'TRUE'、true、'on'、以及所有非0数值。</li> <li>“假”值的有效文本值是: FALSE、'f'、'false'、'n'、'no'、'0'、0、'FALSE'、false、'off'。</li> </ul> <p>使用TRUE和FALSE是比较规范的做法(也是SQL兼容的做法)。</p>
3	TINYINT[(M)] [UNSIGNED]	支持	详细请参见说明。
4	SMALLINT[(M)] [UNSIGNED]	支持	详细请参见说明。
5	MEDIUMINT[(M)] [UNSIGNED]	支持	<p>MySQL存储MEDIUMINT数据需要3字节。</p> <ul style="list-style-type: none"> <li>带符号的范围是-8,388,608 ~ +8,388,607。</li> <li>无符号的范围是0 ~ +16,777,215。</li> </ul> <p>GaussDB映射为INT类型, 存储需要4字节。</p> <ul style="list-style-type: none"> <li>带符号的范围是-2,147,483,648 ~ +2,147,483,647。</li> <li>无符号的范围是0 ~ +4,294,967,295。</li> </ul> <p>其他差异请参见说明。</p>
6	INT[(M)] [UNSIGNED]	支持	详细请参见说明。
7	INTEGER[(M)] [UNSIGNED]	支持	详细请参见说明。

序号	MySQL数据库	GaussDB数据库	差异
8	BIGINT[(M)] [UNSIGNED]	支持	详细请参见说明。

## 说明

- 输入格式：
  - MySQL  
对于类似“asbd”、“12dd”、“12 12”等字符场景的输入，会采取截断或返回0值并上报WARNING处理，在严格模式插表时失败。
  - GaussDB
    - 整数类型（TINYINT、SMALLINT、MEDIUMINT、INT、INTEGER、BIGINT）的输入，当非法字符串部分被截断时，如“12@3”，会直接截断并无提示信息，插表成功。
    - 当整数类型全部被截断（如“@123”）或字符串为空时，返回0，且插表成功。
- 操作符：
  - +、-、\*操作符  
GaussDB：INT/INTEGER/SMALLINT/BIGINT在进行运算时，返回值为类型本身，不会向上提升类型，当返回值超范围时报错。  
MySQL：支持提升类型到BIGINT后计算。
  - |、&、^、~运算符  
GaussDB：在类型所占用BIT位中计算；GaussDB中^表示指数运算，如需使用异或运算符，使用#替换。  
MySQL：提升类型计算。
- 负数显示类型转换：  
GaussDB：宽松模式结果为0，严格模式报错。  
MySQL：依据其二进制将最高位替换成数值位计算结果，例如(-1)::uint4 = 4294967295。
- 其他差异：  
INT[(M)]精度，MySQL控制格式化输出，GaussDB仅语法支持，不支持功能。
- 聚集函数：
  - variance：GaussDB表示样本方差，MySQL表示总体方差。
  - stddev：GaussDB表示样本标准差，MySQL表示总体标准差。
- 显示宽度：
  - 在为整型数字列指明宽度信息时，如果不同时指定ZEROFILL，则宽度信息在表结构描述中不显示。
  - INSERT语句插入字符类型字段，GaussDB统一补齐0后插入。
  - JOIN USING语句，涉及类型推导，MySQL默认第一张表列，GaussDB若结果为有符号类型则宽度信息失效，否则为第一张表字段宽度。
  - greatest/least、ifnull/if、case when/decode，MySQL不补齐0，GaussDB在类型及宽度信息一致时补齐0。
  - 作为函数/存储过程出入参、返回值时，MySQL支持功能、GaussDB语法不报错功能不支持。

表 8-2 任意精度类型

序号	MySQL数据库	GaussDB数据库	差异
1	DECIMAL[(M[,D])]	支持	<ul style="list-style-type: none"> <li>操作符：GaussDB中“^”表示指数运算，如需使用异或运算符，使用“#”替换；MySQL中“^”表示异或。</li> <li>取值范围：精度M，标度D不支持浮点型数值输入，只支持整型数值输入。</li> <li>输入格式：当字符串入参全部被截断时不会报错，如‘@123’；只有被部分截断时才会报错，如‘12@3’。</li> </ul>
2	NUMERIC[(M[,D])]	支持	
3	DEC[(M[,D])]	支持	
4	FIXED[(M[,D])]	不支持	-

表 8-3 浮点类型

序号	MySQL数据库	GaussDB数据库	差异
1	FLOAT[(M,D)]	支持	<ul style="list-style-type: none"> <li>分区表支持：FLOAT数据类型不支持KEY键值分区策略分区表。</li> <li>操作符：GaussDB中“^”表示指数运算，如需使用异或运算符，使用“#”替换；MySQL中“^”表示异或。</li> <li>取值范围：精度M，标度D不支持浮点型数值输入，只支持整型数值输入。</li> <li>输出格式：对于非法入参一律报错ERROR，不会在sql_mode="的宽松模式下报WARNING。</li> </ul>
2	FLOAT(p)	支持	<ul style="list-style-type: none"> <li>分区表支持：FLOAT数据类型不支持KEY键值分区策略分区表。</li> <li>操作符：数值类型使用^操作符，与MySQL不一致，GaussDB中^操作符为取指数运算。</li> <li>取值范围：定义精度p时，仅支持使用合法的整型数据类型。</li> <li>输出格式：精度M，标度D不支持浮点型数值输入，只支持整型数值输入。</li> <li>对于非法入参一律报错ERROR，不会在sql_mode="的宽松模式下报WARNING。</li> </ul>

序号	MySQL数据库	GaussDB数据库	差异
3	DOUBLE[(M,D)]	支持	<ul style="list-style-type: none"> <li>分区表支持：DOUBLE数据类型不支持KEY键值分区策略分区表。</li> <li>操作符：GaussDB中“^”表示指数运算，如需使用异或运算符，使用“#”替换；MySQL中“^”表示异或。</li> <li>取值范围：精度M，标度D不支持浮点型数值输入，只支持整型数值输入。</li> <li>输出格式：对于非法入参一律报错ERROR，不会在sql_mode="的宽松模式下报WARNING。</li> </ul>
4	DOUBLE PRECISION[(M,D)]	支持	<ul style="list-style-type: none"> <li>操作符：GaussDB中“^”表示指数运算，如需使用异或运算符，使用“#”替换；MySQL中“^”表示异或</li> <li>取值范围：精度M，标度D不支持浮点型数值输入，只支持整型数值输入。</li> <li>输出格式：对于非法入参一律报错ERROR，不会在sql_mode="的宽松模式下报WARNING。</li> </ul>
5	REAL[(M,D)]	支持	<ul style="list-style-type: none"> <li>分区表支持：REAL数据类型不支持KEY值分区策略分区表。</li> <li>操作符：GaussDB中“^”表示指数运算，如需使用异或运算符，使用“#”替换；MySQL中“^”表示异或</li> <li>取值范围：精度M，标度D不支持浮点型数值输入，只支持整型数值输入。</li> <li>输出格式：对于非法入参一律报错ERROR，不会在sql_mode="的宽松模式下报WARNING。</li> </ul>

表 8-4 序列整数

序号	MySQL数据库	GaussDB数据库	差异
1	SERIAL	不完全兼容	<p>SERIAL介绍详见GaussDB 开发者指南 数值类型章节。</p> <p>规格上与MySQL的差异如下： CREATE TABLE test(f1 serial, f2 CHAR(20));</p> <ul style="list-style-type: none"> <li>类型定义差异，MySQL的serial是映射到BIGINT(20) UNSIGNED NOT NULL AUTO_INCREMENT UNIQUE，GaussDB的serial是映射到INTEGER NOT NULL DEFAULT nextval('test_f1_seq'::regclass)。如： -- MySQL serial的定义： mysql&gt; SHOW CREATE TABLE test\G ***** 1. row ***** Table: test Create Table: CREATE TABLE `test` ( `f1` bigint(20) unsigned NOT NULL AUTO_INCREMENT, `f2` char(20) DEFAULT NULL, UNIQUE KEY `f1` (`f1`) ) ENGINE=InnoDB DEFAULT CHARSET=utf8 1 row in set (0.00 sec)  -- GaussDB serial的定义 gaussdb=# \d+ test  Table "public.test" Column   Type   Modifiers   Storage   Stats target   Description -----+----- +-----+-----+----- f1   integer   not null default nextval('test_f1_seq'::regclass)   plain     f2   character(20)   extended   Has OIDs: no Options: orientation=row, compression=no, storage_type=USTORE</li> <li>INSERT场景下serial类型DEFAULT值的差异。如： -- MySQL插入serial的DEFAULT值 mysql&gt; INSERT INTO test VALUES(DEFAULT, 'aaaa'); Query OK, 1 row affected (0.00 sec)  mysql&gt; INSERT INTO test VALUES(10, 'aaaa'); Query OK, 1 row affected (0.00 sec)  mysql&gt; INSERT INTO test VALUES(DEFAULT, 'aaaa'); Query OK, 1 row affected (0.00 sec)  mysql&gt; SELECT * FROM test; +----+-----+   f1   f2   +----+-----+   1   aaaa     10   aaaa     11   aaaa  </li> </ul>

序号	MySQL数据库	GaussDB数据库	差异
			<pre> +----+-----+ 3 rows in set (0.00 sec)  -- GaussDB插入serial的DEFAULT值 gaussdb=# INSERT INTO test VALUES(DEFAULT, 'aaaa'); INSERT 0 1 gaussdb=# INSERT INTO test VALUES(10, 'aaaa'); INSERT 0 1 gaussdb=# INSERT INTO test VALUES(DEFAULT, 'aaaa'); INSERT 0 1 gaussdb=# SELECT * FROM test;  f1        f2 -----+-----  1   aaaa  2   aaaa 10   aaaa (3 rows) </pre> <ul style="list-style-type: none"> <li>REPLACE场景下serial类型引用列的差异，GaussDB引用列的介绍详见GaussDB 开发者指南 REPLACE语法章节。如： <pre> -- MySQL插入serial引用列的值 mysql&gt; REPLACE INTO test VALUES(f1, 'aaaa'); Query OK, 1 row affected (0.00 sec)  mysql&gt; REPLACE INTO test VALUES(f1, 'bbbb'); Query OK, 1 row affected (0.00 sec)  mysql&gt; SELECT * FROM test; +----+-----+   f1   f2   +----+-----+   1   aaaa     2   bbbb   +----+-----+ 2 rows in set (0.00 sec)  -- GaussDB插入serial引用列的值 gaussdb=# REPLACE INTO test VALUES(f1, 'aaaa'); REPLACE 0 1 gaussdb=# REPLACE INTO test VALUES(f1, 'bbbb'); REPLACE 0 1 gaussdb=# SELECT * FROM test;  f1        f2 -----+-----  0   aaaa  0   bbbb (2 rows) </pre> </li> </ul>



### 8.1.1.2.2 日期与时间数据类型

表 8-5 日期与时间数据类型

序号	MySQL数据库	GaussDB数据库	差异
1	DATE	支持	<p>GaussDB支持date数据类型，与MySQL相比规格上存在如下差异：</p> <ul style="list-style-type: none"> <li>• 输入格式 <ul style="list-style-type: none"> <li>- GaussDB只支持字符类型，不支持数值类型。如支持'2020-01-01'或'20200101'字符串格式，不支持20200101数值输入。MySQL支持数值输入转换为date类型。</li> <li>- 分隔符：GaussDB不支持加号“+”、冒号“:”作为年、月、日之间的分隔符，其他的符号都支持。MySQL所有符号均可作为分隔符。分隔符混合使用的某些场景也不支持，与MySQL也有差异，如'2020-01&gt;01'，'2020/01+01'等，不建议混合使用分隔符，建议使用最常用的“-”、“/”作为分隔符。</li> <li>- 无分隔符：推荐使用完整格式，如'YYYYMMDD'或者'YYMMDD'。其他不完整的格式（包括超长格式）解析的规则与MySQL存在差异，可能报错或者解析的结果与MySQL不一致，不推荐使用。</li> </ul> </li> <li>• 输出格式 <p>GaussDB在sql_mode参数不包含'strict_trans_tables'选项（定义为宽松模式，否则为严格模式）时，允许年、月、日的值是0，但是输出时会按照年、月、日的顺序依次转换为合法的值，如date'0000-00-10'转换为：0002-12-10 BC。非法输入或者超过范围时，会报warning信息，并返回0000-00-00值。MySQL对于包含0值年、月、日的date值会原样输出。</p> </li> <li>• 取值范围 <p>GaussDB的范围是4713-01-01 BC ~ 5874897-12-31 AD，支持公元前的日期，宽松模式下超过范围时，返回的是0值：0000-00-00，严格模式下会报错。MySQL的范围是 0000-00-00 ~ 9999-12-31，宽松模式下超过范围后，各个场景下的表现并不一致，可能报错（如select查询语句中），也可能返回0000-00-00值（如insert时）。此差异会导致date类型作为函数入参时，函数返回的结果存在差异。</p> </li> </ul>

序号	MySQL数据库	GaussDB数据库	差异
			<ul style="list-style-type: none"> <li>● 操作符                             <ul style="list-style-type: none"> <li>- GaussDB仅支持date类型之间的比较操作符“=”、“!=”、“&lt;”、“&lt;=”、“&gt;”、“&gt;=”，返回true或者false；date与interval类型的加法运算，返回结果为date类型；date与interval类型的减法运算，返回结果为date类型；date类型之间的减法运算，返回结果为interval类型。</li> <li>- MySQL date类型和其他数值类型运算时，会先将date转换为数值类型，然后按照数值类型运算，结果也为数值类型。与GaussDB存在差异。如：                                     <pre data-bbox="901 795 1428 1232"> -- MySQL: date + 数值, 先将date类型转换为数值 20200101,再与1相加, 结果为数值类型20200102 mysql&gt; select date'2020-01-01' + 1; +-----+   date'2020-01-01' + 1   +-----+             20200102   +-----+ 1 row in set (0.00 sec)  -- GaussDB: date + 数值, 数值类型会转换为interval 类型 1 day, 然后相加得到新的日期。 gaussdb=# select date'2020-01-01' + 1; ?column? ----- 2020-01-02 (1 row) </pre> </li> </ul> </li> <li>● 类型转换                             相比较MySQL，GaussDB仅支持date类型与char(n)、nchar(n)、datetime、timestamp类型之间的相互转换，不支持与binary、decimal、json、integer、unsigned integer、time 类型之间的转换。集合等场景和复杂表达式场景下公共类型的确定原则与MySQL也不一致，参考<a href="#">数据类型转换</a>章节的描述。                         </li> </ul>

序号	MySQL数据库	GaussDB数据库	差异
2	DATETIME[(fsp)]	支持	<p>GaussDB支持datetime数据类型，与MySQL相比规格上存在如下差异：</p> <ul style="list-style-type: none"> <li>● 输入格式 <ul style="list-style-type: none"> <li>- GaussDB只支持字符类型，不支持数值类型。如支持'2020-01-01 10:20:30.123456'或'20200101102030.123456'字符串格式，不支持如20200101102030.123456的数值类型输入。MySQL支持数值输入转换为datetime类型。</li> <li>- 分隔符：GaussDB不支持加号“+”、冒号“:”作为年、月、日之间的分隔符，其他的符号都支持。仅支持冒号“.”作为时、分、秒之间的分隔符，其他的符号都不支持。分隔符混合使用的某些场景也不支持，与MySQL也有差异，不推荐使用。MySQL支持所有符号作为分隔符。</li> <li>- 无分隔符：GaussDB推荐使用完整格式'YYYYMMDDhhmiss.ffffff'。其他不完整的格式（包括超长格式）解析的规则可能与MySQL存在差异，可能报错或者解析的结果与MySQL不一致，不推荐使用。</li> </ul> </li> <li>● 输出格式： <ul style="list-style-type: none"> <li>- 统一为'YYYY-MM-DD hh:mi:ss.ffffff'的格式，格式与MySQL无差异，且不受DateStyle参数的影响。但是对于精度部分，如果最后几位为0，GaussDB不显示，MySQL会显示。</li> <li>- GaussDB在sql_mode参数不包含'strict_trans_tables'选项（定义为宽松模式，否则为严格模式）时，允许年、月、日值是0，但是输出时会按照年、月、日的顺序依次转换为合法的值，如datetime '0000-00-10 00:00:00'转换为：0002-12-10 00:00:00 BC。非法输入或者超过范围时，会报warning信息，并返回0000-00-00 00:00:00值。MySQL对于包含0值年、月、日的datetime值会原样输出。</li> </ul> </li> <li>● 取值范围 <ul style="list-style-type: none"> <li>4713-11-24 00:00:00.000000 BC ~ 294277-01-09 04:00:54.775806 AD。294277-01-09 04:00:54.775807 AD 返回的是infinity。对于超过范围的值，严格模</li> </ul> </li> </ul>

序号	MySQL数据库	GaussDB数据库	差异
			<p>式下GaussDB会报错，MySQL是否报错取决于使用场景。一般查询场景不报错，而执行DML SQL语句更改表属性的值时报错。宽松模式下GaussDB返回0000-00-00 00:00:00值，MySQL根据使用场景可能报错，也可能返回0000-00-00 00:00:00值或者null值。这个差异会导致以datetime类型为入参的函数执行结果与MySQL也存在差异。</p> <ul style="list-style-type: none"> <li>● 精度 范围0~6，作为表列的类型时缺省为0，与MySQL一致。对于 datetime[(p)] 'str' 表达式场景，GaussDB将(p)作为精度解析，缺省为6，将'str'按照p指定的精度格式化成datetime类型。MySQL不支持 datetime[(p)] 'str'表达式。</li> <li>● 操作符 <ul style="list-style-type: none"> <li>- GaussDB仅支持datetime类型之间的比较操作符“=”、“!=”、“&lt;”、“&lt;=”、“&gt;”、“&gt;=”，返回true或者false；datetime与interval类型的加法运算，返回结果为datetime类型；datetime与interval类型的减法运算，返回结果为datetime类型；datetime类型之间的减法运算，返回结果为interval类型。</li> <li>- MySQL datetime类型和其他数值类型运算时，会先将datetime转换为数值类型，然后按照数值类型运算，结果也为数值类型。与GaussDB存在差异。如： <pre data-bbox="901 1391 1422 1921"> -- MySQL: datetime + 数值, 先将datetime类型转换为数值20201010123456,再与1相加, 结果为数值类型 20201010123457 mysql&gt; select cast('2020-10-10 12:34:56.123456' as datetime) + 1; +-----+   cast('2020-10-10 12:34:56.123456' as datetime) + 1   +-----+                                 20201010123457   +-----+ 1 row in set (0.00 sec)  -- GaussDB: datetime + 数值, 数值类型会转换为 interval类型 1 day, 然后相加得到新的datetime。 gaussdb=# select cast('2020-10-10 12:34:56.123456' as datetime) + 1; ?column? ----- 2020-10-11 12:34:56 (1 row) </pre> </li> </ul> </li> </ul>

序号	MySQL数据库	GaussDB数据库	差异
			<p>将datetime类型与数值的运算结果作为函数的入参，可能导致函数的结果与MySQL也存在差异。</p> <ul style="list-style-type: none"> <li> <b>类型转换</b>                      相比较MySQL，GaussDB仅支持datetime类型与char(n)、varchar(n)、timestamp类型之间的相互转换、datetime到date、time类型的转换（仅赋值和显式转换）。不支持与binary、decimal、json、integer、unsigned integer类型之间的转换。集合等场景和复杂表达式场景下公共类型的确定原则与MySQL也不一致，参考<a href="#">数据类型转换</a>章节的描述。                 </li> <li> <b>时区</b>                      GaussDB支持datetime值中携带时区信息（时区偏移或者时区名），如'2020-01-01 12:34:56.123456 +01:00' 或者 '2020-01-01 2:34:56.123456 CST'。GaussDB会将其转换为当前服务器时区的时间。MySQL不支持（5.7版本不支持，8.0及之后的版本支持）。                 </li> <li>                     GaussDB的datetime数据类型的表字段实际上会被转换为timestamp(p) without time zone类型，查询表信息或者使用工具导出的表结构，其字段的数据类型显示的是timestamp(p) without time zone，而不是datetime。MySQL显示的是datetime(p)。                 </li> </ul>

序号	MySQL数据库	GaussDB数据库	差异
3	TIMESTAMP[(fsp)]	支持	<p>GaussDB支持timestamp数据类型，与MySQL相比规格上存在如下差异：</p> <ul style="list-style-type: none"> <li>● 输入格式： <ul style="list-style-type: none"> <li>- 只支持字符类型，不支持数值类型。如支持'2020-01-01 10:20:30.123456'或'20200101102030.123456'字符串格式，不支持如20200101102030.123456的数值类型输入。MySQL支持数值输入转换为timestamp类型。</li> <li>- 分隔符：不支持加号“+”、冒号“:”作为年、月、日之间的分隔符，其他的符号都支持。仅支持冒号“:”作为时、分、秒之间的分隔符，其他的符号都不支持。分隔符混合使用的某些场景也不支持，与MySQL也有差异，不推荐使用。MySQL支持所有符号作为分隔符。</li> <li>- 无分隔符：推荐使用完整格式'YYYYMMDDhhmiss.ffffff'。其他不完整的格式（包括超长格式）解析的规则可能与MySQL存在差异，可能报错或者解析的结果与MySQL不一致，不推荐使用。</li> </ul> </li> <li>● 输出格式： <ul style="list-style-type: none"> <li>- 统一为'YYYY-MM-DD hh:mi:ss.ffffff'的格式，格式与MySQL无差异，且不受DateStyle参数的影响。但是对于精度部分，如果最后几位为0，GaussDB不显示，MySQL会显示。</li> <li>- GaussDB在sql_mode参数不包含'strict_trans_tables'选项（定义为宽松模式，否则为严格模式）时，允许年、月、日值是0，但是输出时会按照年、月、日的顺序依次转换为合法的值得，如timestamp '0000-00-10 00:00:00' 转换为：0002-12-10 00:00:00 BC。非法输入或者超过范围时，会报warning信息，并返回0000-00-00 00:00:00值。MySQL对于包含0值年、月、日的timestamp值会原样输出。</li> </ul> </li> <li>● 取值范围： <ul style="list-style-type: none"> <li>4713-11-24 00:00:00.000000 BC ~ 294277-01-09 04:00:54.775806 AD。</li> <li>294277-01-09 04:00:54.775807 AD 返回的是infinity。对于超过范围的值，严格模式下GaussDB会报错，MySQL是否报错取决于使用场景。一般查询场景不报错，而</li> </ul> </li> </ul>

序号	MySQL数据库	GaussDB数据库	差异
			<p>执行DML SQL语句更改表属性的值时报错。宽松模式下GaussDB返回0000-00-00 00:00:00值，MySQL根据使用场景可能报错，也可能返回0000-00-00 00:00:00值或者null值。这个差异会导致以timestamp类型为入参的函数执行结果与MySQL也存在差异。</p> <ul style="list-style-type: none"> <li>● 精度： <ul style="list-style-type: none"> <li>范围0~6，作为表列的类型时缺省为0，与MySQL一致。对于 timestamp[(p)] 'str' 表达式场景： <ul style="list-style-type: none"> <li>- GaussDB将(p)作为精度解析，缺省为6，将'str'按照p指定的精度格式化成timestamp类型。</li> <li>- MySQL将timestamp 'str'的含义与GaussDB一致，缺省精度也为6。但是将timestamp(p) 'str'解析为函数调用，p作为timestamp函数的入参，结果返回一个timestamp类型的值，'str'作为投影列的别名。</li> </ul> </li> </ul> </li> <li>● 操作符： <ul style="list-style-type: none"> <li>- GaussDB仅支持timestamp类型之间的比较操作符“=”、“!=”、“&lt;”、“&lt;=”、“&gt;”、“&gt;=”，返回true或者false；timestamp与interval类型的加法运算，返回结果为timestamp类型；timestamp与interval类型的减法运算，返回结果为timestamp类型；timestamp类型之间的减法运算，返回结果为interval类型。</li> <li>- MySQL timestamp类型和其他数值类型运算时，会先将timestamp转换为数值类型，然后按照数值类型运算，结果也为数值类型。与GaussDB存在差异。如： <pre data-bbox="901 1579 1428 1982"> -- MySQL: timestamp + 数值, 先将timestamp类型转换为数值20201010123456.123456,再与1相加, 结果为数值类型20201010123457.123456 mysql&gt; select timestamp '2020-10-10 12:34:56.123456' + 1; +-----+   timestamp '2020-10-10 12:34:56.123456' + 1   +-----+                  20201010123457.123456   +-----+ 1 row in set (0.00 sec)  -- GaussDB: timestamp + 数值, 数值类型会转换为interval类型 1 day, 然后相加得到新的timestamp。 gaussdb=# select timestamp '2020-10-10 12:34:56.123456' + 1; </pre> </li> </ul> </li> </ul>

序号	MySQL数据库	GaussDB数据库	差异
			<div style="border: 1px solid gray; padding: 5px; margin-bottom: 10px;">                     ?column?                      -----                      2020-10-11 12:34:56.123456                      (1 row)                 </div> <p>将timestamp类型与数值的运算结果作为函数的入参，可能导致函数的结果与MySQL也存在差异。</p> <ul style="list-style-type: none"> <li> <b>类型转换：</b>                      相比较MySQL，GaussDB仅支持timestamp类型与char(n)、varchar(n)、datetime类型之间的相互转换、timestamp到date、time类型的转换（仅赋值和显式转换）。不支持与binary、decimal、json、integer、unsigned integer类型之间的转换。集合等场景和复杂表达式场景下公共类型的确定原则与MySQL也不一致，参考<a href="#">数据类型转换</a>章节的描述。                 </li> <li> <b>时区：</b>                      GaussDB支持timestamp值中携带时区信息（时区偏移或者时区名），如 '2020-01-01 12:34:56.123456 +01:00' 或者 '2020-01-01 2:34:56.123456 CST'。GaussDB会将其转换为当前服务器时区的时间。如果更改服务器时区，timestamp类型的值输出时会转换为更改后时区的时间戳。MySQL不支持（5.7版本不支持，8.0及之后的版本支持）。                 </li> <li>                     GaussDB的timestamp数据类型的表字段实际上会被转换为timestamp(p) with time zone类型，查询表信息或者使用工具导出的表结构，其字段的数据类型显示的是timestamp(p) with time zone，而不是timestamp。MySQL显示的是timestamp(p)。                 </li> </ul>



序号	MySQL数据库	GaussDB数据库	差异
4	TIME[(fsp)]	支持	<p>GaussDB支持time数据类型，与MySQL相比规格上存在如下差异：</p> <ul style="list-style-type: none"> <li>● 输入格式： <ul style="list-style-type: none"> <li>- 只支持字符类型，不支持数值类型。如支持'1 10:20:30'或'102030'字符串格式，不支持102030数值输入。MySQL支持数值输入转换为time类型。</li> <li>- 分隔符：GaussDB仅支持冒号“:”作为时、分、秒之间的分隔符，其他的符号都不支持。MySQL支持所有的符号作为分隔符。</li> <li>- 无分隔符：推荐使用完整格式，如'hhmiss.ffffff'。其他不完整的格式（包括超长格式）解析的规则可能与MySQL存在差异，可能报错或者解析的结果与MySQL不一致，不推荐使用。</li> <li>- 分、秒、精度输入负数时，GaussDB数据库可能会忽略第一个负数开始的部分，涉及的部分解析为0，如：'00:00:-10'解析结果为'00:00:00'。也可能报错，如：'00:00:-10000'会解析报错。取决于输入值的范围。而MySQL数据库统一报错。</li> </ul> </li> <li>● 输出格式： <ul style="list-style-type: none"> <li>统一为hh:mi:ss.ffffff的格式，格式与MySQL无差异。但是对于精度部分，如果最后几位为0，GaussDB不显示，MySQL会显示。</li> </ul> </li> <li>● 取值范围： <ul style="list-style-type: none"> <li>-838:59:59.000000 ~ 838:59:59.000000，与MySQL一致。对于超过范围的值，宽松模式下GaussDB无论是查询还是insert/update等DML操作，返回的值都是就近的边界值：-838:59:59或838:59:59。MySQL是查询时报错，DML操作返回的值才是就近边界值，场景上存在差异。此差异会导致time类型作为函数入参时，函数返回的结果也存在差异。</li> </ul> </li> <li>● 精度： <ul style="list-style-type: none"> <li>范围0~6，作为表列的类型时缺省为0，与MySQL一致。对于time(p) 'str'表达式场景，GaussDB将(p)作为精度解析，缺省为6，将'str'按照p指定的精度格式化time类型。MySQL是解析为time函数，p是入参，'str'是投影列的别名。</li> </ul> </li> <li>● 操作符：</li> </ul>

序号	MySQL数据库	GaussDB数据库	差异
			<p>- GaussDB仅支持time类型之间的比较操作符“=”、“!=”、“&lt;”、“&lt;=”、“&gt;”、“&gt;=”，返回true或者false；time与interval类型的加法运算，返回结果为time类型；time与interval类型的减法运算，返回结果为time类型；time类型之间的减法运算，返回结果为interval类型。</p> <p>- MySQL time类型和其他数值类型运算时，会先将time转换为数值类型，然后按照数值类型运算，结果也为数值类型。与GaussDB存在差异。如：</p> <pre>-- MySQL: time + 数值, 先将time类型转换为数值 123456,再与1相加, 结果为数值类型123457 mysql&gt; select time '12:34:56' + 1; +-----+   time '12:34:56' + 1   +-----+            123457   +-----+ 1 row in set (0.00 sec)</pre> <p>-- GaussDB: time + 数值, 数值类型会转换为interval类型 1 day, 然后相加得到新的time, 由于是加了24小时, 得到的仍然是12:34:56。</p> <pre>gaussdb=# select time '12:34:56' + 1; ?column? ----- 12:34:56 (1 row)</pre> <p>将time类型与数值的运算结果作为函数的入参, 可能导致函数的结果与MySQL也存在差异。</p> <ul style="list-style-type: none"> <li>● 类型转换: 相比较MySQL, GaussDB仅支持time类型与char(n)、nchar(n)类型之间的相互转换、datetime、timestamp到time类型的转换。不支持与binary、decimal、date、json、integer、unsigned integer类型之间的转换。集合等场景和复杂表达式场景下公共类型的确定原则与MySQL也不一致, 参考<a href="#">数据类型转换</a>章节的描述。</li> </ul>

序号	MySQL数据库	GaussDB数据库	差异
5	YEAR[(4)]	支持	<p>GaussDB支持year数据类型，与MySQL相比规格上存在如下差异：</p> <ul style="list-style-type: none"> <li>操作符： <ul style="list-style-type: none"> <li>GaussDB仅支持year类型之间的比较操作符：<code>=</code>、<code>!=</code>、<code>&lt;</code>、<code>&lt;=</code>、<code>&gt;</code>、<code>&gt;=</code>。返回true或者false。</li> <li>GaussDB仅支持year类型与int4类型之间的算术操作符：<code>+</code>、<code>-</code>。返回整型值，MySQL是返回无符号整型值。</li> </ul> </li> <li>类型转换： <p>相比较MySQL，GaussDB仅支持year类型与int4类型的转换，仅支持int4、varchar、numeric、date、time、timestamp、timestampz类型到year类型的转换。集合等场景和复杂表达式场景下公共类型的确定原则与MySQL也不一致，参考<a href="#">数据类型转换</a>章节的描述。</p> </li> </ul>
6	INTERVAL	支持	<p>GaussDB支持INTERVAL数据类型，但INTERVAL在MySQL中为表达式，同时存在以下差异：</p> <ul style="list-style-type: none"> <li>不支持字符串类型的日期输入作为运算，如：<code>select '2023-01-01' + interval 1 day</code>。</li> <li>不支持interval expr unit语法中，expr为负整数或浮点数的输入，如：<code>select date'2023-01-01' + interval -1 day</code>。</li> <li>不支持interval expr unit语法中，expr为运算表达式的输入，如：<code>select date'2023-01-01' + interval 4/2 day</code>。</li> <li>interval表达式参与运算时，返回值固定为datetime类型，MySQL为datetime或date类型。运算的逻辑与原有GaussDB保持一致，与MySQL有差异。</li> <li>interval expr unit语法中，expr数值支持的范围会根据unit单位的不同有所差异，最大可支持的范围为[-2147483648, 2147483647]。超过范围时，严格模式报error，宽松模式报warning并返回0值。</li> <li>interval expr unit语法中，expr指定的字段数量大于unit预期的字段数量时，在严格模式，报error；在宽松模式，报warning并返回0值。如unit取值为DAY_HOUR，预期的字段数量为2，expr取值为'1-2-3'，字段数量为3。</li> </ul>

### 8.1.1.2.3 字符串数据类型

表 8-6 字符串数据类型

序号	MySQL数据库	GaussDB数据库	差异
1	CHAR[(M)]	支持	<ul style="list-style-type: none"> <li>● 输入格式                             <ul style="list-style-type: none"> <li>- GaussDB自定义函数参数和返回值不支持长度校验，存储过程参数不支持长度校验，同时也不支持在 PAD_CHAR_TO_FULL_LENGTH打开时补齐正确的空格，MySQL支持。</li> <li>- GaussDB不支持转义字符输入，不支持""双引号输入，MySQL支持。</li> </ul> </li> <li>● 语法                             <p>GaussDB的 Cast ( expr as char ) 语法无法根据输入的字符串长度转成对应的类型，只支持转成varchar类型。不支持 cast( ' ' as char) 和cast( ' ' as char(0))将空串转成char(0)类型。MySQL支持按长度转成对应的类型。</p> </li> <li>● 操作符                             <ul style="list-style-type: none"> <li>- GaussDB能正常转成浮点型的字符串与整型值加减乘除求余，返回值是整型值，MySQL是返回浮点型。</li> <li>- GaussDB除以0会报错，MySQL返回null。</li> <li>- “~”：GaussDB返回负数，MySQL返回8字节无符号整数。</li> <li>- “^”：GaussDB表示次方幂，MySQL表示按位异或。</li> </ul> </li> </ul>

序号	MySQL数据库	GaussDB数据库	差异
2	VARCHAR(M)	支持	<ul style="list-style-type: none"> <li>● 输入格式                             <ul style="list-style-type: none"> <li>- GaussDB的自定义函数参数和返回值不支持长度校验，存储过程参数不支持长度校验，MySQL支持。</li> <li>- GaussDB的自定义函数和存储过程中的临时变量支持长度校验以及严格宽松模式下的报错和截断告警，MySQL不支持。</li> <li>- GaussDB不支持转义字符输入，不支持""双引号输入，MySQL支持。</li> </ul> </li> <li>● 操作符                             <ul style="list-style-type: none"> <li>- GaussDB能正常转成浮点型的字符串与整型值加减乘除求余，返回值是整型值，MySQL是返回浮点型。</li> <li>- GaussDB除以0会报错，MySQL返回null。</li> <li>- “~”：GaussDB返回负数，MySQL返回8字节无符号整数。</li> <li>- “^”：GaussDB表示次方幂，MySQL表示按位异或。</li> </ul> </li> </ul>
3	TINYTEXT	支持	<ul style="list-style-type: none"> <li>● 输入格式                             <ul style="list-style-type: none"> <li>- GaussDB不支持长度限制255字节（最大不超过1G），不支持超限后，根据严格宽松模式报错和截断告警。MySQL支持。</li> <li>- GaussDB不支持转义字符输入，不支持""双引号输入，MySQL支持。</li> </ul> </li> <li>● 操作符                             <ul style="list-style-type: none"> <li>- GaussDB能正常转成浮点型的字符串与整型值加减乘除求余，返回值是整型值，MySQL是返回浮点型。</li> <li>- GaussDB除以0会报错，MySQL返回null。</li> <li>- “~”：GaussDB返回负数，MySQL返回8字节无符号整数。</li> <li>- “^”：GaussDB表示次方幂，MySQL表示按位异或。</li> </ul> </li> </ul>

序号	MySQL数据库	GaussDB数据库	差异
4	TEXT	支持	<ul style="list-style-type: none"> <li>● 输入格式                             <ul style="list-style-type: none"> <li>- GaussDB不支持长度限制65535字节（最大不超过1G），不支持超限后，根据严格宽松模式报错和截断告警。MySQL支持。</li> <li>- GaussDB不支持转义字符输入，不支持""双引号输入，MySQL支持。</li> </ul> </li> <li>● 操作符                             <ul style="list-style-type: none"> <li>- GaussDB能正常转成浮点型的字符串与整型值加减乘除求余，返回值是整型值，MySQL是返回浮点型。</li> <li>- GaussDB除以0会报错，MySQL返回null。</li> <li>- “~”：GaussDB返回负数，MySQL返回8字节无符号整数。</li> <li>- “^”：GaussDB表示次方幂，MySQL表示按位异或。</li> </ul> </li> </ul>
5	MEDIUMTEXT	支持	<ul style="list-style-type: none"> <li>● 输入格式                             <ul style="list-style-type: none"> <li>- GaussDB不支持长度限制16777215字节（最大不超过1G），不支持超限后，根据严格宽松模式报错和截断告警。MySQL支持。</li> <li>- GaussDB不支持转义字符输入，不支持""双引号输入，MySQL支持。</li> </ul> </li> <li>● 操作符                             <ul style="list-style-type: none"> <li>- GaussDB能正常转成浮点型的字符串与整型值加减乘除求余，返回值是整型值，MySQL是返回浮点型。</li> <li>- GaussDB除以0会报错，MySQL返回null。</li> <li>- “~”：GaussDB返回负数，MySQL返回8字节无符号整数。</li> <li>- “^”：GaussDB表示次方幂，MySQL表示按位异或。</li> </ul> </li> </ul>

序号	MySQL数据库	GaussDB数据库	差异
6	LONGTEXT	支持	<ul style="list-style-type: none"> <li>● 输入格式 <ul style="list-style-type: none"> <li>- GaussDB只支持不超过1G，MySQL支持4G-1字节长度。</li> <li>- GaussDB不支持转义字符输入，不支持""双引号输入，MySQL支持。</li> </ul> </li> <li>● 操作符 <ul style="list-style-type: none"> <li>- GaussDB能正常转成浮点型的字符串与整型值加减乘除求余，返回值是整型值，MySQL是返回浮点型。</li> <li>- GaussDB除以0会报错，MySQL返回null。</li> <li>- “~”：GaussDB返回负数，MySQL返回8字节无符号整数。</li> <li>- “^”：GaussDB表示次方幂，MySQL表示按位异或。</li> </ul> </li> </ul>
7	ENUM('value 1','value2',...)	不支持	-
8	SET('value1','value2',...)	不支持	-

#### 8.1.1.2.4 二进制数据类型

表 8-7 二进制数据类型

序号	MySQL数据库	GaussDB数据库	差异
1	BINARY[(M)]	不支持	-
2	VARBINARY(M)	不支持	-
3	TINYBLOB	支持	<ul style="list-style-type: none"> <li>● 取值范围：不支持长度限制255字节（最大不超过1G），不支持超限后，根据严格宽松模式报错和截断告警。</li> <li>● 输入格式：不支持转义字符输入，不支持""双引号输入。</li> <li>● 操作符：不支持算数运算符“+”、“-”、“*”、“/”、“%”；不支持常用逻辑运算符或、与、非（“  ”、“&amp;&amp;”、“!”）；不支持常用位运算符“~”、“&amp;”、“ ”、“^”。</li> </ul>

序号	MySQL数据库	GaussDB数据库	差异
4	BLOB	支持	<ul style="list-style-type: none"> <li>取值范围：不支持长度限制65535字节（最大不超过1G），不支持超限后，根据严格宽松模式报错和截断告警。MySQL支持。</li> <li>输入格式：不支持转义字符输入，不支持""双引号输入。</li> <li>操作符：不支持算数运算符“+”、“-”、“*”、“/”、“%”；不支持常用逻辑运算符或、与、非（“  ”、“&amp;&amp;”、“!”）；不支持常用位运算符“~”、“&amp;”、“ ”、“^”。</li> </ul>
5	MEDIUMBLOB	支持	<ul style="list-style-type: none"> <li>取值范围：不支持长度限制16777215字节（最大不超过1G），不支持超限后，根据严格宽松模式报错和截断告警。MySQL支持。</li> <li>输入格式：不支持转义字符输入，不支持""双引号输入。</li> <li>操作符：不支持算数运算符“+”、“-”、“*”、“/”、“%”；不支持常用逻辑运算符或、与、非（“  ”、“&amp;&amp;”、“!”）；不支持常用位运算符“~”、“&amp;”、“ ”、“^”。</li> </ul>
6	LONGBLOB	支持	<ul style="list-style-type: none"> <li>取值范围：只支持不超过1G，具体范围参照bytea数据类型集中式和分布式规格。</li> <li>输入格式：不支持转义字符输入，不支持""双引号输入。</li> <li>操作符：不支持算数运算符“+”、“-”、“*”、“/”、“%”；不支持常用逻辑运算符或、与、非（“  ”、“&amp;&amp;”、“!”）；不支持常用位运算符“~”、“&amp;”、“ ”、“^”。</li> </ul>
7	BIT[(M)]	不支持	-

### 8.1.1.2.5 JSON 数据类型

表 8-8 JSON 数据类型

序号	MySQL数据库	GaussDB数据库
1	JSON	不完全兼容



### 8.1.1.2.6 数据类型支持的属性

表 8-9 数据类型支持的属性

序号	MySQL数据库	GaussDB数据库
1	NULL	支持
2	NOT NULL	支持
3	DEFAULT	支持
4	ON UPDATE	支持
5	PRIMARY KEY	支持
6	CHARACTER SET name	支持
7	COLLATE name	支持

### 8.1.1.2.7 数据类型转换

不同的数据类型之间支持转换。有如下场景涉及到数据类型转换：

- 操作符（比较操作符、运算操作符等）的操作数的数据类型不一致。常见于查询条件或者关联条件中的比较运算。
- 函数调用时实参和形参的数据类型不一致。
- DML语句要更新（包括insert、update、merge、replace等）的目标列，数据的类型和列的定义类型不一致。
- 显式的类型转换：cast(expr as datatype)，将expr表达式类型转换为datatype类型。
- 集合运算（UNION、MINUS、EXCEPT、INTERSECT）确定最终投影列的目标数据类型后，各个SELECT查询的投影列的类型和目标数据类型不一致。
- 其他表达式计算场景，根据不同表达式的数据类型，来决定用于比较或者最终结果的目标数据类型。
  - DECODE
  - CASE WHEN
  - lexpr [ NOT ] IN (expr\_list)
  - BETWEEN AND
  - JOIN USING(a,b)
  - GREATEST和LEAST
  - NVL 和 COALESCE

GaussDB和MySQL数据库对于数据类型转换、转换的目标数据类型有着完全不同的规则。如下示例体现了两者处理的差异：

```
-- MySQL: in 执行结果为0，表示false。根据规则，会将'1970-01-01'与列表中的表达式依次比较，结果都为0，因此最终结果为0。
mysql> select '1970-01-01' in ('1970-01-02', 1, '1970-01-02');
+-----+
| '1970-01-01' in ('1970-01-02', 1, '1970-01-02') |
```

```
+-----+
|                0 |
+-----+

-- GaussDB: in 执行结果为true, 与MySQL结果相反: 根据规则选定的公共类型为int类型, 因此将左表达式
'1970-01-01'转换为int类型与列表中的表达式转换为int类型后的值依次比较。
-- '1970-01-01'和'1970-01-02'转换为int类型时都为1970 (兼容MySQL模式下, 转换时遇到非法字符后忽略, 将
前面部分转换为int类型), 比较结果为相等, 因此返回结果为true。
gaussdb=# select '1970-01-01' in ('1970-01-02', 1::int, '1970-01-02') as result;
result
-----
t
(1 row)
```

### 1. 数据类型转换规则的差异:

- GaussDB数据库对于不同数据类型之间的转换规则有明确的定义:
  - 是否支持转换: pg\_cast系统表中是否定义两种类型的转换路径, 没有定义则不支持。
  - 支持转换的场景: 支持任意场景转换、仅支持显式 ( cast表达式 ) 转换、仅支持赋值时转换。不支持的场景下即使定义了转换路径, 也不能做数据类型转换。
- MySQL数据库支持任意两种数据类型之间做转换。

由于存在以上差异, 基于MySQL数据库的应用程序向GaussDB数据库迁移时, SQL语句可能由于不支持不同数据类型之间的转换而报错。或者支持转换的场景下, 转换的规则有差异导致SQL语句执行的结果不同。

推荐的做法是: SQL语句中尽量使用相同的数据类型做比较或者赋值等操作, 避免因数据类型转换导致非预期结果或者性能损耗。

### 2. 选择目标数据类型的规则差异:

对于有些场景, 比较的数据类型或者返回的数据类型需要综合考虑多个表达式的类型才能确定。比如UNION运算中, 不同SELECT语句中相同位置的投影列具有不同的数据类型, 查询结果的最终数据类型, 需要由各个SELECT语句投影列的数据类型共同确定。

确定目标数据类型的规则, GaussDB数据库和MySQL数据库存在体系上的差异。

- GaussDB数据库规则:
  - 操作符的操作数类型不一致时, 并不是将操作数的类型统一转换为目标类型再计算。而是直接注册两个数据类型的操作符, 操作符处理中定义两个不同类型的处理规则。此方式不存在类型隐式转换, 但自定义的处理规则隐含了转换的操作。
  - 集合运算和表达式场景, 确定目标数据类型的规则:
    - 如果所有类型都相同, 则此类型即为目标类型。
    - 两个数据类型如果不同, 检查数据类型是否属于同一种类的数据类型, 如数值类型、字符类型、日期时间类型等。不属于同一种类的数据类型, 无法确定目标类型, 此时SQL语句执行会报错。
    - 对于category属性 ( 在pg\_type系统表中定义 ) 相同的数据类型, 具有preferred属性 ( 在pg\_type系统表中定义 ) 的数据类型会被选为目标类型。或者操作数1能转换为操作数2 ( 没有转换路径 ), 而操作数2无法转换为操作数1或数值类型优先级小于操作数2, 则选择操作数2作为目标类型。

- 如果涉及到3个及以上的数据类型，确定目标类型的规则为：  
 $\text{common\_type}(\text{type1}, \text{type2}, \text{type3}) = \text{common\_type}(\text{common\_type}(\text{type1}, \text{type2}), \text{type3})$ ，依次迭代处理，得到最终的结果。
- 对于IN和NOT IN表达式，如果根据以上规则无法确认目标类型，会将lexpr与expr\_list中每一个表达式单独按照等值操作符(=)逐个比较。
- 精度的确定：以最终选定的表达式的精度作为最终结果。
- MySQL数据库规则：
  - 操作符的操作数类型不一致时，先按照如下规则确定目标类型。确定后将类型不一致的操作数转换成目标类型后再做处理。
    - 两个参数都是string类型，则都按照string类型比较。
    - 两个参数都是integer类型，则都按照integer类型比较。
    - 十六进制数值如果不与数值比较，则当做二进制字符串比较。
    - 一个参数是datetime/timestamp类型，另一个参数是常量，将常量转换为时间戳类型然后比较。
    - 如果其中一个参数是decimal类型，比较时使用的数据类型取决于另外一个参数。另外一个是decimal或者integer类型时，按照decimal类型；另外一个是其他类型，按照real类型比较。
    - 其他场景都转换为 real 类型后比较。
  - 集合运算和表达式场景，确定目标数据类型的规则如下：
    - 建立任意两个类型之间的目标类型矩阵。给定两个类型，通过矩阵即可以确定目标类型。
    - 如果涉及到3个及以上的数据类型，确定目标类型的规则为：  
 $\text{common\_type}(\text{type1}, \text{type2}, \text{type3}) = \text{common\_type}(\text{common\_type}(\text{type1}, \text{type2}), \text{type3})$ ，依次迭代处理，得到最终的结果。
    - 如果目标类型是integer类型，且各个表达式类型包含有符号和无符号的混合场景，则会将类型提升到更高精度的integer类型。符号的确定：所有表达式都是无符号时，结果才为无符号，否则结果为有符号。
    - 精度确定：以表达式中的最大精度作为最终结果。

从以上规则可知：GaussDB和MySQL数据库在数据类型的转换规则上有很大差异，不能直接对比。在上述场景下，SQL语句的执行结果可能和MySQL数据库不一致。当前版本推荐各个表达式使用相同的类型，或提前使用cast转换成需要的类型来规避差异。

### 8.1.1.3 系统函数

GaussDB数据库兼容绝大多数MySQL的系统函数，但存在部分差异。如未列出，函数行为默认为GaussDB原生行为。

### 8.1.1.3.1 流量控制函数

表 8-10 流量控制函数列表

序号	MySQL 数据库	GaussDB 数据库	差异
1	IF()	支持	<ul style="list-style-type: none"> <li>expr1 入参仅支持 bool 类型。非 bool 类型入参若不能转换为 bool 类型则报错。</li> <li>若 expr2、expr3 两入参类型不同且两类型间不存在隐式转换函数则报错。</li> <li>两入参类型相同时，返回该入参类型。</li> <li>若 expr2、expr3 两入参类型分别为 NUMERIC、STRING 或 TIME 其中一个时，输出为 TEXT 类型，MySQL 输出为 VARCHAR 类型。</li> </ul>
2	IFNULL()	支持	<ul style="list-style-type: none"> <li>若 expr1、expr2 两入参类型不同且两类型间不存在隐式转换函数则报错。</li> <li>两入参类型相同时，返回该入参类型。</li> <li>若 expr1、expr2 两入参类型范畴分别为 NUMERIC、STRING 或 TIME 其中一个时，输出为 TEXT 类型，MySQL 输出为 VARCHAR 类型。</li> <li>两入参类型其中一个为 FLOAT4 类型另一个为 numeric 范畴中任一类型，返回值为 DOUBLE 类型。MySQL 其中一入参为 FLOAT4，另一入参为 TINYINT、UNSIGNED TINYINT、SMALLINT、UNSIGNED SMALLINT、MEDIUMINT、UNSIGNED MEDIUMINT、BOOL 任一类型时，返回 FLOAT4 类型，第一个入参为 FLOAT4，第二个入参为 BIGINT 或 UNSIGNED BIGINT 时，返回 FLOAT 类型。</li> </ul>

序号	MySQL 数据库	GaussDB 数据库	差异
3	NULLIF()	支持	<ul style="list-style-type: none"> <li>GaussDB中NULLIF()类型推导遵从以下逻辑： <ol style="list-style-type: none"> <li>如果两个参数的数据类型不同，且两入参类型存在等值比较操作符，则返回对应等值操作符对应的左值类型，否则会对两入参类型进行强制类型兼容。</li> <li>若强制类型兼容后，存在等值比较操作符，则返回强制类型兼容后对应等值操作符的左值类型。</li> <li>若强制类型兼容后，仍找不到对应等值操作符，则报错。  <pre>--两入参类型存在等值比较操作符 gaussdb=# select pg_typeof(nullif(1::int2, 2::int8)); pg_typeof ----- smallint (1 row) --两入参类型不存在等值比较操作符，但在强制类型兼容后可以找到等值比较操作符 gaussdb=# select pg_typeof(nullif(1::int1, 2::int2)); pg_typeof ----- bigint (1 row)  --两入参类型不存在等值比较操作符，且强制类型兼容后也不存在等值比较操作符 gaussdb=# SELECT nullif(1::bit, '1'::MONEY); ERROR: operator does not exist: bit = money LINE 1: SELECT nullif(1::bit, '1'::MONEY);           ^ HINT: No operator matches the given name and argument type(s). You might need to add explicit type casts. CONTEXT: referenced column: nullif</pre> </li> </ol> </li> <li>MySQL输出类型仅与第一个入参类型有关： <ol style="list-style-type: none"> <li>第一个入参为tinyint、smallint、mediumint、int、bool时，输出为int类型。</li> <li>第一个入参为bigint时，输出为bigint类型。</li> <li>第一个入参为unsigned tinyint、unsigned smallint、unsigned mediumint、unsigned int、bit时，输出为unsigned int类型。</li> <li>第一个入参为unsigned bigint时，输出为unsigned bigint。</li> <li>第一个入参为浮点型即float、double、real时，输出为double类型。</li> <li>第一个入参类型为decimal或numeric类型时，输出为decimal类型。</li> <li>第一个入参类型为时间类型或字符串类型即date、time、date、datetime、timestamp、char、varchar以及tinytext、enum、set时，输出为varchar类型。</li> <li>第一个入参类型为text、mediumtext、longtext时，输出为longtext类型。</li> </ol> </li> </ul>

序号	MySQL 数据库	GaussDB 数据库	差异
			9. 第一个入参类型为tinyblob时，输出为varbinary类型。 10. 第一个入参类型为mediumblob或longblob时，输出为longblob类型。 11. 第一个入参为blob时，输出为blob类型。
4	ISNULL()	支持	GaussDB中返回值为BOOLEAN类型的t或f，MySQL中返回值为INT类型的1或0。

### 8.1.1.3.2 日期和时间函数

#### 📖 说明

以下为GaussDB MySQL兼容性日期时间函数公共说明。

- 开发者指南中函数入参为时间类型表达式的情况：

时间类型表达式主要包括text、datetime、date或time，但所有可以隐式转换为时间表达式的类型都可以作为入参，比如数字类型可以通过先隐式转化为text，再作为时间类型表达式生效。

但是生效的情况根据函数有所不同，比如：datediff由于只计算日期的差值，所以时间表达式会当做日期来解析。timestampdiff由于计算时间差值时是根据unit来决定的，所以会根据unit选择将时间表达式当做date或者time或者datetime来解析。

- 函数入参为无效日期的情况：

一般而言，日期时间函数支持date、datetime的范围和mysql保持一致。date支持的范围为'0000-01-01'到'9999-12-31'，datetime支持的范围为'0000-01-01 00:00:00'到'9999-12-31 23:59:59'。虽然GaussDB支持的date、datetime范围大于mysql，但是越界仍然算无效日期。大部分时间函数会告警并返回NULL，只有能通过cast正常转换的日期，才是正常合理的日期。

GaussDB的日期时间函数的大部分功能场景与MySQL一致，但仍有差异，一些差异如下：

- 函数入参为NULL时，函数返回NULL，无warning或error告警。这些函数包括：from\_days、date\_format、str\_to\_date、datediff、timestampdiff、date\_add、subtime、month、time\_to\_sec、to\_days、to\_seconds、dayname、monthname、convert\_tz、sec\_to\_time、addtime、adddate、date\_sub、timediff、last\_day、weekday、from\_unixtime、unix\_timestamp、subdate、day、year、weekofyear、dayofmonth、dayofyear、week、yearweek、dayofweek、time\_format、hour、minute、second、microsecond、quarter、get\_format、extract、makedate、period\_add、timestampadd、period\_diff、utc\_time、utc\_timestamp、maketime、curtime

示例：

```
gaussdb=# select day(null);
 day
-----
(1 row)
```

- 纯数字入参个别函数与MySQL有差异，不带引号的数字入参统一转成text入参来处理。

示例：

```
gaussdb=# select day(19231221.123141);
WARNING: Incorrect datetime value: "19231221.123141"
```

```
CONTEXT: referenced column: day
day
-----
```

```
(1 row)
```

- 时间日期运算函数：adddate、subdate、date\_add、date\_sub。当运算后的日期结果在范围[0000-01-01, 0001-01-01]中时，正常返回结果，MySQL返回'0000-00-00'。

示例：

```
gaussdb=# select subdate('0001-01-01', interval 1 day);
subdate
```

```
-----
0000-12-31
```

```
(1 row)
```

- 对于日期和时间函数的date或datetime类型入参，含有0月或0日则为非法值，在严格模式下报error；在宽松模式，当输入为字符串或数字时，报warning，输入为date或datetime类型时视为上一年12月或上一月最后一日处理。

对于cast函数，转换为date、datetime时，严格模式下会报error；宽松模式下不会报warning，而是视为上一年12月或上一月最后一日处理，需要注意此区别。MySQL对于包含0年、0月或0日的情况会原样输出。

示例：

```
gaussdb=# select adddate('2023-01-00', 1);-- 严格模式
ERROR: Incorrect datetime value: "2023-01-00"
CONTEXT: referenced column: adddate
```

```
gaussdb=# select adddate('2023-01-00', 1);-- 宽松模式
WARNING: Incorrect datetime value: "2023-01-00"
CONTEXT: referenced column: adddate
adddate
```

```
-----
(1 row)
```

```
gaussdb=# select adddate(date'2023-00-00', 1);-- 宽松模式
adddate
```

```
-----
2022-12-01
(1 row)
```

```
gaussdb=# select cast('2023/00/00' as date);-- 宽松模式
date
```

```
-----
2022-11-30
(1 row)
```

```
gaussdb=# select cast('0000-00-00' as datetime);-- 宽松模式
timestamp
```

```
-----
0000-00-00 00:00:00
(1 row)
```

- 若函数入参为numeric数据类型，在非法输入的情况下不会产生报错，会把入参当做0值处理。

示例：

```
gaussdb=# select from_unixtime('aa');
from_unixtime
```

```
-----
1970-01-01 08:00:00
(1 row)
```

- 最多保留6位小数，不保留后置都为0的小数。

示例:

```
gaussdb=# select from_unixtime('1234567899.00000');
      from_unixtime
-----
2009-02-14 07:31:39
(1 row)
```

- 时间函数参数为字符串时，只保证年月日之间使用“-”分隔，时分秒之间使用“:”分隔时结果正确。

示例:

```
gaussdb=# select adddate('20-12-12',interval 1 day);
      adddate
-----
2020-12-13
(1 row)
```

- 在MySQL中，当函数的返回值为varchar时，在GaussDB中，函数对应的返回值为text。

```
-- GaussDB中函数的返回值。
gaussdb=# SELECT pg_typeof(adedate('2023-01-01', 1));
      pg_typeof
-----
text
(1 row)

-- MySQL中函数的返回值。
mysql> CREATE VIEW v1 AS SELECT adddate('2023-01-01', 1);
Query OK, 0 rows affected (0.00 sec)

mysql> DESC v1;
+-----+-----+-----+-----+-----+-----+
| Field          | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| adddate('2023-01-01', 1) | varchar(29) | YES  |     | NULL    |      |
+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

表 8-11 日期与和时间函数列表

序号	MySQL数据库	GaussDB数据库	差异
1	ADDDATE()	支持	此函数的表现会因为interval表达式的差异与MySQL有差异，具体可见 <a href="#">INTERVAL差异说明</a> 。



序号	MySQL数据库	GaussDB数据库	差异
2	ADDTIME()	支持	<ul style="list-style-type: none"> <li>● MySQL对第二入参为DATETIME样式字符串返回NULL，GaussDB可以计算。</li> <li>● 入参取值范围为['0001-01-01 00:00:00', 9999-12-31 23:59:59.999999]。</li> <li>● MySQL中ADDTIME函数如果第一个参数是动态参数（例如在预准备语句中），则返回类型为TIME。否则，函数的解析类型派生自第一个参数的解析类型。GaussDB中ADDTIME函数的返回值规则如下： <ul style="list-style-type: none"> <li>- 第一个入参为date，第二个入参为date，返回值为time。</li> <li>- 第一个入参为date，第二个入参为text，返回值为text。</li> <li>- 第一个入参为date，第二个入参为datetime，返回值为time。</li> <li>- 第一个入参为date，第二个入参为time，返回值为time。</li> <li>- 第一个入参为text，第二个入参为date，返回值为text。</li> <li>- 第一个入参为text，第二个入参为text，返回值为text。</li> <li>- 第一个入参为text，第二个入参为datetime，返回值为text。</li> <li>- 第一个入参为text，第二个入参为time，返回值为text。</li> <li>- 第一个入参为datetime，第二个入参为date，返回值为datetime。</li> <li>- 第一个入参为datetime，第二个入参为text，返回值为text。</li> <li>- 第一个入参为datetime，第二个入参为datetime，返回值为datetime。</li> <li>- 第一个入参为datetime，第二个入参为time，返回值为datetime。</li> <li>- 第一个入参为time，第二个入参为date，返回值为time。</li> </ul> </li> </ul>

序号	MySQL数据库	GaussDB数据库	差异
			<ul style="list-style-type: none"> <li>- 第一个入参为time，第二个入参为text，返回值为text。</li> <li>- 第一个入参为time，第二个入参为datetime，返回值为time。</li> <li>- 第一个入参为time，第二个入参为time，返回值为time。</li> </ul>
3	CONVERT_TZ()	支持	-
4	CURDATE()	支持	-
5	CURRENT_DATE(), CURRENT_DATE	支持	-
6	CURRENT_TIME(), CURRENT_TIME	支持	<p>GaussDB的按精度输出的时间值（小数点后的值）是四舍五入的；MySQL是直接截断的。GaussDB按精度输出的时间值（小数点后的值）末尾0都不显示；MySQL会显示。GaussDB只支持输入[0,6]范围内的整型值，作为返回时间的精度，其他均报错；MySQL的精度值有效值是[0,6]，但是输入的整型值内部会对256求余（例257，会返回精度1的时间值）。</p>
7	CURRENT_TIMESTAM P(), CURRENT_TIMESTAMP	支持	<p>GaussDB的按精度输出的时间值（小数点后的值）是四舍五入的；MySQL是直接截断的。GaussDB按精度输出的时间值（小数点后的值）末尾0都不显示；MySQL会显示。GaussDB只支持输入[0,6]范围内的整型值，作为返回时间的精度，超过6的整型值，会告警并按照精度6输出时间值；MySQL的精度值有效值是[0,6]，但是输入的整型值内部会对256求余（例257，会返回精度1的时间值）。</p>

序号	MySQL数据库	GaussDB数据库	差异
8	CURTIME()	支持	GaussDB此函数输入字符串或者非整型值，会被隐式转成整型，然后再校验精度，[0,6]范围之外的会报错，范围之内会正常输出时间值；MySQL直接报错。GaussDB的按精度输出的时间值（小数点后的值）是四舍五入的；MySQL是直接截断的。GaussDB按精度输出的时间值（小数点后的值）末尾0都不显示；MySQL会显示。GaussDB只支持输入[0,6]范围内的整型值，作为返回时间的精度，其他均报错；MySQL的精度值有效值是[0,6]，但是输入的整型值内部会对256求余（例257，会返回精度1的时间值）。
9	YEARWEEK()	支持	-
10	DATE_ADD()	支持	此函数的表现会因为interval表达式的差异与MySQL有差异，具体可见 <a href="#">INTERVAL差异说明</a> 。
11	DATE_FORMAT()	支持	-
12	DATE_SUB()	支持	此函数的表现会因为interval表达式的差异与MySQL有差异，具体可见 <a href="#">INTERVAL差异说明</a> 。
13	DATEDIFF()	支持	-
14	DAY()	支持	-
15	DAYNAME()	支持	-
16	DAYOFMONTH()	支持	-
17	DAYOFWEEK()	支持	-
18	DAYOFYEAR()	支持	-
19	EXTRACT()	支持	-
20	FROM_DAYS()	支持	-
21	FROM_UNIXTIME()	支持	-
22	GET_FORMAT()	支持	-
23	HOUR()	支持	-
24	LAST_DAY	支持	-

序号	MySQL数据库	GaussDB数据库	差异
25	LOCALTIME(), LOCALTIME	支持	GaussDB的按精度输出的时间值（小数点后的值）是四舍五入的；MySQL是直接截断的。GaussDB按精度输出的时间值（小数点后的值）末尾0都不显示；MySQL会显示。GaussDB只支持输入[0,6]范围内的整型值，作为返回时间的精度，其他整型值直接报错；MySQL的精度值有效值是[0,6]，但是输入的整型值内部会对256求余（例257，会返回精度1的时间值）。
26	LOCALTIMESTAMP, LOCALTIMESTAMP()	支持	GaussDB的按精度输出的时间值（小数点后的值）是四舍五入的；MySQL是直接截断的。GaussDB按精度输出的时间值（小数点后的值）末尾0都不显示；MySQL会显示。GaussDB只支持输入[0,6]范围内的整型值，作为返回时间的精度，超过6的整型值，会告警并按照精度6输出时间值；MySQL的精度值有效值是[0,6]，但是输入的整型值内部会对256求余（例257，会返回精度1的时间值）。
27	MAKEDATE()	支持	-
28	MAKETIME()	支持	与MySQL相比, 入参为NULL时, GaussDB不支持maketime函数自嵌套, MySQL支持。
29	MICROSECOND()	支持	-
30	MINUTE()	支持	-
31	MONTH()	支持	-
32	MONTHNAME()	支持	-
33	NOW()	支持	GaussDB的按精度输出的时间值（小数点后的值）是四舍五入的；MySQL是直接截断的。GaussDB按精度输出的时间值（小数点后的值）末尾0都不显示；MySQL会显示。GaussDB只支持输入[0,6]范围内的整型值，作为返回时间的精度，超过6的整型值，会告警并按照精度6输出时间值；MySQL的精度值有效值是[0,6]，但是输入的整型值内部会对256求余（例257，会返回精度1的时间值）。

序号	MySQL数据库	GaussDB数据库	差异
34	PERIOD_ADD()	支持	当入参period或结果小于0时，GaussDB参考MySQL8.0.x版本的表现，报错处理。MySQL5.7会发生整数回绕，导致计算结果异常。
35	PERIOD_DIFF()	支持	当入参或结果小于0时，GaussDB参考MySQL8.0.x版本的表现，报错处理。MySQL5.7会发生整数回绕，导致计算结果异常。
36	QUARTER()	支持	-
37	SEC_TO_TIME()	支持	-
38	SECOND()	支持	-
39	STR_TO_DATE()	支持	返回值与MySQL有差异，GaussDB返回的是text，MySQL返回的是datetime、date。
40	SUBDATE()	支持	此函数的表现会因为interval表达式的差异与MySQL有差异，具体可见 <a href="#">INTERVAL差异说明</a> 。

序号	MySQL数据库	GaussDB数据库	差异
41	SUBTIME()	支持	<ul style="list-style-type: none"> <li>• MySQL对第二入参为DATETIME样式字符串返回NULL，GaussDB可以计算</li> <li>• 入参取值范围为['0001-01-01 00:00:00', 9999-12-31 23:59:59.999999]。</li> <li>• MySQL中SUBTIME函数如果第一个参数是动态参数（例如在预准备语句中），则返回类型为TIME。否则，函数的解析类型派生自第一个参数的解析类型。GaussDB中SUBTIME函数的返回值规则如下： <ul style="list-style-type: none"> <li>- 第一个入参为date，第二个入参为date，返回值为time。</li> <li>- 第一个入参为date，第二个入参为text，返回值为text。</li> <li>- 第一个入参为date，第二个入参为datetime，返回值为time。</li> <li>- 第一个入参为date，第二个入参为time，返回值为time。</li> <li>- 第一个入参为text，第二个入参为date，返回值为text。</li> <li>- 第一个入参为text，第二个入参为text，返回值为text。</li> <li>- 第一个入参为text，第二个入参为datetime，返回值为text。</li> <li>- 第一个入参为text，第二个入参为time，返回值为text。</li> <li>- 第一个入参为datetime，第二个入参为date，返回值为datetime。</li> <li>- 第一个入参为datetime，第二个入参为text，返回值为text。</li> <li>- 第一个入参为datetime，第二个入参为datetime，返回值为datetime。</li> <li>- 第一个入参为datetime，第二个入参为time，返回值为datetime。</li> <li>- 第一个入参为time，第二个入参为date，返回值为time。</li> </ul> </li> </ul>

序号	MySQL数据库	GaussDB数据库	差异
			<ul style="list-style-type: none"> <li>- 第一个入参为time，第二个入参为text，返回值为text。</li> <li>- 第一个入参为time，第二个入参为datetime，返回值为time。</li> <li>- 第一个入参为time，第二个入参为time，返回值为time。</li> </ul>
42	SYSDATE()	支持	MySQL入参整型值会按照一字节最大值255整数回绕，Gauss不回绕。
43	YEAR()	支持	-
44	TIME_FORMAT()	支持	-
45	TIME_TO_SEC()	支持	-
46	TIMEDIFF()	支持	-
47	WEEKOFYEAR()	支持	-
48	TIMESTAMPADD()	支持	-
49	TIMESTAMPDIFF()	支持	-
50	TO_DAYS()	支持	-
51	TO_SECONDS()	支持	-
52	UNIX_TIMESTAMP()	支持	返回值与MySQL有差异，GaussDB返回的是numeric，MySQL返回的是int。
53	UTC_DATE()	支持	MySQL支持无括号调用，GaussDB不支持。MySQL入参整型值会按照一字节最大值255整数回绕。
54	UTC_TIME()	支持	
55	UTC_TIMESTAMP()	支持	
56	WEEK()	支持	-
57	WEEKDAY()	支持	-

## 8.1.1.3.3 字符串函数

表 8-12 字符串函数列表

序号	MySQL数据库	GaussDB数据库	差异
1	BIN()	支持	函数入参1支持类型存在差异，GaussDB入参1支持类型如下： <ul style="list-style-type: none"><li>• 整数类型：tinyint、smallint、mediumint、int、bigint</li><li>• 无符号整数类型：tinyint unsigned、smallint unsigned、int unsigned、bigint unsigned</li><li>• 字符和文本类型：char, varchar、tinytext, text, mediumtext, longtext, 仅支持纯数字整数字符串，且整数范围在bigint范围内。</li><li>• 浮点类型：float、real、double</li><li>• 定点类型：numeric、decimal、dec</li><li>• 布尔类型：bool</li></ul>
2	CONCAT()	支持	无论参数的数据类型如何，concat返回值的数据类型始终为text；MySQL的concat在含有二进制类型参数时，返回值为二进制类型。
3	CONCAT_WS()	支持	无论参数的数据类型如何，concat_ws返回值的数据类型始终为text；MySQL的concat_ws在含有二进制类型参数时，返回值为二进制类型，其他情况返回值为字符串类型。



序号	MySQL数据库	GaussDB数据库	差异
4	ELT()	支持	<ul style="list-style-type: none"> <li>函数入参1支持类型存在差异，GaussDB入参1支持类型如下：                      整数类型：tinyint、smallint、mediumint、int、bigint                      无符号整数类型：tinyint unsigned、smallint unsigned、int unsigned                      字符和文本类型：char，varchar、tinytext，text，mediumtext，longtext，仅支持纯数字整数字符串，且整数范围在bigint范围内。                      浮点类型：float、real、double                      定点类型：numeric、decimal、dec                      布尔类型：bool</li> <li>函数入参2支持类型存在差异，GaussDB入参2支持类型如下：                      整数类型：tinyint、smallint、mediumint、int、bigint                      无符号整数类型：tinyint unsigned、smallint unsigned、int unsigned、bigint unsigned                      字符和文本类型：char，varchar、tinytext，text，mediumtext，longtext                      浮点类型：float、real、double                      定点类型：numeric、decimal、dec                      布尔类型：bool                      大对象类型：tinyblob，blob，mediumblob，longblob                      日期类型：datetime，timestamp，date，time</li> </ul>
5	FIELD	支持	函数入参在bigint最大值 ~ bigint unsigned最大值范围内的数字，存在不兼容。 函数入参为浮点型float(m, d)、double(m, d)、real(m, d)时精度更高，存在不兼容。
6	FIND_IN_SET()	支持	当数据库encoding = 'SQL_ASCII'时，不支持默认的大小写判断规则，即在用户不指定字符集规则的情况下，大写与小写区分判断。

序号	MySQL数据库	GaussDB数据库	差异
7	INSERT()	支持	<ul style="list-style-type: none"><li>• Int64类型传参有范围限制，一旦超出-9223372036854775808~9223372036854775807范围会直接报错，MySQL对数值类型传参范围无限制，异常会告警按照上限或下限数值处理。</li><li>• 字符串传参有限制，入参text类型字符串长度最大为2<sup>30</sup>-5字节，入参bytea类型字符串长度最大为2<sup>30</sup>-512字节。</li><li>• s1和s2任意参数为bytea类型时，涉及到结果出现非法字符的情况可能展示结果与MySQL有差异但是字符编码与MySQL是一致的。</li></ul>
8	LOCATE()	支持	入参1为bytea类型，入参2为text类型时，GaussDB与MySQL行为存在差异。
9	MAKE_SET()	支持	<ul style="list-style-type: none"><li>• bits参数为整型时，最大范围支持到int128，低于MySQL范围。</li><li>• bits参数为日期类型datetime、timestamp、date、time，由于时间类型转整型与MySQL存在差异，目前均未做支持。</li><li>• bit类型或bool类型由于此类数据类型Gauss与MySQL存在差异，返回结果导致的差异为Gauss与MySQL固有差异。bits入参为bool类型，str入参为bit类型与bool类型均不做支持。</li><li>• bits入参为字符串或文本类型时，仅支持纯整型数字形式，其他形式存在差异。且纯整型数字范围限制在bigint范围。</li><li>• str入参整型数值超过正负81个9，返回值与MySQL有差异。</li><li>• str入参当以科学计数法表示时，GaussDB末尾0值会显示，MySQL不显示，以科学计数法打印，此为固有差异。</li></ul>

序号	MySQL数据库	GaussDB数据库	差异
10	QUOTE()	支持	<ul style="list-style-type: none"> <li>已知str字符串中含有“\Z”，“\r”，“\%”，“\_”，GaussDB未进行转义，与MySQL存在差异。斜线后跟部分数字也会引起差异，如“\563”。由转义字符引起的本函数与MySQL的差异，此为Gauss与MySQL的转义字符差异。</li> <li>str字符串中的“\b”，输出结果表现形式与MySQL有差异。此为Gauss与MySQL的固有差异</li> <li>str字符串中含有“\0”时，GaussDB由于UTF-8字符集不识别该字符，输入不成功。此为GaussDB与MySQL的固有差异</li> <li>str为bit或bool类型时，由于Gauss与MySQL此类型目前有差异，暂不支持此类类型。</li> <li>Gauss最大支持1GB数据传输，str入参长度最大支持536870908，函数返回结果字符串最大支持1GB。</li> <li>str入参整型数值超过正负81个9，返回值与MySQL有差异。</li> <li>str入参当以科学计数法表示时，GaussDB末尾0值会显示，MySQL不显示，以科学计数法打印，此为固有差异。</li> </ul>
11	SPACE()	支持	<ol style="list-style-type: none"> <li>GaussDB入参1最大支持1073741818，超出返回空字符串。MySQL的入参1默认最大支持4194304，超出告警。</li> <li>函数入参1支持类型存在差异，GaussDB入参1支持类型如下： <ul style="list-style-type: none"> <li>整数类型：tinyint、smallint、mediumint、int、bigint</li> <li>无符号整数类型：tinyint unsigned、smallint unsigned、int unsigned</li> <li>字符和文本类型：char，varchar、tinytext，text，mediumtext，longtext，仅支持纯数字整数字符串，且整数范围在bigint范围内。</li> <li>浮点类型：float、real、double</li> <li>定点类型：numeric、decimal、dec</li> <li>布尔类型：bool</li> </ul> </li> </ol>
12	SUBSTR()	支持	-

序号	MySQL数据库	GaussDB数据库	差异
13	SUBSTRING()	支持	-
14	SUBSTRING_INDEX()	支持	-
15	STRCMP()	支持	<p>1. 支持入参类型与M存在差异，GaussDB支持类型如下：</p> <ul style="list-style-type: none"> <li>• 字符类型：CHAR、VARCHAR、NVARCHAR2、TEXT</li> <li>• 二进制类型：BYTEA</li> <li>• 数值类型：TINYINT [UNSIGNED]、SMALLINT [UNSIGNED]、INTEGER [UNSIGNED]、BIGINT [UNSIGNED]、FLOAT4、FLOAT8、NUMERIC</li> <li>• 日期时间类型：DATE、TIME WITHOUT TIME ZONE、DATETIME、TIMESTAMPZ</li> </ul> <p>2. 对于数值类型中的浮点类型，由于连接参数设置不同，精度可能与M有差异，不建议使用该场景，或使用NUMERIC类型代替。</p>
16	SHA()/SHA1()	支持	-
17	SHA2()	支持	-

#### 8.1.1.3.4 强制转换函数

表 8-13 强制转换函数列表

序号	MySQL数据库	GaussDB数据库	差异
1	CAST()	支持	数据类型转换规则和支持的转换类型均以 GaussDB支持的转换范围和规则为准。
2	CONVERT()	支持	数据类型转换规则和支持的转换类型均以 GaussDB支持的转换范围和规则为准。



序号	MySQL数据库	GaussDB数据库	差异
7	JSON_DEPTH()	支持	返回值与MySQL有差异，GaussDB返回的是int，MySQL返回的是bigint。
8	JSON_EXTRACT()	支持	-
9	JSON_INSERT()	支持	-
10	JSON_KEYS()	支持	-
11	JSON_LENGTH()	支持	返回值与MySQL有差异，GaussDB返回的是int，MySQL返回的是bigint。
12	JSON_MERGE()	支持	-
13	JSON_OBJECT()	支持	-
14	JSON_QUOTE()	支持	返回值与MySQL有差异，GaussDB返回的是json，MySQL返回的是varchar或者text。
15	JSON_REMOVE()	支持	-
16	JSON_REPLACE()	支持	-
17	JSON_SEARCH()	支持	返回值与MySQL有差异，GaussDB返回的是text，MySQL返回的是json。
18	JSON_SET()	支持	-
19	JSON_TYPE()	支持	数值类型的json值统一识别为number，与MySQL有差异。

序号	MySQL数据库	GaussDB数据库	差异
20	JSON_UNQUOTE()	支持	-
21	JSON_VALID()	支持	-

### 8.1.1.3.7 聚合函数

表 8-16 聚合函数列表

序号	MySQL数据库	GaussDB数据库	差异
1	GROUP_CONCAT()	支持	<ol style="list-style-type: none"> <li>1. 当group_concat参数中同时有DISTINCT和ORDER BY语法时，所有ORDER BY后的表达式必须也在DISTINCT的表达式之中。</li> <li>2. group_concat(... order by 数字)不代表按照第几个参数的顺序，数字只是一个常量表达式，相当于不排序。</li> <li>3. 无论参数的数据类型如何，group_concat返回值的数据类型始终为text；MySQL的group_concat在含有二进制类型参数时，返回值为二进制类型，其他情况返回值为字符串类型，并且返回值长度大于512时，其数据类型为字符串大对象或二进制大对象。</li> <li>4. GUC参数 group_concat_max_len有效范围是0-1073741823，最大值比MySQL小。</li> </ol>

序号	MySQL数据库	GaussDB数据库	差异
2	DEFAULT()	支持	<ol style="list-style-type: none"><li>1. 字段默认值为数组形式，GaussDB返回数组形式，MySQL不支持数组类型。</li><li>2. GaussDB字段是隐藏列（比如xmin、cmin），函数返回空值。</li><li>3. GaussDB支持分区表、临时表、多表连接查询默认值。</li><li>4. GaussDB支持查询列名包含字符串值节点（表示名称）和A_Star节点（表示出现“*”），如default(tt.t4.id)和default(tt.t4.*）。不合法的查询列名和A_Star节点，GaussDB和MySQL报错信息有差异。</li><li>5. GaussDB创建字段默认值，没有检验字段类型的范围，使用default函数可能报错。</li><li>6. 字段的默认值是函数表达式时，GaussDB的default函数返回建表时字段的default表达式的计算值。MySQL的default函数返回NULL。</li></ol>



### 8.1.1.3.8 数字操作函数

表 8-17 数字操作函数列表

序号	MySQL数据库	GaussDB数据库	差异
1	log2()	支持	<ol style="list-style-type: none"> <li>1. 小数位显示与MySQL存在差异，受GaussDB浮点数据类型限制，可通过参数extra_float_digits控制小数位个数显示。</li> <li>2. 由于输入精度内部处理差异，GaussDB与MySQL会存在结果计算差异。</li> <li>3. 支持数据类型有： <ul style="list-style-type: none"> <li>• bigint、int16、int、smallint、tinyint整数类型。</li> <li>• bigint unsigned、integer unsigned、smallint unsigned、tinyint unsigned无符号整数类型。</li> <li>• numeric、real浮点数类型。</li> <li>• character、character varying、clob、text字符串类型，仅支持纯数字整数字符串。</li> <li>• set类型。</li> <li>• NULL空类型。</li> </ul> </li> </ol>
2	log10()	支持	<ol style="list-style-type: none"> <li>1. 小数位显示与MySQL存在差异，受GaussDB浮点数据类型限制，可通过参数extra_float_digits控制小数位个数显示；</li> <li>2. 由于输入精度内部处理差异，GaussDB与MySQL会存在结果计算差异；</li> <li>3. 支持数据类型有： <ul style="list-style-type: none"> <li>• bigint、int16、int、smallint、tinyint整数类型。</li> <li>• bigint unsigned、integer unsigned、smallint unsigned、tinyint unsigned无符号整数类型。</li> <li>• numeric、real浮点数类型。</li> <li>• character、character varying、clob、text字符串类型，仅支持纯数字整数字符串。</li> <li>• set类型。</li> <li>• NULL空类型。</li> </ul> </li> </ol>

### 8.1.1.3.9 其他函数

表 8-18 其他函数列表

序号	MySQL数据库	GaussDB数据库	差异
1	UUID()	支持	-
2	UUID_SHORT()	支持	-

### 8.1.1.4 操作符

GaussDB数据库兼容绝大多数MySQL的操作符，但存在部分差异。如未列出，操作符行为默认为GaussDB原生行为。

序号	MySQL数据库	GaussDB数据库	差异
1	安全等于 (<=>)	支持	-

序号	MySQL数据库	GaussDB数据库	差异
2	[NOT] REGEXP	支持	<ul style="list-style-type: none"> <li>当开启 b_format_dev_version='s2' 时，模式字符串pattern 中有 '\a', '\\d', '\\e', '\\n', '\\Z', '\\u' 等转义字符时，匹配源字符串 '\a', '\\d', '\\e', '\\n', '\\Z', '\\u' 时，GaussDB行为与MySQL5.7不一致。MySQL5.7存在bug，MySQL后续版本已经修复与GaussDB一致。</li> <li>当开启 b_format_dev_version='s2' 时，GaussDB '\\b' 可以与 '\\b' 匹配，MySQL匹配失败。</li> <li>模式字符串pat非法入参，只存在右单括号 ')' 时，GaussDB数据库报错。MySQL存在bug，后续版本已经修复此问题。</li> <li>在de abc 匹配序列de或abc的匹配规则，当 左右存在空值时，MySQL存在bug，会报错，后续版本已经修复此问题。</li> <li>空白字符[\t]正则匹配字符类 [:blank:]，GaussDB可匹配，兼容MySQL\t不能匹配 [:blank:]，MySQL存在bug，后续版本已经修复此问题。</li> <li>GaussDB支持非贪婪模式匹配，即尽可能少的匹配字符，在部分特殊字符后加 '?' 问号字符，例如：“??, *, +?, {n}?, {n,}?, {n,m}?”。MySQL 5.7版本不支持非贪婪模式匹配，并报错：Got error 'repetition-operator operand invalid' from regexp。MySQL 8.0版本已经支持。</li> <li>在binary字符集下，text类型、blob类型均会转换成 bytea类型，由于REGEXP操作符不支持bytea类型，因此无法匹配。</li> </ul>
3	[NOT] RLIKE	支持	同[NOT] REGEXP。

### 8.1.1.5 字符集

GaussDB数据库支持指定数据库、模式、表或列的字符集，支持的范围如下。

表 8-19 字符集列表

序号	MySQL数据库	GaussDB数据库
1	utf8mb4	支持
2	gbk	支持
3	gb18030	支持

### 8.1.1.6 排序规则

GaussDB数据库支持指定模式、表或列的排序规则，支持的范围如下。

#### 说明

排序规则差异说明：

- 当前仅有字符串类型、部分二进制类型支持指定排序规则，其他类型不支持指定排序规则，可以通过查询pg\_type系统表中类型的typcollation属性不为0来判断该类型支持字符序。MySQL中所有类型可以指定字符序，但除字符串、二进制类型其他排序规则无实际意义。
- 当前排序规则仅支持在其对应字符集与库级字符集一致时可以指定。
- utf8mb4字符集下默认字符序为utf8mb4\_general\_ci，与MySQL5.7保持一致，utf8mb4\_0900\_ai\_ci为utf8mb4的非默认字符序。
- GaussDB中utf8和utf8mb4为同一个字符集。

表 8-20 排序规则列表

序号	MySQL数据库	GaussDB数据库
1	utf8mb4_general_ci	支持
2	utf8mb4_unicode_ci	支持
3	utf8mb4_bin	支持
4	gbk_chinese_ci	支持
5	gbk_bin	支持
6	gb18030_chinese_ci	支持
7	gb18030_bin	支持
8	binary	支持
9	utf8mb4_0900_ai_ci	支持
10	utf8_general_ci	支持
11	utf8_bin	支持

### 8.1.1.7 SQL

GaussDB数据库兼容绝大多数MySQL语法，但存在部分差异。本章节介绍GaussDB数据库当前支持的MySQL语法。

#### 8.1.1.7.1 DDL

序号	概述	详细语法说明	差异
1	建表和修改表时支持创建主键、UNIQUE索引	ALTER TABLE、 CREATE TABLE	<ul style="list-style-type: none"> <li>GaussDB当前不支持UNIQUE INDEX KEY index_name语法，使用UNIQUE INDEX KEY index_name语法时会报错。MySQL支持。</li> <li>当约束被建立为全局二级索引，SQL语句中指定using btree时，底层会建立为ubtree。</li> <li>当约束关联的表为ustore，且SQL语句中指定为using btree时，底层会建立为ubtree。</li> </ul>
2	支持前缀索引	CREATE INDEX	<ul style="list-style-type: none"> <li>前缀长度不得超过2676，键值的实际长度受内部页面限制，若字段中含有多字节字符或者一个索引上有多个键，索引行长度可能会超限报错。</li> <li>CREATE INDEX语法中，不支持以下关键字作为前缀键的字段名称：COALESCE、EXTRACT、GREATEST、LEAST、LNNVL、NULLIF、NVL、NVL2、OVERLAY、POSITION、REGEXP_LIKE、SUBSTRING、TIMESTAMPDIFF、TREAT、TRIM、XMLCONCAT、XMLELEMENT、XMLEXISTS、XMLFOREST、XMLPARSE、XMLPI、XMLROOT、XMLSERIALIZE。</li> <li>主键和唯一键索引中不支持前缀键。</li> </ul>
3	支持指定字符集与排序规则	ALTER SCHEMA、 ALTER TABLE、 CREATE SCHEMA、 CREATE TABLE	-

序号	概述	详细语法说明	差异
4	创建分区表语法兼容	CREATE TABLE PARTITION、 CREATE TABLE SUBPARTITION	-
5	建表和修改表时支持指定表级和列级 comment	CREATE TABLE、 ALTER TABLE	-
6	创建索引时支持指定索引级 comment	CREATE INDEX	-

序号	概述	详细语法说明	差异
7	交换普通表和分区表分区的数据	ALTER TABLE PARTITION	<p>ALTER TABLE EXCHANGE PARTITION的差异点:</p> <ul style="list-style-type: none"> <li>MySQL表或分区使用 tablespace时, 则无法进行分区和普通表数据的交换; GaussDB表或分区使用不同的 tablespace时, 仍可进行分区和普通表数据的交换。</li> <li>对于列默认值, MySQL不会校验默认值, 因此默认值不同时也可进行分区和普通表数据的交换; GaussDB会校验默认值, 如果默认值不同, 则无法进行分区和普通表数据的交换。</li> <li>MySQL在分区表或普通表上进行DROP列操作后, 表结构仍然一致, 则可进行分区和普通表数据的交换; GaussDB需要保证普通表和分区表的被删除列严格对齐才能进行分区和普通表数据的交换。</li> <li>MySQL和GaussDB的哈希算法不同, 所以两者在相同的hash分区存储的数据可能不一致, 导致最后交换的数据也可能不一致。</li> <li>MySQL的分区表不支持外键, 普通表包含外键或其他表引用普通表的外键, 则无法进行分区和普通表数据的交换; GaussDB的分区表支持外键, 在两个表的外键约束一致时, 则可进行分区和普通表数据的交换; GaussDB的分区表不带外键, 普通表有其他表引用, 如果分区表和普通表表一致, 则可进行分区和普通表数据的交换。</li> </ul>

序号	概述	详细语法说明	差异
8	支持自增列	ALTER TABLE、 CREATE TABLE	<ul style="list-style-type: none"> <li>• 目前仅支持各个DN独立自增的本地自动增长列。</li> <li>• 自动增长列建议为索引（非全局二级索引）的第一个字段，否则建表时产生警告，含有自动增长列的表进行某些操作时会产生错误，例如：ALTER TABLE EXCHANGE PARTITION。MySQL自动增长列必须为索引第一个字段。</li> <li>• AUTO_INCREMENT = value语法，value必须为小于<math>2^{127}</math>的正数。MySQL不校验value。</li> <li>• 当自增值已经达到字段数据类型的最大值时，继续自增将产生错误。MySQL有些场景产生错误或警告，有些场景仍自增为最大值。</li> <li>• 不支持 innodb_autoinc_lock_mode系统变量，GaussDB的GUC参数 auto_increment_cache=0时，批量插入自动增长列的行为与MySQL系统变量 innodb_autoinc_lock_mode=1相似。</li> <li>• 自动增长列在导入数据或者进行Batch Insert执行计划的插入操作时，对于混合0、NULL和确定值的场景，如果产生错误，后续插入自增值不一定与MySQL完全一致。 <ul style="list-style-type: none"> <li>- 提供auto_increment_cache参数，可以控制预留自增值的数量。</li> </ul> </li> <li>• 批量插入在不同执行计划下，自增的顺序、自增值预留数量可能与MySQL不完全相同。例如：“INSERT INTO table VALUES(...),(...),...”由于要分布到不同DN，在某些执行计划中，DN获取不到即将插入的行数。 <ul style="list-style-type: none"> <li>- 提供auto_increment_cache参数，可以控制预留自增值的数量。</li> </ul> </li> </ul>



序号	概述	详细语法说明	差异
			<ul style="list-style-type: none"> <li>● 并行导入或插入自动增长列触发自增时，每个并行线程预留的缓存值也只在其线程中使用，未完全使用完毕的话，也会出现表中自动增长列的值不连续的情况。并行插入产生的自增值结果无法保证与MySQL完全一致。</li> <li>● SERIAL数据类型为原有的自增列，与AUTO_INCREMENT自增列有差异。MySQL的SERIAL数据类型就是AUTO_INCREMENT自增列。</li> <li>● 不允许auto_increment_offset的值大于auto_increment_increment的值，会产生错误。MySQL允许，并说明auto_increment_offset会被忽略。</li> <li>● 在表有主键或索引的情况下，ALTER TABLE命令重写表数据的顺序与MySQL不一定相同，GaussDB按表数据存储顺序重写，MySQL会按主键或索引顺序重写，导致自增值的顺序可能不同。</li> <li>● ALTER TABLE命令添加或修改自增列时，第一次预留自增值的数量是表统计信息中的行数，统计信息的行数不一定与MySQL一致。</li> <li>● 在触发器或用户自定义函数中自增时，刷新last_insert_id返回值。MySQL不刷新。</li> <li>● 对GUC参数auto_increment_offset和auto_increment_increment设置超出范围的值会产生错误。MySQL会自动改为边界值。</li> <li>● 暂不支持last_insert_id函数。</li> <li>● 本地临时表暂不支持自动增长列。</li> </ul>
9	支持删除表的主键约束	ALTER TABLE	-

序号	概述	详细语法说明	差异
10	支持CREATE TABLE ... LIKE语法兼容	CREATE TABLE ... LIKE	<ul style="list-style-type: none"> <li>● 在MySQL 8.0.16 之前的版本中，CHECK约束会被语法解析但功能会被忽略，表现为不复制CHECK约束，GaussDB支持复制CHECK约束。</li> <li>● 对于set数据类型，在建表时，MySQL支持复制，GaussDB不支持复制。</li> <li>● 对于主键约束名称，在建表时，MySQL所有主键约束名称固定为PRIMARY KEY，GaussDB不支持复制。</li> <li>● 对于唯一键约束名称，在建表时，MySQL支持复制，GaussDB不支持复制。</li> <li>● 对于CHECK约束名称，在建表时，MySQL 8.0.16 之前的版本无CHECK约束信息，GaussDB支持复制。</li> <li>● 对于索引名称，在建表时，MySQL支持复制，GaussDB不支持复制。</li> <li>● 在跨sql_mode模式建表时，MySQL受宽松模式和严格模式控制，GaussDB可能存在严格模式失效的情况。 例如：源表存在默认值“0000-00-00”，在“no_zero_date”严格模式下，GaussDB建表成功，且包含默认值“0000-00-00”，严格模式失效；而MySQL建表失败，受严格模式控制。</li> <li>● 针对跨数据库创建表，MySQL支持，GaussDB不支持。</li> <li>● 针对源表为临时表，创建非临时表，MySQL支持，GaussDB不支持。</li> </ul>

序号	概述	详细语法说明	差异
1 1	支持更改表名兼容语法	ALTER TABLE[ IF EXISTS ] tbl_name RENAME [TO   AS   =] new_tbl_name; RENAME {TABLE   TABLES} tbl_name TO new_tbl_name [, tbl_name2 TO new_tbl_name2, ...];	<ul style="list-style-type: none"><li>• GaussDB的alter rename语法仅支持修改表名称功能操作，不能耦合其它功能操作；</li><li>• GaussDB的仅旧表名字段支持如schema.table_name用法；且新表名与旧表名将属于同一schema下；</li><li>• GaussDB的不支持新旧表跨shema重命名操作；但如有权限，则可在当前schema下修改其它schema下表名称；</li><li>• GaussDB的rename多组表的语法支持全为本地临时表的重命名，不支持本地临时表和非本地临时表组合的场景。</li></ul>

序号	概述	详细语法说明	差异
1 2	支持增加子分区语法兼容	<p>ALTER TABLE [ IF EXISTS ] { table_name [*]   ONLY table_name   ONLY ( table_name ) } action [, ... ];</p> <p>action: move_clause   exchange_clause   row_clause   merge_clause   modify_clause   split_clause   add_clause   drop_clause   ilm_clause</p> <p>add_clause: ADD { {partition_less_than_item   partition_start_end_item   partition_list_item }   PARTITION( {partition_less_than_item   partition_start_end_item   partition_list_item } ) }</p>	<ul style="list-style-type: none"> <li>不支持ALTER TABLE table_name ADD PARTITION (partition_definition1, partition_definition1,...);语法添加多分区。</li> <li>仅支持原有添加多分区语法: ALTER TABLE table_name ADD PARTITION (partition_definition1), ADD PARTITION (partition_definition2[y1] ), ...;。</li> </ul>

### 8.1.1.7.2 DML

序号	概述	详细语法说明	差异
1	DELETE支持ORDER BY和LIMIT	DELETE	-
2	UPDATE支持ORDER BY和LIMIT	UPDATE	-

序号	概述	详细语法说明	差异
3	REPLACE INTO语法兼容	REPLACE	<ul style="list-style-type: none"> <li>时间类型初始值的差异。例如： <ul style="list-style-type: none"> <li>MySQL不受严格模式和宽松模式的影响，可向表中插入时间0值，即： <pre>mysql&gt; CREATE TABLE test(f1 TIMESTAMP NOT NULL, f2 DATETIME NOT NULL, f3 DATE NOT NULL); Query OK, 1 row affected (0.00 sec)  mysql&gt; REPLACE INTO test VALUES(f1, f2, f3); Query OK, 1 row affected (0.00 sec)  mysql&gt; SELECT * FROM test; +-----+-----+   f1            f2              f3                          +-----+-----+   0000-00-00 00:00:00   0000-00-00 00:00:00   0000-00-00   +-----+-----+ 1 row in set (0.00 sec)</pre> </li> <li>GaussDB在宽松模式下才可以成功插入时间0值，即 <pre>gaussdb=# SET b_format_version = '5.7'; SET gaussdb=# SET b_format_dev_version = 's1'; SET gaussdb=# SET sql_mode = ""; SET gaussdb=# CREATE TABLE test(f1 TIMESTAMP NOT NULL, f2 DATETIME NOT NULL, f3 DATE NOT NULL) DISTRIBUTE BY HASH(f1); CREATE TABLE gaussdb=# REPLACE INTO test VALUES(f1, f2, f3); REPLACE 0 1 gaussdb=# SELECT * FROM test; f1            f2            f3 +-----+-----+ 0000-00-00 00:00:00   0000-00-00 00:00:00   0000-00-00 (1 row) 在严格模式下，则报错date/ time field value out of range: "0000-00-00 00:00:00"。</pre></li> </ul> </li> <li>BIT类型在有NOT NULL时初始值的差异。例如：</li> </ul>

序号	概述	详细语法说明	差异
			<ul style="list-style-type: none"> <li>- MySQL BIT类型的初始值为空串", 即: mysql&gt; CREATE TABLE test(f1 BIT(3) NOT NULL); Query OK, 0 rows affected (0.01 sec)  mysql&gt; REPLACE INTO test VALUES(f1); Query OK, 1 row affected (0.00 sec)  mysql&gt; SELECT f1, f1 IS NULL FROM test; +----+-----+   f1   f1 is null   +----+-----+     0       0   +----+-----+ 2 rows in set (0.00 sec)</li> <li>- GaussDB BIT类型的初始值为NULL, 则报错。 gaussdb=# CREATE TABLE test(f1 int, f2 BIT(3) NOT NULL) DISTRIBUTE BY HASH(f1); CREATE TABLE gaussdb=# REPLACE INTO test VALUES(1, f2); ERROR: null value in column "f2" violates not-null constraint DETAIL: Failing row contains (1, null).</li> </ul>
4	SELECT支持指定多分区查询	SELECT	-
5	UPDATE支持指定多分区更新	UPDATE	-

序号	概述	详细语法说明	差异
6	LOAD DATA导入数据功能	LOAD DATA	<ul style="list-style-type: none"> <li>● LOAD DATA语法执行结果与MySQL严格模式一致，宽松模式暂未适配。</li> <li>● IGNORE与LOCAL参数功能仅为当导入数据与表中数据存在冲突时，忽略当前冲突行数据功能和当文件中字段数小于指定表中列数时自动为其余列填充默认值功能，其余功能暂未适配。</li> <li>● 指定LOCAL关键字，且文件路径为相对路径时，文件从二进制目录下搜索；不指定LOCAL关键字，且文件路径为相对路径时，文件从数据目录下搜索。</li> <li>● 语法中指定分隔符，转义字符，分行符等符号时，若指定为单引号，将导致词法解析错误。</li> <li>● [(col_name_or_user_var [, col_name_or_user_var] ...)]指定列参数不支持重复指定列。</li> <li>● [FIELDS TERMINATED BY 'string']指定换行符不能与[LINES TERMINATED BY 'string']分隔符相同。</li> <li>● 执行LOAD DATA语法写入表中的数据若无法转换为表中数据类型格式时报错。</li> <li>● 指定列时，仅支持指定为列名，不支持指定为用户变量。</li> <li>● LOAD DATA SET表达式中不支持指定列名计算。</li> <li>● 若set表达式返回值类型与对应列类型之间不存在隐式转换函数则报错。</li> <li>● LOAD DATA不支持INSERT、DELETE触发器。</li> <li>● LOAD DATA只能用于表，不能用于视图。</li> <li>● Windows下的文件与Linux环境下文件默认换行符存在差异，LOAD DATA无法识别此场景会</li> </ul>

序号	概述	详细语法说明	差异
			报错，建议用户导入时检查导入文件行结尾的换行符。



序号	概述	详细语法说明	差异
7	INSERT IGNORE兼容	INSERT IGNORE	<ul style="list-style-type: none"> <li>GaussDB会返回降级后的错误信息，MySQL则会将降级后的错误信息记录到错误堆栈中，然后调用show warnings;命令查看。</li> <li>时间类型的差异。例如： <ul style="list-style-type: none"> <li>GaussDB中date、datetime、timestamp默认零值。  <pre>gaussdb=# CREATE TABLE test(f1 DATE NOT NULL, f2 DATETIME NOT NULL, f3 TIMESTAMP NOT NULL); CREATE TABLE gaussdb=# INSERT IGNORE INTO test VALUES(NULL, NULL, NULL); WARNING: null value in column "f1" violates not-null constraint DETAIL: Failing row contains (null, null, null, null). WARNING: null value in column "f2" violates not-null constraint DETAIL: Failing row contains (null, null, null, null). WARNING: null value in column "f3" violates not-null constraint DETAIL: Failing row contains (null, null, null, null). INSERT 0 1 gaussdb=# SELECT * FROM test;   f1         f2        -----+----- 1970-01-01   1970-01-01 00:00:00   1970-01-01 00:00:00 (1 row)</pre> </li> <li>MySQL中date、datetime、timestamp默认零值。  <pre>mysql&gt; CREATE TABLE test(f1 DATE NOT NULL, f2 DATETIME NOT NULL, f3 TIMESTAMP NOT NULL); Query OK, 0 rows affected (0.00 sec)  mysql&gt; INSERT IGNORE INTO test VALUES(NULL, NULL, NULL); Query OK, 1 row affected, 3 warnings (0.00 sec)  mysql&gt; show warnings; +-----+-----+   Level   Code   Message                                 +-----+-----+ </pre> </li> </ul> </li> </ul>

序号	概述	详细语法说明	差异
			<pre>   Warning   1048   Column 'f1' cannot be null     Warning   1048   Column 'f2' cannot be null     Warning   1048   Column 'f3' cannot be null   +-----+----- +-----+ 3 rows in set (0.00 sec)  mysql&gt; SELECT * FROM test; +-----+-----+ +-----+   f1        f2        f3        +-----+-----+ +-----+   0000-00-00   0000-00-00 00:00:00     0000-00-00 00:00:00   +-----+-----+ +-----+ 1 row in set (0.00 sec) </pre> <ul style="list-style-type: none"> <li>• 由于GaussDB不支持MySQL的bit类型，因此忽略bit类型NOT NULL约束和插入的bit类型长度与定义不同的场景下不支持INSERT IGNORE错误降级。 <ul style="list-style-type: none"> <li>- GaussDB中bit类型 <pre> gaussdb=# CREATE TABLE test(f1 BIT(10) NOT NULL); CREATE TABLE gaussdb=# INSERT IGNORE INTO test VALUES(NULL); ERROR: Un-support feature DETAIL: ignore null for insert statement is not supported in column f1. gaussdb=# INSERT IGNORE INTO test VALUES('1010'); ERROR: bit string length 4 does not match type bit(10) CONTEXT: referenced column: f1 </pre> </li> <li>- MySQL中bit类型 <pre> mysql&gt; CREATE TABLE test(f1 BIT(10) NOT NULL); Query OK, 0 rows affected (0.00 sec)  mysql&gt; INSERT IGNORE INTO test VALUES(NULL); Query OK, 1 row affected, 1 warning (0.00 sec)  mysql&gt; INSERT IGNORE INTO test VALUES('1010'); Query OK, 1 row affected, 1 warning (0.01 sec) </pre> </li> </ul> </li> <li>• MySQL数据库时间类型指定精度时，插入时间零值会显示精</li> </ul>

序号	概述	详细语法说明	差异
			<p>度，GaussDB则不显示，例如：</p> <ul style="list-style-type: none"> <li>- GaussDB指定时间精度 <pre> gaussdb=# CREATE TABLE test(f1 TIME(3) NOT NULL, f2 DATETIME(3) NOT NULL, f3 TIMESTAMP(3) NOT NULL); CREATE TABLE gaussdb=# INSERT IGNORE INTO test VALUES(NULL,NULL,NULL); WARNING: null value in column "f1" violates not-null constraint DETAIL: Failing row contains (null, null, null). WARNING: null value in column "f2" violates not-null constraint DETAIL: Failing row contains (null, null, null). WARNING: null value in column "f3" violates not-null constraint DETAIL: Failing row contains (null, null, null). INSERT 0 1 gaussdb=# SELECT * FROM test;  f1   f2   f3 -----+-----+----- 00:00:00   1970-01-01 00:00:00   1970-01-01 00:00:00 (1 row) </pre> </li> <li>- MySQL指定时间精度 <pre> mysql&gt; CREATE TABLE test(f1 TIME(3) NOT NULL, f2 DATETIME(3) NOT NULL, f3 TIMESTAMP(3) NOT NULL); Query OK, 0 rows affected (0.00 sec)  mysql&gt; INSERT IGNORE INTO test VALUES(NULL,NULL,NULL); Query OK, 1 row affected, 3 warnings (0.00 sec)  mysql&gt; SELECT * FROM test; +-----+ +-----+ +-----+   f1        f2        f3        +-----+   00:00:00.000   0000-00-00 00:00:00.000   0000-00-00 00:00:00.000   +-----+ +-----+ +-----+ 1 row in set (0.00 sec) </pre> </li> </ul> <ul style="list-style-type: none"> <li>• 由于MySQL数据库和GaussDB执行过程的差异，因此，产生</li> </ul>

序号	概述	详细语法说明	差异
			<p>的warnings条数可能不同，例如：</p> <ul style="list-style-type: none"> <li>- GaussDB产生的warnings条数                     <pre>gaussdb=# CREATE TABLE test(f1 INT, f2 INT not null); CREATE TABLE gaussdb=# INSERT INTO test VALUES(1,0),(3,0),(5,0); INSERT 0 3 gaussdb=# INSERT IGNORE INTO test SELECT f1+1, f1/f2 FROM test; WARNING: division by zero CONTEXT: referenced column: f2 WARNING: null value in column "f2" violates not-null constraint DETAIL: Failing row contains (2, null). WARNING: division by zero CONTEXT: referenced column: f2 WARNING: null value in column "f2" violates not-null constraint DETAIL: Failing row contains (4, null). WARNING: division by zero CONTEXT: referenced column: f2 WARNING: null value in column "f2" violates not-null constraint DETAIL: Failing row contains (6, null). INSERT 0 3</pre> </li> <li>- MySQL产生的warnings条数                     <pre>mysql&gt; CREATE TABLE test(f1 INT, f2 INT not null); Query OK, 0 rows affected (0.01 sec)  mysql&gt; INSERT INTO test VALUES(1,0),(3,0),(5,0); Query OK, 3 rows affected (0.00 sec) Records: 3 Duplicates: 0 Warnings: 0  mysql&gt; INSERT IGNORE INTO test SELECT f1+1, f1/f2 FROM test; Query OK, 3 rows affected, 4 warnings (0.00 sec) Records: 3 Duplicates: 0 Warnings: 4</pre> </li> </ul> <ul style="list-style-type: none"> <li>• MySQL数据库和GaussDB INSERT IGNORE在触发器中的差异，例如：             <ul style="list-style-type: none"> <li>- GaussDB触发器中使用 INSERT IGNORE                     <pre>gaussdb=# CREATE TABLE test1(f1 INT NOT NULL); CREATE TABLE gaussdb=# CREATE TABLE test2(f1</pre> </li> </ul> </li> </ul>

序号	概述	详细语法说明	差异
			<pre> INT); CREATE TABLE gaussdb=# CREATE OR REPLACE FUNCTION trig_test() RETURNS TRIGGER AS \$\$ gaussdb\$\$ BEGIN gaussdb\$\$ INSERT IGNORE INTO test1 VALUES(NULL); gaussdb\$\$ RETURN NEW; gaussdb\$\$ END; gaussdb\$\$ \$\$ LANGUAGE plpgsql; CREATE FUNCTION gaussdb=# CREATE TRIGGER trig2 BEFORE INSERT ON test2 FOR EACH ROW EXECUTE PROCEDURE trig_test(); CREATE TRIGGER gaussdb=# INSERT INTO test2 VALUES(NULL); WARNING: null value in column "f1" violates not-null constraint DETAIL: Failing row contains (null). CONTEXT: SQL statement "INSERT IGNORE INTO test1 VALUES(NULL)" PL/pgSQL function trig_test() line 3 at SQL statement INSERT 0 1 gaussdb=# SELECT * FROM test1; f1 ---- 0 (1 rows)  gaussdb=# SELECT * FROM test2; f1 ---- (1 rows) </pre> <p>- MySQL触发器中使用INSERT IGNORE</p> <pre> mysql&gt; CREATE TABLE test1(f1 INT NOT NULL); Query OK, 0 rows affected (0.01 sec)  mysql&gt; CREATE TABLE test2(f1 INT); Query OK, 0 rows affected (0.00 sec)  mysql&gt; DELIMITER    mysql&gt; CREATE TRIGGER trig2 BEFORE INSERT ON test2 FOR EACH ROW -&gt; BEGIN -&gt; INSERT IGNORE into test1 values(NULL); -&gt; END   Query OK, 0 rows affected (0.01 sec)  mysql&gt; DELIMITER ; </pre>

序号	概述	详细语法说明	差异
			<pre>mysql&gt; INSERT INTO test2 VALUES(NULL); ERROR 1048 (23000): Column 'f1' cannot be null mysql&gt; INSERT IGNORE INTO test2 VALUES(NULL); Query OK, 1 row affected (0.00 sec)</pre> <pre>mysql&gt; SELECT * FROM test1; +----+   f1   +----+   0   +----+ 1 row in set (0.00 sec)</pre> <pre>mysql&gt; SELECT * FROM test2; +-----+   f1   +-----+   NULL   +-----+ 1 row in set (0.00 sec)</pre> <ul style="list-style-type: none"> <li>● GaussDB的bool、serial的实现机制与MySQL不同，因此其默认零值与MySQL不同，例如： <ul style="list-style-type: none"> <li>- GaussDB的行为 <pre>gaussdb=# CREATE TABLE test(f1 SERIAL, f2 BOOL NOT NULL); NOTICE: CREATE TABLE will create implicit sequence "test_f1_seq" for serial column "test.f1" CREATE TABLE gaussdb=# INSERT IGNORE INTO test values(NULL,NULL); WARNING: null value in column "f1" violates not-null constraint DETAIL: Failing row contains (null, null). WARNING: null value in column "f2" violates not-null constraint DETAIL: Failing row contains (null, null). INSERT 0 1 gaussdb=# SELECT * FROM test; f1   f2 -----+----- 0   f (1 row)</pre> </li> <li>- MySQL的行为 <pre>mysql&gt; CREATE TABLE test(f1 SERIAL, f2 BOOL NOT NULL); Query OK, 0 rows affected (0.00 sec)  mysql&gt; INSERT IGNORE INTO test values(NULL,NULL); Query OK, 1 row affected, 1 warning (0.00 sec)  mysql&gt; SELECT * FROM test;</pre> </li> </ul> </li> </ul>

序号	概述	详细语法说明	差异
			<pre>+----+----+   f1   f2   +----+----+   1   0   +----+----+ 1 row in set (0.00 sec)</pre>
8	HAVING语法	SELECT	GaussDB的HAVING必须仅引用GROUP BY子句中的列或聚合函数中使用的列。但是，MySQL支持对此行为的扩展，并允许HAVING引用列表中的SELECT列和外部子查询中的列。

### 8.1.1.7.3 DCL

序号	概述	详细语法说明	差异
1	SET NAMES指定COLLATE字句	SET [ SESSION   LOCAL ] NAMES {'charset_name' [COLLATE 'collation_name']   DEFAULT};	GaussDB中暂不支持指定charset_name与数据库字符集不同。具体请参考《开发者指南》中“SQL参考 > SQL语法 > S > SET”章节。

### 8.1.1.8 驱动

#### 8.1.1.8.1 JDBC

#### JDBC 接口参考

#### 获取结果集中的数据

ResultSet对象提供了丰富的方法，以获取结果集中的数据。获取数据常用的方法如表8-21所示，其他方法请参考JDK官方文档。

表 8-21 ResultSet 对象的常用方法

方法	描述	差异
int getInt(int columnIndex)	按列标获取int型数据。	-
int getInt(String columnLabel)	按列名获取int型数据。	-

方法	描述	差异
String getString(int columnIndex)	按列标获取 String型数 据。	字段类型为整型且带有ZEROFILL属性时， GaussDB按照ZEROFILL属性要求的宽度信息用0 进行补位后输出结果，MySQL直接输出结果。
String getString(String columnLabel)	按列名获取 String型数 据。	字段类型为整型且带有ZEROFILL属性时， GaussDB按照ZEROFILL属性要求的宽度信息用0 进行补位后输出结果，MySQL直接输出结果。
Date getDate(int columnIndex)	按列标获取 Date型数据	-
Date getDate(String columnLabel)	按列名获取 Date型数 据。	-

## 8.2 主备版

### 8.2.1 概述

本手册为GaussDB数据库（B兼容模式和M-Compatibility兼容模式）与MySQL 5.7数据库信息对比。其中，**MySQL数据库兼容性B模式**为B模式下的MySQL数据库兼容性；**MySQL数据库兼容性M-Compatibility模式**为M-Compatibility模式下的MySQL数据库兼容性说明。

### 8.2.2 MySQL 数据库兼容性 B 模式

#### 8.2.2.1 MySQL 数据库兼容性概述

本章节主要介绍GaussDB数据库的MySQL 5.7兼容模式（即sql\_compatibility = 'B'、且设置参数b\_format\_version='5.7'、b\_format\_dev\_version='s1'时）与MySQL5.7数据库的兼容性对比信息。仅介绍503.0.0版本后新增的兼容性特性，特性的相关规格和约束建议在开发者指南中查看。

GaussDB数据库在数据类型、SQL功能和数据库对象等基本功能上与MySQL数据库兼容。

由于GaussDB数据库与MySQL数据库底层框架实现存在差异，GaussDB数据库与MySQL数据库仍存在部分差异。

#### 8.2.2.2 数据类型

GaussDB的数据类型的大部分功能场景与MySQL一致，但存在部分差异。

- 除特别说明，部分数据类型精度、标度、位数大小等不支持用浮点型数值定义，建议使用合法的整型数值定义。



### 8.2.2.2.1 数值数据类型

表 8-22 整数类型

序号	MySQL数据库	GaussDB数据库	差异
1	BOOL	不完全兼容	MySQL: BOOL/BOOLEAN类型实际映射为TINYINT类型。
2	BOOLEAN	不完全兼容	GaussDB: 支持BOOL, 其中: <ul style="list-style-type: none"> <li>“真”值的有效文本值是: TRUE、't'、'true'、'y'、'yes'、'1'、'TRUE'、true、'on'以及所有非0数值。</li> <li>“假”值的有效文本值是: FALSE、'f'、'false'、'n'、'no'、'0'、0、'FALSE'、false、'off'。</li> </ul> 使用TRUE和FALSE是比较规范的做法(也是SQL兼容的做法)。
3	TINYINT[(M)] [UNSIGNED]	支持	详细请参见说明。
4	SMALLINT[(M)] [UNSIGNED]	支持	详细请参见说明。
5	MEDIUMINT[(M)] [UNSIGNED]	支持	MySQL存储MEDIUMINT数据需要3字节。 <ul style="list-style-type: none"> <li>带符号的范围是-8,388,608 ~ +8,388,607。</li> <li>无符号的范围是0 ~ +16,777,215。</li> </ul> GaussDB映射为INT类型, 存储需要4字节。 <ul style="list-style-type: none"> <li>带符号的范围是-2,147,483,648 ~ +2,147,483,647。</li> <li>无符号的范围是0 ~ +4,294,967,295。</li> </ul> 其他差异请参见说明。
6	INT[(M)] [UNSIGNED]	支持	详细请参见说明。
7	INTEGER[(M)] [UNSIGNED]	支持	详细请参见说明。

序号	MySQL数据库	GaussDB数据库	差异
8	BIGINT[(M)] [UNSIGNED]	支持	详细请参见说明。

## 说明

- 输入格式：
  - MySQL  
对于类似“asbd”、“12dd”、“12 12”等字符场景的输入，会采取截断或返回0值并上报WARNING处理，在严格模式插表时失败。
  - GaussDB
    - 整数类型（TINYINT、SMALLINT、MEDIUMINT、INT、INTEGER、BIGINT）的输入，当非法字符串部分被截断时，如“12@3”，会直接截断并无提示信息，插表成功。
    - 当整数类型全部被截断（如“@123”）或字符串为空时，返回0，且插表成功。
- 操作符：
  - +、-、\*操作符  
GaussDB：INT/INTEGER/SMALLINT/BIGINT在进行运算时，返回值为类型本身，不会向上提升类型，当返回值超范围时报错。  
MySQL：支持提升类型到BIGINT后计算。
  - |、&、^、~运算符  
GaussDB：在类型所占用BIT位中计算；GaussDB中^表示指数运算，如需使用异或运算符，使用#替换。  
MySQL：提升类型计算。
- 负数显示类型转换：  
GaussDB：宽松模式结果为0，严格模式报错。  
MySQL：依据其对应的二进制将最高位替换成数值位计算结果，例如(-1)::uint4 = 4294967295。
- 其他差异：  
INT[(M)]精度，MySQL控制格式化输出，GaussDB仅语法支持，不支持功能。
- 聚集函数：
  - variance：GaussDB表示样本方差，MySQL表示总体方差。
  - stddev：GaussDB表示样本标准差，MySQL表示总体标准差。
- 显示宽度：
  - 在为整型数字列指明宽度信息时，如果不同时指定ZEROFILL，则宽度信息在表结构描述中不显示。
  - INSERT语句插入字符类型字段，GaussDB统一补齐0后插入。
  - JOIN USING语句，涉及类型推导，MySQL默认第一张表列，GaussDB若结果为有符号类型则宽度信息失效，否则为第一张表字段宽度。
  - greatest/least、ifnull/if、case when / decode，MySQL不补齐0，GaussDB在类型及宽度信息一致时补齐0。
  - 作为函数/存储过程出入参、返回值时，MySQL支持功能、GaussDB语法不报错功能不支持。

表 8-23 任意精度类型

序号	MySQL数据库	GaussDB数据库	差异
1	DECIMAL[(M[,D])]	支持	<ul style="list-style-type: none"> <li>操作符：GaussDB中“^”表示指数运算，如需使用异或运算符，使用“#”替换；MySQL中“^”表示异或。</li> <li>取值范围：精度M，标度D不支持浮点型数值输入，只支持整型数值输入。</li> <li>输入格式：当字符串入参全部被截断时不会报错，如 '@123'；只有被部分截断时才会报错，如 '12@3'。</li> </ul>
2	NUMERIC[(M[,D])]	支持	
3	DEC[(M[,D])]	支持	
4	FIXED[(M[,D])]	不支持	-

表 8-24 浮点类型

序号	MySQL数据库	GaussDB数据库	差异
1	FLOAT[(M,D)]	支持	<ul style="list-style-type: none"> <li>分区表支持：FLOAT数据类型不支持KEY键值分区策略分区表。</li> <li>操作符：GaussDB中“^”表示指数运算，如需使用异或运算符，使用“#”替换；MySQL中“^”表示异或。</li> <li>取值范围：精度M，标度D不支持浮点型数值输入，只支持整型数值输入。</li> <li>输出格式：对于非法入参一律报错ERROR，不会在sql_mode="的宽松模式下报WARNING。</li> </ul>
2	FLOAT(p)	支持	<ul style="list-style-type: none"> <li>分区表支持：FLOAT数据类型不支持KEY键值分区策略分区表。</li> <li>操作符：数值类型使用^操作符，与MySQL不一致，GaussDB中^操作符为取指数运算。</li> <li>取值范围：定义精度p时，仅支持使用合法的整型数据类型。</li> <li>输出格式：精度M，标度D不支持浮点型数值输入，只支持整型数值输入。</li> <li>对于非法入参一律报错ERROR，不会在sql_mode="的宽松模式下报WARNING。</li> </ul>

序号	MySQL数据库	GaussDB数据库	差异
3	DOUBLE[(M,D)]	支持	<ul style="list-style-type: none"> <li>分区表支持：DOUBLE数据类型不支持KEY键值分区策略分区表。</li> <li>操作符：GaussDB中“^”表示指数运算，如需使用异或运算符，使用“#”替换；MySQL中“^”表示异或。</li> <li>取值范围：精度M，标度D不支持浮点型数值输入，只支持整型数值输入。</li> <li>输出格式：对于非法入参一律报错ERROR，不会在sql_mode="的宽松模式下报WARNING。</li> </ul>
4	DOUBLE PRECISION[(M,D)]	支持	<ul style="list-style-type: none"> <li>操作符：GaussDB中“^”表示指数运算，如需使用异或运算符，使用“#”替换；MySQL中“^”表示异或</li> <li>取值范围：精度M，标度D不支持浮点型数值输入，只支持整型数值输入。</li> <li>输出格式：对于非法入参一律报错ERROR，不会在sql_mode="的宽松模式下报WARNING。</li> </ul>
5	REAL[(M,D)]	支持	<ul style="list-style-type: none"> <li>分区表支持：REAL数据类型不支持KEY值分区策略分区表。</li> <li>操作符：GaussDB中“^”表示指数运算，如需使用异或运算符，使用“#”替换；MySQL中“^”表示异或</li> <li>取值范围：精度M，标度D不支持浮点型数值输入，只支持整型数值输入。</li> <li>输出格式：对于非法入参一律报错ERROR，不会在sql_mode="的宽松模式下报WARNING。</li> </ul>

表 8-25 序列整数

序号	MySQL数据库	GaussDB数据库	差异
1	SERIAL	不完全兼容	<p>SERIAL介绍详见GaussDB 开发者指南 数值类型章节。</p> <p>规格上与MySQL的差异如下： CREATE TABLE test(f1 serial, f2 CHAR(20));</p> <ul style="list-style-type: none"> <li>类型定义差异，MySQL的serial是映射到BIGINT(20) UNSIGNED NOT NULL AUTO_INCREMENT UNIQUE，GaussDB的serial是映射到INTEGER NOT NULL DEFAULT nextval('test_f1_seq'::regclass)。如： -- MySQL serial的定义： mysql&gt; SHOW CREATE TABLE test\G ***** 1. row ***** Table: test Create Table: CREATE TABLE `test` ( `f1` bigint(20) unsigned NOT NULL AUTO_INCREMENT, `f2` char(20) DEFAULT NULL, UNIQUE KEY `f1` (`f1`) ) ENGINE=InnoDB DEFAULT CHARSET=utf8 1 row in set (0.00 sec)  -- GaussDB serial的定义 gaussdb=# \d+ test  Table "public.test" Column   Type   Modifiers   Storage   Stats target   Description -----+----- +-----+-----+----- f1   integer   not null default nextval('test_f1_seq'::regclass)   plain     f2   character(20)   extended   Has OIDs: no Options: orientation=row, compression=no, storage_type=USTORE</li> <li>INSERT场景下serial类型DEFAULT值的差异。如： -- MySQL插入serial的DEFAULT值 mysql&gt; INSERT INTO test VALUES(DEFAULT, 'aaaa'); Query OK, 1 row affected (0.00 sec)  mysql&gt; INSERT INTO test VALUES(10, 'aaaa'); Query OK, 1 row affected (0.00 sec)  mysql&gt; INSERT INTO test VALUES(DEFAULT, 'aaaa'); Query OK, 1 row affected (0.00 sec)  mysql&gt; SELECT * FROM test; +----+-----+   f1   f2   +----+-----+   1   aaaa     10   aaaa     11   aaaa  </li> </ul>

序号	MySQL数据库	GaussDB数据库	差异
			<pre> +----+-----+ 3 rows in set (0.00 sec)  -- GaussDB插入serial的DEFAULT值 gaussdb=# INSERT INTO test VALUES(DEFAULT, 'aaaa'); INSERT 0 1 gaussdb=# INSERT INTO test VALUES(10, 'aaaa'); INSERT 0 1 gaussdb=# INSERT INTO test VALUES(DEFAULT, 'aaaa'); INSERT 0 1 gaussdb=# SELECT * FROM test;  f1        f2 -----+-----  1   aaaa  2   aaaa 10   aaaa (3 rows) </pre> <ul style="list-style-type: none"> <li>REPLACE场景下serial类型引用列的差异，GaussDB引用列的介绍详见GaussDB 开发者指南 REPLACE语法章节。如： <pre> -- MySQL插入serial引用列的值 mysql&gt; REPLACE INTO test VALUES(f1, 'aaaa'); Query OK, 1 row affected (0.00 sec)  mysql&gt; REPLACE INTO test VALUES(f1, 'bbbb'); Query OK, 1 row affected (0.00 sec)  mysql&gt; SELECT * FROM test; +----+-----+   f1   f2   +----+-----+   1   aaaa     2   bbbb   +----+-----+ 2 rows in set (0.00 sec)  -- GaussDB插入serial引用列的值 gaussdb=# REPLACE INTO test VALUES(f1, 'aaaa'); REPLACE 0 1 gaussdb=# REPLACE INTO test VALUES(f1, 'bbbb'); REPLACE 0 1 gaussdb=# SELECT * FROM test;  f1        f2 -----+-----  0   aaaa  0   bbbb (2 rows) </pre> </li> </ul>

### 8.2.2.2.2 日期与时间数据类型

表 8-26 日期与时间数据类型

序号	MySQL数据库	GaussDB数据库	差异
1	DATE	支持	<p>GaussDB支持date数据类型，与MySQL相比规格上存在如下差异：</p> <ul style="list-style-type: none"> <li>• 输入格式 <ul style="list-style-type: none"> <li>- GaussDB只支持字符类型，不支持数值类型。如支持'2020-01-01'或'20200101'字符串格式，不支持20200101数值输入。MySQL支持数值输入转换为date类型。</li> <li>- 分隔符：GaussDB不支持加号“+”、冒号“:”作为年、月、日之间的分隔符，其他的符号都支持。MySQL所有符号均可作为分隔符。分隔符混合使用的某些场景也不支持，与MySQL也有差异，如'2020-01&gt;01'，'2020/01+01'等，不建议混合使用分隔符，建议使用最常用的“-”、“/”作为分隔符。</li> <li>- 无分隔符：推荐使用完整格式，如'YYYYMMDD'或者'YYMMDD'。其他不完整的格式（包括超长格式）解析的规则与MySQL存在差异，可能报错或者解析的结果与MySQL不一致，不推荐使用。</li> </ul> </li> <li>• 输出格式 <p>GaussDB在sql_mode参数不包含'strict_trans_tables'选项（定义为宽松模式，否则为严格模式）时，允许年、月、日的值是0，但是输出时会按照年、月、日的顺序依次转换为合法的值，如date'0000-00-10'转换为：0002-12-10 BC。非法输入或者超过范围时，会报warning信息，并返回0000-00-00值。MySQL对于包含0值年、月、日的date值会原样输出。</p> </li> <li>• 取值范围 <p>GaussDB的范围是4713-01-01 BC ~ 5874897-12-31 AD，支持公元前的日期，宽松模式下超过范围时，返回的是0值：0000-00-00，严格模式下会报错。MySQL的范围是 0000-00-00 ~ 9999-12-31，宽松模式下超过范围后，各个场景下的表现并不一致，可能报错（如select查询语句中），也可能返回0000-00-00值（如insert时）。此差异会导致date类型作为函数入参时，函数返回的结果存在差异。</p> </li> </ul>

序号	MySQL数据库	GaussDB数据库	差异
			<ul style="list-style-type: none"> <li>● 操作符                             <ul style="list-style-type: none"> <li>- GaussDB仅支持date类型之间的比较操作符“=”、“!=”、“&lt;”、“&lt;=”、“&gt;”、“&gt;=”，返回true或者false；date与interval类型的加法运算，返回结果为date类型；date与interval类型的减法运算，返回结果为date类型；date类型之间的减法运算，返回结果为interval类型。</li> <li>- MySQL date类型和其他数值类型运算时，会先将date转换为数值类型，然后按照数值类型运算，结果也为数值类型。与GaussDB存在差异。如：                                     <pre data-bbox="901 795 1428 1232"> -- MySQL: date + 数值, 先将date类型转换为数值 20200101,再与1相加, 结果为数值类型20200102 mysql&gt; select date'2020-01-01' + 1; +-----+   date'2020-01-01' + 1   +-----+            20200102   +-----+ 1 row in set (0.00 sec)  -- GaussDB: date + 数值, 数值类型会转换为interval 类型 1 day, 然后相加得到新的日期。 gaussdb=# select date'2020-01-01' + 1; ?column? ----- 2020-01-02 (1 row) </pre> </li> </ul> </li> <li>● 类型转换                             相比较MySQL，GaussDB仅支持date类型与char(n)、nchar(n)、datetime、timestamp类型之间的相互转换，不支持与binary、decimal、json、integer、unsigned integer、time 类型之间的转换。集合等场景和复杂表达式场景下公共类型的确定原则与MySQL也不一致，参考<a href="#">数据类型转换</a>章节的描述。                         </li> </ul>



序号	MySQL数据库	GaussDB数据库	差异
2	DATETIME[(fsp)]	支持	<p>GaussDB支持datetime数据类型，与MySQL相比规格上存在如下差异：</p> <ul style="list-style-type: none"> <li>● 输入格式： <ul style="list-style-type: none"> <li>- GaussDB只支持字符类型，不支持数值类型。如支持'2020-01-01 10:20:30.123456'或'20200101102030.123456'字符串格式，不支持如20200101102030.123456的数值类型输入。MySQL支持数值输入转换为datetime类型。</li> <li>- 分隔符：GaussDB不支持加号“+”、冒号“:”作为年、月、日之间的分隔符，其他的符号都支持。仅支持冒号“:”作为时、分、秒之间的分隔符，其他的符号都不支持。分隔符混合使用的某些场景也不支持，与MySQL也有差异，不推荐使用。MySQL支持所有符号作为分隔符。</li> <li>- 无分隔符：GaussDB推荐使用完整格式'YYYYMMDDhhmiss.ffffff'。其他不完整的格式（包括超长格式）解析的规则可能与MySQL存在差异，可能报错或者解析的结果与MySQL不一致，不推荐使用。</li> </ul> </li> <li>● 输出格式： <ul style="list-style-type: none"> <li>- 统一为'YYYY-MM-DD hh:mi:ss.ffffff'的格式，格式与MySQL无差异，且不受DateStyle参数的影响。但是对于精度部分，如果最后几位为0，GaussDB不显示，MySQL会显示。</li> <li>- GaussDB在sql_mode参数不包含'strict_trans_tables'选项（定义为宽松模式，否则为严格模式）时，允许年、月、日值是0，但是输出时会按照年、月、日的顺序依次转换为合法的值，如datetime '0000-00-10 00:00:00'转换为：0002-12-10 00:00:00 BC。非法输入或者超过范围时，会报warning信息，并返回0000-00-00 00:00:00值。MySQL对于包含0值年、月、日的datetime值会原样输出。</li> </ul> </li> <li>● 取值范围 <ul style="list-style-type: none"> <li>4713-11-24 00:00:00.000000 BC ~ 294277-01-09 04:00:54.775806 AD。294277-01-09 04:00:54.775807 AD 返回的是infinity。对于超过范围的值，严格模</li> </ul> </li> </ul>

序号	MySQL数据库	GaussDB数据库	差异
			<p>式下GaussDB会报错，MySQL是否报错取决于使用场景。一般查询场景不报错，而执行DML SQL语句更改表属性的值时报错。宽松模式下GaussDB返回0000-00-00 00:00:00值，MySQL根据使用场景可能报错，也可能返回0000-00-00 00:00:00值或者null值。这个差异会导致以datetime类型为入参的函数执行结果与MySQL也存在差异。</p> <ul style="list-style-type: none"> <li>● 精度 范围0~6，作为表列的类型时缺省为0，与MySQL一致。对于 datetime[(p)] 'str' 表达式场景，GaussDB将(p)作为精度解析，缺省为6，将'str'按照p指定的精度格式化成datetime类型。MySQL不支持 datetime[(p)] 'str'表达式。</li> <li>● 操作符 <ul style="list-style-type: none"> <li>- GaussDB仅支持datetime类型之间的比较操作符“=”、“!=”、“&lt;”、“&lt;=”、“&gt;”、“&gt;=”，返回true或者false；datetime与interval类型的加法运算，返回结果为datetime类型；datetime与interval类型的减法运算，返回结果为datetime类型；datetime类型之间的减法运算，返回结果为interval类型。</li> <li>- MySQL datetime类型和其他数值类型运算时，会先将datetime转换为数值类型，然后按照数值类型运算，结果也为数值类型。与GaussDB存在差异。如： <pre> -- MySQL: datetime + 数值, 先将datetime类型转换为数值20201010123456,再与1相加, 结果为数值类型 20201010123457 mysql&gt; select cast('2020-10-10 12:34:56.123456' as datetime) + 1; +-----+   cast('2020-10-10 12:34:56.123456' as datetime) + 1   +-----+                                       20201010123457   +-----+ 1 row in set (0.00 sec)  -- GaussDB: datetime + 数值, 数值类型会转换为interval类型 1 day, 然后相加得到新的datetime。 gaussdb=# select cast('2020-10-10 12:34:56.123456' as datetime) + 1; ?column? ----- 2020-10-11 12:34:56 (1 row) </pre> </li> </ul> </li> </ul>

序号	MySQL数据库	GaussDB数据库	差异
			<p>将datetime类型与数值的运算结果作为函数的入参，可能导致函数的结果与MySQL也存在差异。</p> <ul style="list-style-type: none"> <li> <b>类型转换</b>                      相比较MySQL，GaussDB仅支持datetime类型与char(n)、nchar(n)、timestamp类型之间的相互转换、datetime到date、time类型的转换（仅赋值和显式转换）。不支持与binary、decimal、json、integer、unsigned integer类型之间的转换。集合等场景和复杂表达式场景下公共类型的确定原则与MySQL也不一致，参考<a href="#">数据类型转换</a>章节的描述。                 </li> <li> <b>时区</b>                      GaussDB支持datetime值中携带时区信息（时区偏移或者时区名），如'2020-01-01 12:34:56.123456 +01:00' 或者 '2020-01-01 2:34:56.123456 CST'。GaussDB会将其转换为当前服务器时区的时间。MySQL不支持（5.7版本不支持，8.0及之后的版本支持）。                 </li> <li>                     GaussDB的datetime数据类型的表字段实际上会被转换为timestamp(p) without time zone类型，查询表信息或者使用工具导出的表结构，其字段的数据类型显示的是timestamp(p) without time zone，而不是datetime。MySQL显示的是datetime(p)。                 </li> </ul>

序号	MySQL数据库	GaussDB数据库	差异
3	TIMESTAMP[(fsp)]	支持	<p>GaussDB支持timestamp数据类型，与MySQL相比规格上存在如下差异：</p> <ul style="list-style-type: none"> <li>● 输入格式 <ul style="list-style-type: none"> <li>- 只支持字符类型，不支持数值类型。如支持'2020-01-01 10:20:30.123456'或'20200101102030.123456'字符串格式，不支持如20200101102030.123456的数值类型输入。MySQL支持数值输入转换为timestamp类型。</li> <li>- 分隔符：不支持加号“+”、冒号“:”作为年、月、日之间的分隔符，其他的符号都支持。仅支持冒号“:”作为时、分、秒之间的分隔符，其他的符号都不支持。分隔符混合使用的某些场景也不支持，与MySQL也有差异，不推荐使用。MySQL支持所有符号作为分隔符。</li> <li>- 无分隔符：推荐使用完整格式'YYYYMMDDhhmiss.ffffff'。其他不完整的格式（包括超长格式）解析的规则可能与MySQL存在差异，可能报错或者解析的结果与MySQL不一致，不推荐使用。</li> </ul> </li> <li>● 输出格式 <ul style="list-style-type: none"> <li>- 统一为'YYYY-MM-DD hh:mi:ss.ffffff'的格式，格式与MySQL无差异，且不受DateStyle参数的影响。但是对于精度部分，如果最后几位为0，GaussDB不显示，MySQL会显示。</li> <li>- GaussDB在sql_mode参数不包含'strict_trans_tables'选项（定义为宽松模式，否则为严格模式）时，允许年、月、日值是0，但是输出时会按照年、月、日的顺序依次转换为合法的值得，如timestamp '0000-00-10 00:00:00' 转换为：0002-12-10 00:00:00 BC。非法输入或者超过范围时，会报warning信息，并返回0000-00-00 00:00:00值。MySQL对于包含0值年、月、日的timestamp值会原样输出。</li> </ul> </li> <li>● 取值范围 <ul style="list-style-type: none"> <li>4713-11-24 00:00:00.000000 BC ~ 294277-01-09 04:00:54.775806 AD。294277-01-09 04:00:54.775807 AD 返回的是infinity。对于超过范围的值，严格模式下GaussDB会报错，MySQL是否报错取决于使用场景。一般查询场景不报错，而</li> </ul> </li> </ul>

序号	MySQL数据库	GaussDB数据库	差异
			<p>执行DML SQL语句更改表属性的值时报错。宽松模式下GaussDB返回0000-00-00 00:00:00值，MySQL根据使用场景可能报错，也可能返回0000-00-00 00:00:00值或者null值。这个差异会导致以timestamp类型为入参的函数执行结果与MySQL也存在差异。</p> <ul style="list-style-type: none"> <li>● 精度                     <ul style="list-style-type: none"> <li>范围0~6，作为表列的类型时缺省为0，与MySQL一致。对于 timestamp[(p)] 'str' 表达式场景：                             <ul style="list-style-type: none"> <li>- GaussDB将(p)作为精度解析，缺省为6，将'str'按照p指定的精度格式化成timestamp类型。</li> <li>- MySQL将timestamp 'str'的含义与GaussDB一致，缺省精度也为6。但是将timestamp(p) 'str'解析为函数调用，p作为timestamp函数的入参，结果返回一个timestamp类型的值，'str'作为投影列的别名。</li> </ul> </li> </ul> </li> <li>● 操作符                     <ul style="list-style-type: none"> <li>- GaussDB仅支持timestamp类型之间的比较操作符“=”、“!=”、“&lt;”、“&lt;=”、“&gt;”、“&gt;=”，返回true或者false；timestamp与interval类型的加法运算，返回结果为timestamp类型；timestamp与interval类型的减法运算，返回结果为timestamp类型；timestamp类型之间的减法运算，返回结果为interval类型。</li> <li>- MySQL timestamp类型和其他数值类型运算时，会先将timestamp转换为数值类型，然后按照数值类型运算，结果也为数值类型。与GaussDB存在差异。如：                             <pre> -- MySQL: timestamp + 数值, 先将timestamp类型转换为数值20201010123456.123456,再与1相加, 结果为数值类型20201010123457.123456 mysql&gt; select timestamp '2020-10-10 12:34:56.123456' + 1; +-----+   timestamp '2020-10-10 12:34:56.123456' + 1   +-----+                  20201010123457.123456   +-----+ 1 row in set (0.00 sec)  -- GaussDB: timestamp + 数值, 数值类型会转换为interval类型 1 day, 然后相加得到新的timestamp。 gaussdb=# select timestamp '2020-10-10 12:34:56.123456' + 1; </pre> </li> </ul> </li> </ul>

序号	MySQL数据库	GaussDB数据库	差异
			<div style="border: 1px solid gray; padding: 5px; margin-bottom: 10px;">                     ?column?                      -----                      2020-10-11 12:34:56.123456                      (1 row)                 </div> <p>将timestamp类型与数值的运算结果作为函数的入参，可能导致函数的结果与MySQL也存在差异。</p> <ul style="list-style-type: none"> <li> <b>类型转换</b>                      相比较MySQL，GaussDB仅支持timestamp类型与char(n)、varchar(n)、datetime类型之间的相互转换、timestamp到date、time类型的转换（仅赋值和显式转换）。不支持与binary、decimal、json、integer、unsigned integer类型之间的转换。集合等场景和复杂表达式场景下公共类型的确定原则与MySQL也不一致，参考<a href="#">数据类型转换</a>章节的描述。                 </li> <li> <b>时区</b>                      GaussDB支持timestamp值中携带时区信息（时区偏移或者时区名），如 '2020-01-01 12:34:56.123456 +01:00' 或者 '2020-01-01 2:34:56.123456 CST'。GaussDB会将其转换为当前服务器时区的时间。如果更改服务器时区，timestamp类型的值输出时会转换为更改后时区的时间戳。MySQL不支持（5.7版本不支持，8.0及之后的版本支持）。                 </li> <li>                     GaussDB的timestamp数据类型的表字段实际上会被转换为timestamp(p) with time zone类型，查询表信息或者使用工具导出的表结构，其字段的数据类型显示的是timestamp(p) with time zone，而不是timestamp。MySQL显示的是timestamp(p)。                 </li> </ul>

序号	MySQL数据库	GaussDB数据库	差异
4	TIME[(fsp)]	支持	<p>GaussDB支持time数据类型，与MySQL相比规格上存在如下差异：</p> <ul style="list-style-type: none"> <li>● 输入格式 <ul style="list-style-type: none"> <li>- 只支持字符类型，不支持数值类型。如支持'1 10:20:30'或'102030'字符串格式，不支持102030数值输入。MySQL支持数值输入转换为time类型。</li> <li>- 分隔符：GaussDB仅支持冒号“:”作为时、分、秒之间的分隔符，其他的符号都不支持。MySQL支持所有的符号作为分隔符。</li> <li>- 无分隔符：推荐使用完整格式，如'hhmiss.ffffff'。其他不完整的格式（包括超长格式）解析的规则可能与MySQL存在差异，可能报错或者解析的结果与MySQL不一致，不推荐使用。</li> <li>- 分、秒、精度输入负数时，GaussDB数据库可能会忽略第一个负数开始的部分，涉及的部分解析为0，如：'00:00:-10'解析结果为'00:00:00'。也可能报错，如：'00:00:-10000'会解析报错。取决于输入值的范围。而MySQL数据库统一报错。</li> </ul> </li> <li>● 输出格式 <ul style="list-style-type: none"> <li>统一为hh:mi:ss.ffffff的格式，格式与MySQL无差异。但是对于精度部分，如果最后几位为0，GaussDB不显示，MySQL会显示。</li> </ul> </li> <li>● 取值范围 <ul style="list-style-type: none"> <li>-838:59:59.000000 ~ 838:59:59.000000，与MySQL一致。对于超过范围的值，宽松模式下GaussDB无论是查询还是insert/update等DML操作，返回的值都是就近的边界值：-838:59:59或838:59:59。MySQL是查询时报错，DML操作返回的值才是就近边界值，场景上存在差异。此差异会导致time类型作为函数入参时，函数返回的结果也存在差异。</li> </ul> </li> <li>● 精度 <ul style="list-style-type: none"> <li>范围0~6，作为表列的类型时缺省为0，与MySQL一致。对于time(p) 'str'表达式场景，GaussDB将(p)作为精度解析，缺省为6，将'str'按照p指定的精度格式化time类型。MySQL是解析为time函数，p是入参，'str'是投影列的别名。</li> </ul> </li> <li>● 操作符</li> </ul>

序号	MySQL数据库	GaussDB数据库	差异
			<p>- GaussDB仅支持time类型之间的比较操作符“=”、“!=”、“&lt;”、“&lt;=”、“&gt;”、“&gt;=”，返回true或者false；time与interval类型的加法运算，返回结果为time类型；time与interval类型的减法运算，返回结果为time类型；time类型之间的减法运算，返回结果为interval类型。</p> <p>- MySQL time类型和其他数值类型运算时，会先将time转换为数值类型，然后按照数值类型运算，结果也为数值类型。与GaussDB存在差异。如：</p> <pre>-- MySQL: time + 数值, 先将time类型转换为数值 123456,再与1相加, 结果为数值类型123457 mysql&gt; select time '12:34:56' + 1; +-----+   time '12:34:56' + 1   +-----+            123457   +-----+ 1 row in set (0.00 sec)</pre> <p>-- GaussDB: time + 数值, 数值类型会转换为interval类型 1 day, 然后相加得到新的time, 由于是加了24小时, 得到的仍然是12:34:56。</p> <pre>gaussdb=# select time '12:34:56' + 1; ?column? ----- 12:34:56 (1 row)</pre> <p>将time类型与数值的运算结果作为函数的入参, 可能导致函数的结果与MySQL也存在差异。</p> <ul style="list-style-type: none"> <li>● 类型转换 相比较MySQL, GaussDB仅支持time类型与char(n)、nchar(n)类型之间的相互转换、datetime、timestamp到time类型的转换。不支持与binary、decimal、date、json、integer、unsigned integer类型之间的转换。集合等场景和复杂表达式场景下公共类型的确定原则与MySQL也不一致, 参考<a href="#">数据类型转换</a>章节的描述。</li> </ul>



序号	MySQL数据库	GaussDB数据库	差异
5	YEAR[(4)]	支持	<p>GaussDB支持year数据类型，与MySQL相比规格上存在如下差异：</p> <ul style="list-style-type: none"> <li>操作符 <ul style="list-style-type: none"> <li>GaussDB仅支持year类型之间的比较操作符“=”、“!=”、“&lt;”、“&lt;=”、“&gt;”、“&gt;=”，返回true或者false。</li> <li>GaussDB仅支持year类型与int4类型之间的算术操作符“+”、“-”，返回整型值，MySQL是返回无符号整型值。</li> </ul> </li> <li>类型转换 <p>相比较MySQL，GaussDB仅支持year类型与int4类型的转换，仅支持int4、varchar、numeric、date、time、timestamp、timestampz类型到year类型的转换。集合等场景和复杂表达式场景下公共类型的确定原则与MySQL也不一致，参考<a href="#">数据类型转换</a>章节的描述。</p> </li> </ul>
6	INTERVAL	支持	<p>GaussDB支持INTERVAL数据类型，但INTERVAL在MySQL中为表达式，同时存在以下差异：</p> <ul style="list-style-type: none"> <li>不支持字符串类型的日期输入作为运算，如：select '2023-01-01' + interval 1 day。</li> <li>不支持interval expr unit语法中，expr为负整数或浮点数的输入，如：select date'2023-01-01' + interval -1 day。</li> <li>不支持interval expr unit语法中，expr为运算表达式的输入，如：select date'2023-01-01' + interval 4/2 day。</li> <li>interval表达式参与运算时，返回值固定为datetime类型，MySQL为datetime或date类型。运算的逻辑与原有GaussDB保持一致，与MySQL有差异。</li> <li>interval expr unit语法中，expr数值支持的范围会根据unit单位的不同有所差异，最大可支持的范围为[-2147483648, 2147483647]。超过范围时，严格模式报error，宽松模式报warning并返回0值。</li> <li>interval expr unit语法中，expr指定的字段数量大于unit预期的字段数量时，在严格模式，报error；在宽松模式，报warning并返回0值。如unit取值为DAY_HOUR，预期的字段数量为2，expr取值为'1-2-3'，字段数量为3。</li> </ul>

### 8.2.2.2.3 字符串数据类型

表 8-27 字符串数据类型

序号	MySQL数据库	GaussDB数据库	差异
1	CHAR[(M)]	支持	<ul style="list-style-type: none"> <li>● 输入格式                             <ul style="list-style-type: none"> <li>- GaussDB自定义函数参数和返回值不支持长度校验，存储过程参数不支持长度校验，同时也不支持在 PAD_CHAR_TO_FULL_LENGTH打开时补齐正确的空格，MySQL支持。</li> <li>- GaussDB不支持转义字符输入，不支持""双引号输入，MySQL支持。</li> </ul> </li> <li>● 语法                             <p>GaussDB的 Cast ( expr as char ) 语法无法根据输入的字符串长度转成对应的类型，只支持转成varchar类型。不支持 cast( ' ' as char) 和cast( ' ' as char(0))将空串转成char(0)类型。MySQL支持按长度转成对应的类型。</p> </li> <li>● 操作符                             <ul style="list-style-type: none"> <li>- GaussDB能正常转成浮点型的字符串与整型值加减乘除求余，返回值是整型值，MySQL是返回浮点型。</li> <li>- GaussDB除以0会报错，MySQL返回 null。</li> <li>- “~”：GaussDB返回负数，MySQL返回8字节无符号整数。</li> <li>- “^”：GaussDB表示次方幂，MySQL表示按位异或。</li> </ul> </li> </ul>

序号	MySQL数据库	GaussDB数据库	差异
2	VARCHAR(M)	支持	<ul style="list-style-type: none"> <li>● 输入格式：                             <ul style="list-style-type: none"> <li>- GaussDB的自定义函数参数和返回值不支持长度校验，存储过程参数不支持长度校验，MySQL支持。</li> <li>- GaussDB的自定义函数和存储过程中的临时变量支持长度校验以及严格宽松模式下的报错和截断告警，MySQL不支持。</li> <li>- GaussDB不支持转义字符输入，不支持""双引号输入，MySQL支持。</li> </ul> </li> <li>● 操作符                             <ul style="list-style-type: none"> <li>- GaussDB能正常转成浮点型的字符串与整型值加减乘除求余，返回值是整型值，MySQL是返回浮点型。</li> <li>- GaussDB除以0会报错，MySQL返回null。</li> <li>- “~”：GaussDB返回负数，MySQL返回8字节无符号整数。</li> <li>- “^”：GaussDB表示次方幂，MySQL表示按位异或。</li> </ul> </li> </ul>
3	TINYTEXT	支持	<ul style="list-style-type: none"> <li>● 输入格式                             <ul style="list-style-type: none"> <li>- GaussDB不支持长度限制255字节（最大不超过1G），不支持超限后，根据严格宽松模式报错和截断告警。MySQL支持。</li> <li>- GaussDB不支持转义字符输入，不支持""双引号输入，MySQL支持。</li> </ul> </li> <li>● 操作符                             <ul style="list-style-type: none"> <li>- GaussDB能正常转成浮点型的字符串与整型值加减乘除求余，返回值是整型值，MySQL是返回浮点型。</li> <li>- GaussDB除以0会报错，MySQL返回null。</li> <li>- “~”：GaussDB返回负数，MySQL返回8字节无符号整数。</li> <li>- “^”：GaussDB表示次方幂，MySQL表示按位异或。</li> </ul> </li> </ul>

序号	MySQL数据库	GaussDB数据库	差异
4	TEXT	支持	<ul style="list-style-type: none"> <li>● 输入格式                             <ul style="list-style-type: none"> <li>- GaussDB不支持长度限制65535字节（最大不超过1G），不支持超限后，根据严格宽松模式报错和截断告警。MySQL支持。</li> <li>- GaussDB不支持转义字符输入，不支持""双引号输入，MySQL支持。</li> </ul> </li> <li>● 操作符                             <ul style="list-style-type: none"> <li>- GaussDB能正常转成浮点型的字符串与整型值加减乘除求余，返回值是整型值，MySQL是返回浮点型。</li> <li>- GaussDB除以0会报错，MySQL返回null。</li> <li>- “~”：GaussDB返回负数，MySQL返回8字节无符号整数。</li> <li>- “^”：GaussDB表示次方幂，MySQL表示按位异或。</li> </ul> </li> </ul>
5	MEDIUMTEXT	支持	<ul style="list-style-type: none"> <li>● 输入格式                             <ul style="list-style-type: none"> <li>- GaussDB不支持长度限制16777215字节（最大不超过1G），不支持超限后，根据严格宽松模式报错和截断告警。MySQL支持。</li> <li>- GaussDB不支持转义字符输入，不支持""双引号输入，MySQL支持。</li> </ul> </li> <li>● 操作符                             <ul style="list-style-type: none"> <li>- GaussDB能正常转成浮点型的字符串与整型值加减乘除求余，返回值是整型值，MySQL是返回浮点型。</li> <li>- GaussDB除以0会报错，MySQL返回null。</li> <li>- “~”：GaussDB返回负数，MySQL返回8字节无符号整数。</li> <li>- “^”：GaussDB表示次方幂，MySQL表示按位异或。</li> </ul> </li> </ul>

序号	MySQL数据库	GaussDB数据库	差异
6	LONGTEXT	支持	<ul style="list-style-type: none"> <li>输入格式 <ul style="list-style-type: none"> <li>GaussDB只支持不超过1G，MySQL支持4G-1字节长度。</li> <li>GaussDB不支持转义字符输入，不支持""双引号输入，MySQL支持。</li> </ul> </li> <li>操作符 <ul style="list-style-type: none"> <li>GaussDB能正常转成浮点型的字符串与整型值加减乘除求余，返回值是整型值，MySQL是返回浮点型。</li> <li>GaussDB除以0会报错，MySQL返回null。</li> <li>“~”：GaussDB返回负数，MySQL返回8字节无符号整数。</li> <li>“^”：GaussDB表示次方幂，MySQL表示按位异或。</li> </ul> </li> </ul>
7	ENUM('value 1','value2',...)	不支持	-
8	SET('value1','value2',...)	支持	-

#### 8.2.2.2.4 二进制数据类型

表 8-28 二进制数据类型

序号	MySQL数据库	GaussDB数据库	差异
1	BINARY[(M)]	不支持	-
2	VARBINARY(M)	不支持	-
3	TINYBLOB	支持	<ul style="list-style-type: none"> <li>取值范围：不支持长度限制255字节（最大不超过1G），不支持超限后，根据严格宽松模式报错和截断告警。</li> <li>输入格式：不支持转义字符输入，不支持""双引号输入。</li> <li>操作符：不支持算数运算符“+”、“-”、“*”、“/”、“%”；不支持常用逻辑运算符或、与、非（“  ”、“&amp;&amp;”、“!”）；不支持常用位运算符“~”、“&amp;”、“ ”、“^”。</li> </ul>

序号	MySQL数据库	GaussDB数据库	差异
4	BLOB	支持	<ul style="list-style-type: none"> <li>取值范围：不支持长度限制65535字节（最大不超过1G），不支持超限后，根据严格宽松模式报错和截断告警。MySQL支持。</li> <li>输入格式：不支持转义字符输入，不支持""双引号输入。</li> <li>操作符：不支持算数运算符“+”、“-”、“*”、“/”、“%”；不支持常用逻辑运算符与、或、非（“  ”、“&amp;&amp;”、“!”）；不支持常用位运算符“~”、“&amp;”、“ ”、“^”。</li> </ul>
5	MEDIUMBLOB	支持	<ul style="list-style-type: none"> <li>取值范围：不支持长度限制16777215字节（最大不超过1G），不支持超限后，根据严格宽松模式报错和截断告警。MySQL支持。</li> <li>输入格式：不支持转义字符输入，不支持""双引号输入。</li> <li>操作符：不支持算数运算符“+”、“-”、“*”、“/”、“%”；不支持常用逻辑运算符与、或、非（“  ”、“&amp;&amp;”、“!”）；不支持常用位运算符“~”、“&amp;”、“ ”、“^”。</li> </ul>
6	LONGBLOB	支持	<ul style="list-style-type: none"> <li>取值范围：只支持不超过1G，具体范围参照bytea数据类型集中式和分布式规格。</li> <li>输入格式：不支持转义字符输入，不支持""双引号输入。</li> <li>操作符：不支持算数运算符“+”、“-”、“*”、“/”、“%”；不支持常用逻辑运算符或、与、非（“  ”、“&amp;&amp;”、“!”）；不支持常用位运算符“~”、“&amp;”、“ ”、“^”。</li> </ul>
7	BIT[(M)]	不支持	-

### 8.2.2.2.5 JSON 数据类型

表 8-29 JSON 数据类型

序号	MySQL数据库	GaussDB数据库
1	JSON	不完全兼容

### 8.2.2.2.6 数据类型支持的属性

表 8-30 数据类型支持的属性

序号	MySQL数据库	GaussDB数据库
1	NULL	支持
2	NOT NULL	支持
3	DEFAULT	支持
4	ON UPDATE	支持
4	PRIMARY KEY	支持
5	AUTO_INCREMENT	支持
6	CHARACTER SET name	支持
7	COLLATE name	支持

### 8.2.2.2.7 数据类型转换

不同的数据类型之间支持转换。有如下场景涉及到数据类型转换：

- 操作符（比较操作符、运算操作符等）的操作数的数据类型不一致。常见于查询条件或者关联条件中的比较运算。
- 函数调用时实参和形参的数据类型不一致。
- DML语句要更新（包括insert、update、merge、replace等）的目标列，数据的类型和列的定义类型不一致。
- 显式的类型转换：cast(expr as datatype)，将expr表达式类型转换为datatype类型。
- 集合运算（UNION、MINUS、EXCEPT、INTERSECT）确定最终投影列的目标数据类型后，各个SELECT查询的投影列的类型和目标数据类型不一致。
- 其他表达式计算场景，根据不同表达式的数据类型，来决定用于比较或者最终结果的目标数据类型。
  - DECODE
  - CASE WHEN
  - lexpr [ NOT ] IN (expr\_list)
  - BETWEEN AND
  - JOIN USING(a,b)
  - GREATEST和LEAST
  - NVL 和 COALESCE

GaussDB和MySQL数据库对于数据类型转换、转换的目标数据类型有着完全不同的规则。如下示例体现了两者处理的差异：

```
-- MySQL: in 执行结果为0，表示false。根据规则，会将'1970-01-01'与列表中的表达式依次比较，结果都为0，因此最终结果为0。  
mysql> select '1970-01-01' in ('1970-01-02', 1, '1970-01-02');
```

```
+-----+
| '1970-01-01' in ('1970-01-02', 1, '1970-01-02') |
+-----+
|                                0 |
+-----+

-- GaussDB: in 执行结果为true, 与MySQL结果相反: 根据规则选定的公共类型为int类型, 因此将左表达式
'1970-01-01'转换为int类型与列表中的表达式转换为int类型后的值依次比较。
-- '1970-01-01'和'1970-01-02'转换为int类型时都为1970 (兼容MySQL模式下, 转换时遇到非法字符后忽略, 将
前面部分转换为int类型), 比较结果为相等, 因此返回结果为true。
gaussdb=# select '1970-01-01' in ('1970-01-02', 1::int, '1970-01-02') as result;
result
-----
t
(1 row)
```

### 1. 数据类型转换规则的差异:

- GaussDB数据库对于不同数据类型之间的转换规则有明确的定义:
  - 是否支持转换: pg\_cast系统表中是否定义两种类型的转换路径, 没有定义则不支持。
  - 支持转换的场景: 支持任意场景转换、仅支持显式 ( cast表达式 ) 转换、仅支持赋值时转换。不支持的场景下即使定义了转换路径, 也不能进行数据类型转换。
- MySQL数据库支持任意两种数据类型之间做转换。

由于存在以上差异, 基于MySQL数据库的应用程序向GaussDB数据库迁移时, SQL语句可能由于不支持不同数据类型之间的转换而报错。或者支持转换的场景下, 转换的规则有差异导致SQL语句执行的结果不同。

推荐的做法是: SQL语句中尽量使用相同的数据类型做比较或者赋值等操作, 避免因数据类型转换导致非预期结果或者性能损耗。

### 2. 选择目标数据类型的规则差异:

对于有些场景, 比较的数据类型或者返回的数据类型需要综合考虑多个表达式的类型才能确定。比如UNION运算中, 不同SELECT语句中相同位置的投影列具有不同的数据类型, 查询结果的最终数据类型, 需要由各个SELECT语句投影列的数据类型共同确定。

确定目标数据类型的规则, GaussDB数据库和MySQL数据库存在体系上的差异。

- GaussDB数据库规则:
  - 操作符的操作数类型不一致时, 并不是将操作数的类型统一转换为目标类型再计算。而是直接注册两个数据类型的操作符, 操作符处理中定义两个不同类型的处理规则。此方式不存在类型隐式转换, 但自定义的处理规则隐含了转换的操作。
  - 集合运算和表达式场景, 确定目标数据类型的规则:
    - 如果所有类型都相同, 则此类型即为目标类型。
    - 两个数据类型如果不同, 检查数据类型是否属于同一种类的数据类型, 如数值类型、字符类型、日期时间类型等。不属于同一种类的数据类型, 无法确定目标类型, 此时SQL语句执行会报错。
    - 对于category属性 ( 在pg\_type系统表中定义 ) 相同的数据类型, 具有preferred属性 ( 在pg\_type系统表中定义 ) 的数据类型会被选为目标类型。或者操作数1能转换为操作数2 ( 没有转换路径 ), 而操作数2无法转



换为操作数1或数值类型优先级小于操作数2，则选择操作数2作为目标类型。

- 如果涉及到3个及以上的数据类型，确定目标类型的规则为：  
`common_type(type1,type2,type3) = common_type(common_type(type1,type2),type3)`，依次迭代处理，得到最终的结果。
- 对于IN和NOT IN表达式，如果根据以上规则无法确认目标类型，会将 `lexpr`与`expr_list`中每一个表达式单独按照等值操作符(=)逐个比较。
- 精度的确定：以最终选定的表达式的精度作为最终结果。
- MySQL数据库规则：
  - 操作符的操作数类型不一致时，先按照如下规则确定目标类型。确定后将类型不一致的操作数转换成目标类型后再做处理。
    - 两个参数都是string类型，则都按照string类型比较。
    - 两个参数都是integer类型，则都按照integer类型比较。
    - 十六进制数值如果不与数值比较，则当做二进制字符串比较。
    - 一个参数是datetime/timestamp类型，另一个参数是常量，将常量转换为时间戳类型然后比较。
    - 如果其中一个参数是decimal类型，比较时使用的数据类型取决于另外一个参数。另外一个为decimal或者integer类型时，按照decimal类型；另外一个为其他类型，按照real类型比较。
    - 其他场景都转换为 real 类型后比较。
  - 集合运算和表达式场景，确定目标数据类型的规则如下：
    - 建立任意两个类型之间的目标类型矩阵。给定两个类型，通过矩阵即可以确定目标类型。
    - 如果涉及到3个及以上的数据类型，确定目标类型的规则为：  
`common_type(type1,type2,type3) = common_type(common_type(type1,type2),type3)`，依次迭代处理，得到最终的结果。
    - 如果目标类型是integer类型，且各个表达式类型包含有符号和无符号的混合场景，则会将类型提升到更高精度的integer类型。符号的确定：所有表达式都是无符号时，结果才为无符号，否则结果为有符号。
    - 精度确定：以表达式中的最大精度作为最终结果。

从以上规则可知：GaussDB和MySQL数据库在数据类型的转换规则上有很大差异，不能直接对比。在上述场景下，SQL语句的执行结果可能和MySQL数据库不一致。当前版本推荐各个表达式使用相同的类型，或提前使用cast转换成需要的类型来规避差异。

### 8.2.2.3 系统函数

GaussDB数据库兼容绝大多数MySQL的系统函数，但存在部分差异。如未列出，函数行为默认为GaussDB原生行为。

 说明

GaussDB MySQL兼容性的绝大部分系统函数目前均存在返回值与MySQL精度不一致（结果后面0的位数）的问题，这是由于部分数据类型在某些场景下仍存在精度丢失问题，无法正确获取精度，导致目前部分函数未完全做到适配。

## 8.2.2.3.1 流量控制函数

表 8-31 流量控制函数列表

序号	MySQL数据库	GaussDB数据库	差异
1	IF()	支持	<ul style="list-style-type: none"><li>• expr1入参仅支持bool类型。非bool类型入参若不能转换为bool类型则报错。</li><li>• 若expr2、expr3两入参类型不同且两类型间不存在隐式转换函数则报错。</li><li>• 两入参类型相同时，返回该入参类型。</li><li>• 若expr2、expr3两入参类型分别为NUMERIC、STRING或TIME其中一个时，输出为text类型，MySQL输出为varchar类型。</li></ul>
2	IFNULL()	支持	<ul style="list-style-type: none"><li>• 若expr1、expr2两入参类型不同且两类型间不存在隐式转换函数则报错。</li><li>• 两入参类型相同时，返回该入参类型。</li><li>• 若expr1、expr2两入参类型范畴分别为NUMERIC、STRING或TIME其中一个时，输出为text类型，MySQL输出为varchar类型。</li><li>• 两入参类型第一个入参为float4，另一个为bigint或unsigned bigint时返回double类型，MySQL返回float类型。</li></ul>

序号	MySQL 数据库	GaussDB 数据库	差异
3	NULLIF()	支持	<ul style="list-style-type: none"> <li>GaussDB中NULLIF()类型推导遵从以下逻辑： <ol style="list-style-type: none"> <li>如果两个参数的数据类型不同，且两入参类型存在等值比较操作符，则返回对应等值操作符对应的左值类型，否则会对两入参类型进行强制类型兼容。</li> <li>若强制类型兼容后，存在等值比较操作符，则返回强制类型兼容后对应等值操作符的左值类型。</li> <li>若强制类型兼容后，仍找不到对应等值操作符，则报错。  <pre>--两入参类型存在等值比较操作符 gaussdb=# select pg_typeof(nullif(1::int2, 2::int8)); pg_typeof ----- smallint (1 row) --两入参类型不存在等值比较操作符，但在强制类型兼容后可以找到等值比较操作符 gaussdb=# select pg_typeof(nullif(1::int1, 2::int2)); pg_typeof ----- bigint (1 row) --两入参类型不存在等值比较操作符，且强制类型兼容后也不存在等值比较操作符 gaussdb=# SELECT nullif(1::bit, '1'::MONEY); ERROR: operator does not exist: bit = money LINE 1: SELECT nullif(1::bit, '1'::MONEY);                         ^ HINT: No operator matches the given name and argument type(s). You might need to add explicit type casts. CONTEXT: referenced column: nullif</pre> </li> </ol> </li> <li>MySQL输出类型仅与第一个入参类型有关： <ol style="list-style-type: none"> <li>第一个入参为tinyint、smallint、mediumint、int、bool时，输出为int类型。</li> <li>第一个入参为bigint时，输出为bigint类型。</li> <li>第一个入参为unsigned tinyint、unsigned smallint、unsigned mediumint、unsigned int、bit时，输出为unsigned int类型。</li> <li>第一个入参为unsigned bigint时，输出为unsigned bigint。</li> <li>第一个入参为浮点型即float、double、real时，输出为double类型。</li> <li>第一个入参类型为decimal或numeric类型时，输出为decimal类型。</li> <li>第一个入参类型为时间类型或字符串类型即date、time、date、datetime、timestamp、char、varchar以及tinytext、enum、set时，输出为varchar类型。</li> <li>第一个入参类型为text、mediumtext、longtext时，输出为longtext类型。</li> </ol> </li> </ul>

序号	MySQL 数据库	GaussDB 数据库	差异
			9. 第一个入参类型为tinyblob时，输出为varbinary类型。 10. 第一个入参类型为mediumblob或longblob时，输出为longblob类型。 11. 第一个入参为blob时，输出为blob类型。
4	ISNULL()	支持	GaussDB中返回值为boolean类型的t或f，MySQL中返回值为int类型的1或0。

### 8.2.2.3.2 日期和时间函数

#### 📖 说明

以下为GaussDB MySQL兼容性日期时间函数公共说明。

- 开发者指南中函数入参为时间类型表达式的情况：

时间类型表达式主要包括TEXT、DATETIME、DATE或TIME，但所有可以隐式转换为时间表达式的类型都可以作为入参，比如数字类型可以通过先隐式转化为TEXT，再作为时间类型表达式生效。

但是生效的情况根据函数有所不同，比如：DATEDIFF由于只计算日期的差值，所以时间表达式会当做日期来解析。TIMESTAMPDIFF由于计算时间差值时是根据UNIT来决定的，所以会根据UNIT选择将时间表达式当做DATE或者TIME或者DATETIME来解析。

- 函数入参为无效日期的情况：

一般而言，日期时间函数支持DATE、DATETIME的范围和MySQL保持一致。DATE支持的范围为'0000-01-01'到'9999-12-31'，DATETIME支持的范围为'0000-01-01 00:00:00'到'9999-12-31 23:59:59'。虽然GaussDB支持的DATE、DATETIME范围大于MySQL，但是越界仍然算无效日期。

大部分时间函数会告警并返回NULL，只有能通过cast正常转换的日期，才是正常合理的日期。

GaussDB的日期时间函数的大部分功能场景与MySQL一致，但仍有差异，一些差异如下：

- 函数入参为NULL时，函数返回NULL，无warning或error告警。这些函数包括：  
 from\_days、date\_format、str\_to\_date、datediff、timestampdiff、date\_add、subtime、month、time\_to\_sec、to\_days、to\_seconds、dayname、monthname、convert\_tz、sec\_to\_time、addtime、adddate、date\_sub、timediff、last\_day、weekday、from\_unixtime、unix\_timestamp、subdate、day、year、weekofyear、dayofmonth、dayofyear、week、yearweek、dayofweek、time\_format、hour、minute、second、microsecond、quarter、get\_format、extract、makedate、period\_add、timestampadd、period\_diff、utc\_time、utc\_timestamp、maketime、curtime

示例：

```
gaussdb=# select day(null);
 day
-----
(1 row)
```

- 纯数字入参个别函数与MySQL有差异，不带引号的数字入参统一转成text入参来处理。

示例:

```
gaussdb=# select day(19231221.123141);
WARNING: Incorrect datetime value: "19231221.123141"
CONTEXT: referenced column: day
day
-----
(1 row)
```

- 时间日期运算函数: `adddate`、`subdate`、`date_add`、`date_sub`。当运算后的日期结果在范围[0000-01-01, 0001-01-01]中时, 正常返回结果, MySQL返回'0000-00-00'。

示例:

```
gaussdb=# select subdate('0001-01-01', interval 1 day);
subdate
-----
0000-12-31
(1 row)
```

- 对于日期和时间函数的`date`或`datetime`类型入参, 含有0月或0日时为非法值, 在严格模式下报`error`; 在宽松模式, 当输入为字符串或数字时, 报`warning`, 输入为`date`或`datetime`类型时视为上一年12月或上一月最后一日处理。

对于`cast`函数, 转换为`date`、`datetime`时, 严格模式下会报`error`。宽松模式下不会报`warning`, 而是视为上一年12月或上一月最后一日处理, 需要注意此区别。MySQL对于包含0年、0月或0日的情况会原样输出。

示例:

```
gaussdb=# select adddate('2023-01-00', 1);-- 严格模式
ERROR: Incorrect datetime value: "2023-01-00"
CONTEXT: referenced column: adddate

gaussdb=# select adddate('2023-01-00', 1);-- 宽松模式
WARNING: Incorrect datetime value: "2023-01-00"
CONTEXT: referenced column: adddate
adddate
-----
(1 row)

gaussdb=# select adddate(date'2023-00-00', 1);-- 宽松模式
adddate
-----
2022-12-01
(1 row)

gaussdb=# select cast('2023/00/00' as date);-- 宽松模式
date
-----
2022-11-30
(1 row)

gaussdb=# select cast('0000-00-00' as datetime);-- 宽松模式
timestamp
-----
0000-00-00 00:00:00
(1 row)
```

- 若函数入参为`numeric`数据类型, 在非法输入的情况下不会产生报错, 会把入参当做0值处理。

示例:

```
gaussdb=# select from_unixtime('aa');
from_unixtime
-----
```

- ```

1970-01-01 08:00:00
(1 row)

```
- 最多保留6位小数，不保留后置都为0的小数。

示例：

```

gaussdb=# select from_unixtime('1234567899.000000');
           from_unixtime
-----
2009-02-14 07:31:39
(1 row)

```
  - 时间函数参数为字符串时，只保证年月日之间使用“-”分隔，时分秒之间使用“:”分隔时结果正确。

示例：

```

gaussdb=# select adddate('20-12-12',interval 1 day);
           adddate
-----
2020-12-13
(1 row)

```
  - 在MySQL中，当函数的返回值为varchar时，在GaussDB中，函数对应的返回值为text。

-- GaussDB中函数的返回值。

```

gaussdb=# SELECT pg_typeof(adeddate('2023-01-01', 1));
           pg_typeof
-----
text
(1 row)

```

-- MySQL中函数的返回值。

```

mysql> CREATE VIEW v1 AS SELECT adddate('2023-01-01', 1);
Query OK, 0 rows affected (0.00 sec)

mysql> DESC v1;
+-----+-----+-----+-----+-----+-----+
| Field          | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| adddate('2023-01-01', 1) | varchar(29) | YES  |     | NULL    |      |
+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

```

表 8-32 日期与和时间函数列表

| 序号 | MySQL数据库  | GaussDB数据库 | 差异                                                                   |
|----|-----------|------------|----------------------------------------------------------------------|
| 1  | ADDDATE() | 支持         | 此函数的表现会因为interval表达式的差异与MySQL有差异，具体可见 <a href="#">INTERVAL差异说明</a> 。 |

| 序号 | MySQL数据库  | GaussDB数据库 | 差异                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
|----|-----------|------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 2  | ADDTIME() | 支持         | <ul style="list-style-type: none"> <li>• MySQL对第二入参为DATETIME样式字符串返回NULL，GaussDB可以计算。</li> <li>• 入参取值范围为['0001-01-01 00:00:00', 9999-12-31 23:59:59.999999]。</li> <li>• MySQL中ADDTIME函数如果第一个参数是动态参数（例如在预准备语句中），则返回类型为TIME。否则，函数的解析类型派生自第一个参数的解析类型。GaussDB中ADDTIME函数的返回值规则如下： <ul style="list-style-type: none"> <li>- 第一个入参为date，第二个入参为date，返回值为time。</li> <li>- 第一个入参为date，第二个入参为text，返回值为text。</li> <li>- 第一个入参为date，第二个入参为datetime，返回值为time。</li> <li>- 第一个入参为date，第二个入参为time，返回值为time。</li> <li>- 第一个入参为text，第二个入参为date，返回值为text。</li> <li>- 第一个入参为text，第二个入参为text，返回值为text。</li> <li>- 第一个入参为text，第二个入参为datetime，返回值为text。</li> <li>- 第一个入参为text，第二个入参为time，返回值为text。</li> <li>- 第一个入参为datetime，第二个入参为date，返回值为datetime。</li> <li>- 第一个入参为datetime，第二个入参为text，返回值为text。</li> <li>- 第一个入参为datetime，第二个入参为datetime，返回值为datetime。</li> <li>- 第一个入参为datetime，第二个入参为time，返回值为datetime。</li> <li>- 第一个入参为time，第二个入参为date，返回值为time。</li> </ul> </li> </ul> |

| 序号 | MySQL数据库                                  | GaussDB数据库 | 差异                                                                                                                                                                                                    |
|----|-------------------------------------------|------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|    |                                           |            | <ul style="list-style-type: none"> <li>- 第一个入参为time, 第二个入参为text, 返回值为text。</li> <li>- 第一个入参为time, 第二个入参为datetime, 返回值为time。</li> <li>- 第一个入参为time, 第二个入参为time, 返回值为time。</li> </ul>                   |
| 3  | CONVERT_TZ()                              | 支持         | -                                                                                                                                                                                                     |
| 4  | CURDATE()                                 | 支持         | -                                                                                                                                                                                                     |
| 5  | CURRENT_DATE(),<br>CURRENT_DATE           | 支持         | -                                                                                                                                                                                                     |
| 6  | CURRENT_TIME(),<br>CURRENT_TIME           | 支持         | GaussDB的按精度输出的时间值（小数点后的值）是四舍五入的；MySQL是直接截断的。GaussDB按精度输出的时间值（小数点后的值）末尾0都不显示；MySQL会显示。GaussDB只支持输入[0,6]范围内的整型值，作为返回时间的精度，其他均报错；MySQL的精度值有效值是[0,6]，但是输入的整型值内部会对256求余（例257，会返回精度1的时间值）。                  |
| 7  | CURRENT_TIMESTAMP(),<br>CURRENT_TIMESTAMP | 支持         | GaussDB的按精度输出的时间值（小数点后的值）是四舍五入的；MySQL是直接截断的。GaussDB按精度输出的时间值（小数点后的值）末尾0都不显示；MySQL会显示。GaussDB只支持输入[0,6]范围内的整型值，作为返回时间的精度，超过6的整型值，会告警并按照精度6输出时间值；MySQL的精度值有效值是[0,6]，但是输入的整型值内部会对256求余（例257，会返回精度1的时间值）。 |



| 序号 | MySQL数据库        | GaussDB数据库 | 差异                                                                                                                                                                                                                                                               |
|----|-----------------|------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 8  | CURTIME()       | 支持         | GaussDB此函数输入字符串或者非整型值，会被隐式转成整型，然后再校验精度，[0,6]范围之外的会报错，范围之内会正常输出时间值；MySQL直接报错。GaussDB的按精度输出的时间值（小数点后的值）是四舍五入的；MySQL是直接截断的。GaussDB按精度输出的时间值（小数点后的值）末尾0都不显示；MySQL会显示，GaussDB只支持输入[0,6]范围内的整型值，作为返回时间的精度，其他均报错；MySQL的精度值有效值是[0,6]，但是输入的整型值内部会对256求余（例257，会返回精度1的时间值）。 |
| 9  | YEARWEEK()      | 支持         | -                                                                                                                                                                                                                                                                |
| 10 | DATE_ADD()      | 支持         | 此函数的表现会因为interval表达式的差异与MySQL有差异，具体可见 <a href="#">INTERVAL差异说明</a> 。                                                                                                                                                                                             |
| 11 | DATE_FORMAT()   | 支持         | -                                                                                                                                                                                                                                                                |
| 12 | DATE_SUB()      | 支持         | 此函数的表现会因为interval表达式的差异与MySQL有差异，具体可见 <a href="#">INTERVAL差异说明</a> 。                                                                                                                                                                                             |
| 13 | DATEDIFF()      | 支持         | -                                                                                                                                                                                                                                                                |
| 14 | DAY()           | 支持         | -                                                                                                                                                                                                                                                                |
| 15 | DAYNAME()       | 支持         | -                                                                                                                                                                                                                                                                |
| 16 | DAYOFMONTH()    | 支持         | -                                                                                                                                                                                                                                                                |
| 17 | DAYOFWEEK()     | 支持         | -                                                                                                                                                                                                                                                                |
| 18 | DAYOFYEAR()     | 支持         | -                                                                                                                                                                                                                                                                |
| 19 | EXTRACT()       | 支持         | -                                                                                                                                                                                                                                                                |
| 20 | FROM_DAYS()     | 支持         | -                                                                                                                                                                                                                                                                |
| 21 | FROM_UNIXTIME() | 支持         | -                                                                                                                                                                                                                                                                |
| 22 | GET_FORMAT()    | 支持         | -                                                                                                                                                                                                                                                                |
| 23 | HOUR()          | 支持         | -                                                                                                                                                                                                                                                                |
| 24 | LAST_DAY        | 支持         | -                                                                                                                                                                                                                                                                |

| 序号 | MySQL数据库                            | GaussDB数据库 | 差异                                                                                                                                                                                                    |
|----|-------------------------------------|------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 25 | LOCALTIME(),<br>LOCALTIME           | 支持         | GaussDB的按精度输出的时间值（小数点后的值）是四舍五入的；MySQL是直接截断的。GaussDB按精度输出的时间值（小数点后的值）末尾0都不显示；MySQL会显示。GaussDB只支持输入[0,6]范围内的整型值，作为返回时间的精度，其他整型值直接报错；MySQL的精度值有效值是[0,6]，但是输入的整型值内部会对256求余（例257，会返回精度1的时间值）。              |
| 26 | LOCALTIMESTAMP,<br>LOCALTIMESTAMP() | 支持         | GaussDB的按精度输出的时间值（小数点后的值）是四舍五入的；MySQL是直接截断的。GaussDB按精度输出的时间值（小数点后的值）末尾0都不显示；MySQL会显示。GaussDB只支持输入[0,6]范围内的整型值，作为返回时间的精度，超过6的整型值，会告警并按照精度6输出时间值；MySQL的精度值有效值是[0,6]，但是输入的整型值内部会对256求余（例257，会返回精度1的时间值）。 |
| 27 | MAKEDATE()                          | 支持         | -                                                                                                                                                                                                     |
| 28 | MAKETIME()                          | 支持         | 与MySQL相比，入参为NULL时，GaussDB不支持maketime函数自嵌套，MySQL支持。                                                                                                                                                    |
| 29 | MICROSECOND()                       | 支持         | -                                                                                                                                                                                                     |
| 30 | MINUTE()                            | 支持         | -                                                                                                                                                                                                     |
| 31 | MONTH()                             | 支持         | -                                                                                                                                                                                                     |
| 32 | MONTHNAME()                         | 支持         | -                                                                                                                                                                                                     |
| 33 | NOW()                               | 支持         | GaussDB的按精度输出的时间值（小数点后的值）是四舍五入的；MySQL是直接截断的。GaussDB按精度输出的时间值（小数点后的值）末尾0都不显示；MySQL会显示。GaussDB只支持输入[0,6]范围内的整型值，作为返回时间的精度，超过6的整型值，会告警并按照精度6输出时间值；MySQL的精度值有效值是[0,6]，但是输入的整型值内部会对256求余（例257，会返回精度1的时间值）。 |

| 序号 | MySQL数据库      | GaussDB数据库 | 差异                                                                       |
|----|---------------|------------|--------------------------------------------------------------------------|
| 34 | PERIOD_ADD()  | 支持         | 当入参period或结果小于0时，GaussDB参考MySQL8.0.x版本的表现，报错处理。MySQL5.7会发生整数回绕，导致计算结果异常。 |
| 35 | PERIOD_DIFF() | 支持         | 当入参或结果小于0时，GaussDB参考MySQL8.0.x版本的表现，报错处理。MySQL5.7会发生整数回绕，导致计算结果异常。       |
| 36 | QUARTER()     | 支持         | -                                                                        |
| 37 | SEC_TO_TIME() | 支持         | -                                                                        |
| 38 | SECOND()      | 支持         | -                                                                        |
| 39 | STR_TO_DATE() | 支持         | 返回值与MySQL有差异，GaussDB返回的是text，MySQL返回的是datetime、date。                     |
| 40 | SUBDATE()     | 支持         | 此函数的表现会因为interval表达式的差异与MySQL有差异，具体可见 <a href="#">INTERVAL差异说明</a> 。     |

| 序号 | MySQL数据库  | GaussDB数据库 | 差异                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
|----|-----------|------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 41 | SUBTIME() | 支持         | <ul style="list-style-type: none"> <li>• MySQL对第二入参为DATETIME样式字符串返回NULL，GaussDB可以计算</li> <li>• 入参取值范围为['0001-01-01 00:00:00', 9999-12-31 23:59:59.999999]。</li> <li>• MySQL中SUBTIME函数如果第一个参数是动态参数（例如在预准备语句中），则返回类型为 TIME。否则，函数的解析类型派生自第一个参数的解析类型。GaussDB中SUBTIME函数的返回值规则如下： <ul style="list-style-type: none"> <li>- 第一个入参为date，第二个入参为date，返回值为time。</li> <li>- 第一个入参为date，第二个入参为text，返回值为text。</li> <li>- 第一个入参为date，第二个入参为datetime，返回值为time。</li> <li>- 第一个入参为date，第二个入参为time，返回值为time。</li> <li>- 第一个入参为text，第二个入参为date，返回值为text。</li> <li>- 第一个入参为text，第二个入参为text，返回值为text。</li> <li>- 第一个入参为text，第二个入参为datetime，返回值为text。</li> <li>- 第一个入参为text，第二个入参为time，返回值为text。</li> <li>- 第一个入参为datetime，第二个入参为date，返回值为datetime。</li> <li>- 第一个入参为datetime，第二个入参为text，返回值为text。</li> <li>- 第一个入参为datetime，第二个入参为datetime，返回值为datetime。</li> <li>- 第一个入参为datetime，第二个入参为time，返回值为datetime。</li> <li>- 第一个入参为time，第二个入参为date，返回值为time。</li> </ul> </li> </ul> |

| 序号 | MySQL数据库         | GaussDB数据库 | 差异                                                                                                                                                                                  |
|----|------------------|------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|    |                  |            | <ul style="list-style-type: none"> <li>- 第一个入参为time, 第二个入参为text, 返回值为text。</li> <li>- 第一个入参为time, 第二个入参为datetime, 返回值为time。</li> <li>- 第一个入参为time, 第二个入参为time, 返回值为time。</li> </ul> |
| 42 | SYSDATE()        | 支持         | MySQL入参整型值会按照一字节最大值255整数回绕, GaussDB不回绕。                                                                                                                                             |
| 43 | YEAR()           | 支持         | -                                                                                                                                                                                   |
| 44 | TIME_FORMAT()    | 支持         | -                                                                                                                                                                                   |
| 45 | TIME_TO_SEC()    | 支持         | -                                                                                                                                                                                   |
| 46 | TIMEDIFF()       | 支持         | -                                                                                                                                                                                   |
| 47 | WEEKOFYEAR()     | 支持         | -                                                                                                                                                                                   |
| 48 | TIMESTAMPADD()   | 支持         | -                                                                                                                                                                                   |
| 49 | TIMESTAMPDIFF()  | 支持         | -                                                                                                                                                                                   |
| 50 | TO_DAYS()        | 支持         | -                                                                                                                                                                                   |
| 51 | TO_SECONDS()     | 支持         | -                                                                                                                                                                                   |
| 52 | UNIX_TIMESTAMP() | 支持         | 返回值与MySQL有差异, GaussDB返回的是numeric, MySQL返回的是int。                                                                                                                                     |
| 53 | UTC_DATE()       | 支持         | MySQL支持无括号调用, GaussDB不支持。MySQL入参整型值会按照一字节最大值255整数回绕。<br>MySQL入参只支持0-6整数, GaussDB支持可以隐式转为0-6的输入。                                                                                     |
| 54 | UTC_TIME()       | 支持         |                                                                                                                                                                                     |
| 55 | UTC_TIMESTAMP()  | 支持         |                                                                                                                                                                                     |
| 56 | WEEK()           | 支持         | -                                                                                                                                                                                   |
| 57 | WEEKDAY()        | 支持         | -                                                                                                                                                                                   |

### 8.2.2.3.3 字符串函数

表 8-33 字符串函数列表

| 序号 | MySQL数据库    | GaussDB数据库 | 差异                                                                                                                                                                                                                                                                                                                                                                                                                 |
|----|-------------|------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1  | BIN()       | 支持         | <p>函数入参1支持类型存在差异，GaussDB入参1支持类型如下：</p> <ul style="list-style-type: none"> <li>• 整数类型：tinyint、smallint、mediumint、int、bigint</li> <li>• 无符号整数类型：tinyint unsigned、smallint unsigned、int unsigned、bigint unsigned</li> <li>• 字符和文本类型：char、varchar、tinytext、text、mediumtext、longtext，仅支持纯数字整数字符串，且整数范围在bigint范围内。</li> <li>• 浮点类型：float、real、double</li> <li>• 定点类型：numeric、decimal、dec</li> <li>• 布尔类型：bool</li> </ul> |
| 2  | CONCAT()    | 支持         | <p>无论参数的数据类型如何，concat返回值的数据类型始终为text；MySQL的concat在含有二进制类型参数时，返回值为二进制类型。</p>                                                                                                                                                                                                                                                                                                                                        |
| 3  | CONCAT_WS() | 支持         | <p>无论参数的数据类型如何，concat_ws返回值的数据类型始终为text；MySQL的concat_ws在含有二进制类型参数时，返回值为二进制类型，其他情况返回值为字符串类型。</p>                                                                                                                                                                                                                                                                                                                    |

| 序号 | MySQL数据库 | GaussDB数据库 | 差异                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|----|----------|------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 4  | ELT()    | 支持         | <p>1. 函数入参1支持类型存在差异, GaussDB入参1支持类型如下:</p> <ul style="list-style-type: none"> <li>● 整数类型: tinyint、smallint、mediumint、int、bigint</li> <li>● 无符号整数类型: tinyint unsigned、smallint unsigned、int unsigned</li> <li>● 字符和文本类型: char, varchar、tinytext, text, mediumtext, longtext, 仅支持纯数字整数字符串, 且整数范围在bigint范围内。</li> <li>● 浮点类型: float、real、double</li> <li>● 定点类型: numeric、decimal、dec</li> <li>● 布尔类型: bool</li> </ul> <p>2. 函数入参2支持类型存在差异, GaussDB入参2支持类型如下:</p> <ul style="list-style-type: none"> <li>● 整数类型: tinyint、smallint、mediumint、int、bigint</li> <li>● 无符号整数类型: tinyint unsigned、smallint unsigned、int unsigned、bigint unsigned</li> <li>● 字符和文本类型: char, varchar、tinytext, text, mediumtext, longtext</li> <li>● 浮点类型: float、real、double</li> <li>● 定点类型: numeric、decimal、dec</li> <li>● 布尔类型: bool</li> <li>● 大对象类型: tinyblob, blob, mediumblob, longblob</li> <li>● 日期类型: datetime, timestamp, date, time</li> </ul> |

| 序号 | MySQL数据库      | GaussDB数据库 | 差异                                                                                                                                                                                                                                                                                                                |
|----|---------------|------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 5  | FIELD()       | 支持         | <p>函数入参为在bigint最大值~ bigint unsigned最大值范围内的数字，存在不兼容。</p> <p>函数入参为浮点型float(m, d)、double(m, d)、real(m, d)时精度更高，存在不兼容。</p>                                                                                                                                                                                            |
| 6  | FIND_IN_SET() | 支持         | <p>当数据库encoding = 'SQL_ASCII'时，不支持默认的大小写判断规则，即在用户不指定字符集规则的情况下，大写与小写区分判断。</p>                                                                                                                                                                                                                                      |
| 7  | INSERT()      | 支持         | <ul style="list-style-type: none"> <li>• Int64类型传参有范围限制，一旦超出-9223372036854775808~9223372036854775807范围会直接报错，MySQL对数值类型传参范围无限制，异常会告警按照上限或下限数值处理。</li> <li>• 字符串传参有限制，入参text类型字符串长度最大为2^30-5字节，入参bytea类型字符串长度最大为2^30-512字节。</li> <li>• s1和s2任意参数为bytea类型时，涉及到结果出现非法字符的情况可能展示结果与MySQL有差异但是字符编码与MySQL是一致的。</li> </ul> |
| 8  | LOCATE()      | 支持         | <p>入参1为bytea类型，入参2为text类型时，GaussDB与MySQL行为存在差异。</p>                                                                                                                                                                                                                                                               |



| 序号 | MySQL数据库   | GaussDB数据库 | 差异                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|----|------------|------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 9  | MAKE_SET() | 支持         | <ul style="list-style-type: none"><li>bits参数为整型时，最大范围支持到int128，低于MySQL范围。</li><li>bits参数为日期类型datetime、timestamp、date、time，由于时间类型转整型与MySQL存在差异，目前均未做支持。</li><li>bit类型或bool类型由于此类数据类型Gauss与MySQL存在差异，返回结果导致的差异为GaussDB与MySQL固有差异。bits入参为bool类型，str入参为bit类型与bool类型均不做支持。</li><li>bits入参为字符串或文本类型时，仅支持纯整型数字形式，其他形式存在差异。且纯整型数字范围限制在bigint范围。</li><li>str入参整型数值超过正负81个9，返回值与MySQL有差异。</li><li>str入参当以科学计数法表示时，GaussDB末尾0值会显示，MySQL不显示，以科学计数法打印，此为固有差异。</li></ul> |

| 序号 | MySQL数据库 | GaussDB数据库 | 差异                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
|----|----------|------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 10 | QUOTE()  | 支持         | <ul style="list-style-type: none"> <li>已知str字符串中含有“\Z”，“\r”，“\%”，“\_”，GaussDB未进行转义，与MySQL存在差异。斜线后跟部分数字也会引起差异，如“\563”。由转义字符引起的本函数与MySQL的差异，此为GaussDB与MySQL的转义字符差异。</li> <li>str字符串中的“\b”，输出结果表现形式与MySQL有差异。此为GaussDB与MySQL的固有差异</li> <li>str字符串中含有“\0”时，GaussDB由于UTF-8字符集不识别该字符，输入不成功。此为GaussDB与MySQL的固有差异</li> <li>str为bit或bool类型时，由于GaussDB与MySQL此类型目前有差异，暂不支持此类类型。</li> <li>Gauss最大支持1GB数据传输，str入参长度最大支持536870908，函数返回结果字符串最大支持1GB。</li> <li>str入参整型数值超过正负81个9，返回值与MySQL有差异。</li> <li>str入参当以科学计数法表示时，GaussDB末尾0值会显示，MySQL不显示，以科学计数法打印，此为固有差异。</li> </ul> |

| 序号 | MySQL数据库          | GaussDB数据库 | 差异                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|----|-------------------|------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 11 | SPACE()           | 支持         | GaussDB入参1最大支持1073741818，超出返回空字符串。MySQL的入参1默认最大支持4194304，超出告警。<br>函数入参1支持类型存在差异，GaussDB入参1支持类型如下： <ul style="list-style-type: none"><li>● 整数类型：tinyint、smallint、mediumint、int、bigint</li><li>● 无符号整数类型：tinyint unsigned、smallint unsigned、int unsigned</li><li>● 字符和文本类型：char，varchar、tinytext，text，mediumtext，longtext，仅支持纯数字整数字符串，且整数范围在bigint范围内。</li><li>● 浮点类型：float、real、double</li><li>● 定点类型：numeric、decimal、dec</li><li>● 布尔类型：bool</li></ul> |
| 12 | SUBSTR()          | 支持         | -                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| 13 | SUBSTRING()       | 支持         | -                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| 14 | SUBSTRING_INDEX() | 支持         | -                                                                                                                                                                                                                                                                                                                                                                                                                                                      |

| 序号 | MySQL数据库       | GaussDB数据库 | 差异                                                                                                                                                                                                                                                                                                                                                                                                           |
|----|----------------|------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 15 | STRCMP()       | 支持         | <p>1. 支持入参类型与M存在差异，GaussDB支持类型如下：</p> <ul style="list-style-type: none"> <li>• 字符类型：CHAR、VARCHAR、NVARCHAR2、TEXT</li> <li>• 二进制类型：BYTEA</li> <li>• 数值类型：TINYING [UNSIGNED]、SMALLINT [UNSIGNED]、INTEGER [UNSIGNED]、BIGINT [UNSIGNED]、FLOAT4、FLOAT8、NUMERIC</li> <li>• 日期时间类型：DATE、TIME WITHOUT TIME ZONE、DATETIME、TIMESTAMP TZ</li> </ul> <p>2. 对于数值类型中的浮点类型，由于连接参数设置不同，精度可能与M有差异，不建议使用该场景，或使用NUMERIC类型代替。</p> |
| 16 | SHA() / SHA1() | 支持         | -                                                                                                                                                                                                                                                                                                                                                                                                            |
| 17 | SHA2()         | 支持         | -                                                                                                                                                                                                                                                                                                                                                                                                            |

#### 8.2.2.3.4 强制转换函数

表 8-34 强制转换函数列表

| 序号 | MySQL数据库  | GaussDB数据库 | 差异                                      |
|----|-----------|------------|-----------------------------------------|
| 1  | CAST()    | 支持         | 数据类型转换规则和支持的转换类型均以 GaussDB支持的转换范围和规则为准。 |
| 2  | CONVERT() | 支持         | 数据类型转换规则和支持的转换类型均以 GaussDB支持的转换范围和规则为准。 |



| 序号 | MySQL数据库             | GaussDB数据库 | 差异                                                     |
|----|----------------------|------------|--------------------------------------------------------|
| 4  | JSON_ARRAY_INSERT()  | 支持         | -                                                      |
| 5  | JSON_CONTAINS()      | 支持         | -                                                      |
| 6  | JSON_CONTAINS_PATH() | 支持         | -                                                      |
| 7  | JSON_DEPTH()         | 支持         | 返回值与MySQL有差异, GaussDB返回的是int, MySQL返回的是bigint。         |
| 8  | JSON_EXTRACT()       | 支持         | -                                                      |
| 9  | JSON_INSERT()        | 支持         | -                                                      |
| 10 | JSON_KEYS()          | 支持         | -                                                      |
| 11 | JSON_LENGTH()        | 支持         | 返回值与MySQL有差异, GaussDB返回的是int, MySQL返回的是bigint。         |
| 12 | JSON_MERGE()         | 支持         | -                                                      |
| 13 | JSON_OBJECT()        | 支持         | -                                                      |
| 14 | JSON_QUOTE()         | 支持         | 返回值与MySQL有差异, GaussDB返回的是json, MySQL返回的是varchar或者text。 |
| 15 | JSON_REMOVE()        | 支持         | -                                                      |
| 16 | JSON_REPLACE()       | 支持         | -                                                      |
| 17 | JSON_SEARCH()        | 支持         | 返回值与MySQL有差异, GaussDB返回的是text, MySQL返回的是json。          |
| 18 | JSON_SET()           | 支持         | -                                                      |

| 序号 | MySQL数据库       | GaussDB数据库 | 差异                               |
|----|----------------|------------|----------------------------------|
| 19 | JSON_TYPE()    | 支持         | 数值类型的json值统一识别为number，与MySQL有差异。 |
| 20 | JSON_UNQUOTE() | 支持         | -                                |
| 21 | JSON_VALID()   | 支持         | -                                |

### 8.2.2.3.8 聚合函数

表 8-38 聚合函数列表

| 序号 | MySQL数据库       | GaussDB数据库 | 差异                                                                                                                                                                                                                                                                                                                                                            |
|----|----------------|------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1  | GROUP_CONCAT() | 支持         | <p>当group_concat参数中同时有DISTINCT和ORDER BY语法时，所有ORDER BY后的表达式必须也在DISTINCT的表达式之中。</p> <p>group_concat(... order by 数字)不代表按照第几个参数的顺序，数字只是一个常量表达式，相当于不排序。</p> <p>无论参数的数据类型如何，group_concat返回值的数据类型始终为text；MySQL的group_concat在含有二进制类型参数时，返回值为二进制类型，其他情况返回值为字符串类型，并且返回值长度大于512时，其数据类型为字符串大对象或二进制大对象。</p> <p>GUC参数group_concat_max_len有效范围是0-1073741823，最大值比MySQL小。</p> |

| 序号 | MySQL数据库  | GaussDB数据库 | 差异                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|----|-----------|------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 2  | DEFAULT() | 支持         | <ul style="list-style-type: none"> <li>• 字段默认值为数组形式，GaussDB返回数组形式，MySQL不支持数组类型。</li> <li>• GaussDB字段是隐藏列（比如xmin、cmin），default函数返回空值。</li> <li>• GaussDB支持分区表、临时表、多表连接查询默认值。</li> <li>• GaussDB支持查询列名包含字符串值节点（表示名称）和A_Star节点（表示出现“*”），如default(tt.t4.id)和default(tt.t4.*)。不合法的查询列名和A_Star节点，GaussDB和MySQL报错信息有差异。</li> <li>• GaussDB创建字段默认值，没有检验字段类型的范围，使用default函数可能报错。</li> <li>• 字段的默认值是函数表达式时，GaussDB的default函数返回建表时字段的default表达式的计算值。MySQL的default函数返回NULL。</li> </ul> |

### 8.2.2.3.9 数字操作函数

表 8-39 数字操作函数列表

| 序号 | MySQL数据库 | GaussDB数据库 | 差异                                                                                                                                                                                                                                                                                                                                                                                                |
|----|----------|------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1  | log2()   | 支持         | <ul style="list-style-type: none"> <li>• 小数位显示与MySQL存在差异，受GaussDB浮点数据类型限制，可通过参数extra_float_digits控制小数位个数显示；</li> <li>• 由于输入精度内部处理差异，GaussDB与MySQL会在结果计算差异；</li> <li>• 支持数据类型有：bigint、int16、int、smallint、tinyint整数类型；bigint unsigned、integer unsigned、smallint unsigned、tinyint unsigned无符号整数类型；numeric、real浮点数类型；character、character varying、clob、text字符串类型，仅支持纯数字整数字符串；set类型；NULL空类型。</li> </ul> |



| 序号 | MySQL数据库     | GaussDB数据库 | 差异                                                                                                                                                                                                                                                                                                                                                                                                 |
|----|--------------|------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 2  | log10()<br>) | 支持         | <ul style="list-style-type: none"> <li>• 小数位显示与MySQL存在差异，受GaussDB浮点数据类型限制，可通过参数extra_float_digits控制小数位个数显示；</li> <li>• 由于输入精度内部处理差异，GaussDB与MySQL会存在结果计算差异；</li> <li>• 支持数据类型有：bigint、int16、int、smallint、tinyint整数类型；bigint unsigned、integer unsigned、smallint unsigned、tinyint unsigned无符号整数类型；numeric、real浮点数类型；character、character varying、clob、text字符串类型，仅支持纯数字整数字符串；set类型；NULL空类型。</li> </ul> |

### 8.2.2.3.10 其他函数

表 8-40 其他函数列表

| 序号 | MySQL数据库     | GaussDB数据库 | 差异 |
|----|--------------|------------|----|
| 1  | UUID()       | 支持         | -  |
| 2  | UUID_SHORT() | 支持         | -  |

### 8.2.2.4 操作符

GaussDB数据库兼容绝大多数MySQL的操作符，但存在部分差异。如未列出，操作符行为默认为GaussDB原生行为。

表 8-41 操作符

| 序号 | MySQL数据库   | GaussDB数据库 | 差异 |
|----|------------|------------|----|
| 1  | 安全等于 (<=>) | 支持         | -  |

| 序号 | MySQL数据库     | GaussDB数据库 | 差异                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|----|--------------|------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 2  | [NOT] REGEXP | 支持         | <ul style="list-style-type: none"> <li>当开启 b_format_dev_version='s2' 时，模式字符串pattern 中有 '\a', '\\d', '\\e', '\\n', '\\Z', '\\u' 等转义字符时，匹配源字符串 '\a', '\\d', '\\e', '\\n', '\\Z', '\\u' 时，GaussDB行为与MySQL5.7不一致。MySQL5.7存在bug，MySQL后续版本已经修复与GaussDB一致。</li> <li>当开启 b_format_dev_version='s2' 时，GaussDB '\\b' 可以与 '\\b' 匹配，MySQL匹配失败</li> <li>模式字符串pat非法入参，只存在右单括号 ')' 时，GaussDB数据库报错。MySQL存在bug，后续版本已经修复此问题。</li> <li>在de abc匹配序列de或abc的匹配规则，当 左右存在空值时，MySQL存在bug，会报错，后续版本已经修复此问题。</li> <li>空白字符[\t]正则匹配字符类[:blank:]，GaussDB可匹配，MySQL\t不能匹配[:blank:]，MySQL存在bug，后续版本已经修复此问题。</li> <li>GaussDB支持非贪婪模式匹配，即尽可能少的匹配字符，在部分特殊字符后加 '?' 问号字符，例如：“??, *, +?, {n}?, {n,}?, {n,m}?”。MySQL 5.7版本不支持非贪婪模式匹配，并报错：Got error 'repetition-operator operand invalid' from regexp。MySQL 8.0版本已经支持。</li> <li>在binary字符集下，text类型、blob类型均会转换成bytea类型，由于REGEXP操作符不支持bytea类型，因此无法匹配。</li> </ul> |
| 3  | [NOT] RLIKE  | 支持         | 同[NOT] REGEXP。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |

### 8.2.2.5 字符集

GaussDB数据库支持指定数据库、模式、表或列的字符集，支持的范围如下。

表 8-42 字符集列表

| 序号 | MySQL数据库 | GaussDB数据库 |
|----|----------|------------|
| 1  | utf8mb4  | 支持         |
| 2  | gbk      | 支持         |
| 3  | gb18030  | 支持         |
| 4  | utf8     | 支持         |
| 5  | binary   | 支持         |

### 8.2.2.6 排序规则

GaussDB数据库支持指定库、模式、表或列的排序规则，支持的范围如下。

#### 📖 说明

排序规则差异说明：

- 当前仅有字符串类型、部分二进制类型支持指定排序规则，其他类型不支持指定排序规则，可以通过查询pg\_type系统表中类型的typcollation属性不为0来判断该类型支持字符序。MySQL中所有类型可以指定字符序，但除字符串、二进制类型其他排序规则无实际意义。
- 当前排序规则（除binary外）仅支持在其对应字符集与库级字符集一致时可以指定，GaussDB数据库中，字符集必须与数据库的字符集一致，且不支持表内多种字符集混合使用。
- utf8mb4字符集下默认字符序为utf8mb4\_general\_ci，与MySQL5.7保持一致。
- GaussDB中utf8和utf8mb4为同一个字符集。

表 8-43 排序规则列表

| 序号 | MySQL数据库           | GaussDB数据库 |
|----|--------------------|------------|
| 1  | utf8mb4_general_ci | 支持         |
| 2  | utf8mb4_unicode_ci | 支持         |
| 3  | utf8mb4_bin        | 支持         |
| 4  | gbk_chinese_ci     | 支持         |
| 5  | gbk_bin            | 支持         |
| 6  | gb18030_chinese_ci | 支持         |
| 7  | gb18030_bin        | 支持         |
| 8  | binary             | 支持         |
| 9  | utf8mb4_0900_ai_ci | 支持         |

| 序号 | MySQL数据库        | GaussDB数据库 |
|----|-----------------|------------|
| 10 | utf8_general_ci | 支持         |
| 11 | utf8_bin        | 支持         |

### 8.2.2.7 表达式

GaussDB数据库兼容绝大多数MySQL的表达式，但存在部分差异。如未列出，表达式行为默认为GaussDB原生行为。

表 8-44 表达式

| 序号 | MySQL数据库         | GaussDB数据库 |
|----|------------------|------------|
| 1  | 用户自定义变量@var_name | 部分支持       |
| 2  | 全局变量@@var_name   | 部分支持       |

### 8.2.2.8 SQL

GaussDB数据库兼容绝大多数MySQL语法，但存在部分差异。本章节介绍GaussDB数据库当前支持的MySQL语法。

#### 8.2.2.8.1 DDL

| 序号 | MySQL数据库功能概述                | 详细语法说明                   | GaussDB数据库实现差异                                                                                                                                                                                                                                          |
|----|-----------------------------|--------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1  | 建表和修改表时支持创建主键、UNIQUE索引、外键约束 | ALTER TABLE、CREATE TABLE | <ul style="list-style-type: none"> <li>GaussDB当前不支持UNIQUE INDEX KEY index_name语法，使用UNIQUE INDEX KEY index_name语法时会报错。</li> <li>当约束被建立为全局二级索引，SQL语句中指定using btree时，底层会建立为ubtree。</li> <li>当约束关联的表为ustore，且SQL语句中指定为using btree时，底层会建立为ubtree。</li> </ul> |

| 序号 | MySQL数据库功能概述 | 详细语法说明                   | GaussDB数据库实现差异                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|----|--------------|--------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 2  | 支持自增列        | ALTER TABLE、CREATE TABLE | <ul style="list-style-type: none"> <li>● 自动增长列建议为索引（非全局二级索引）的第一个字段，否则建表时产生警告，含有自动增长列的表进行某些操作时会产生错误，例如：ALTER TABLE EXCHANGE PARTITION。MySQL自动增长列必须为索引第一个字段。</li> <li>● AUTO_INCREMENT = value语法，value必须为小于<math>2^{127}</math>的正数。MySQL不校验value。</li> <li>● 当自增值已经达到字段数据类型的最大值时，继续自增将产生错误。MySQL有些场景产生错误或警告，有些场景仍自增为最大值。</li> <li>● 不支持 innodb_autoinc_lock_mode系统变量，GaussDB的GUC参数 auto_increment_cache =0时，批量插入自动增长列的行为与MySQL系统变量 innodb_autoinc_lock_mode=1相似。</li> <li>● 自动增长列在导入数据或者进行Batch Insert 执行计划的插入操作时，对于混合0、NULL和确定值的场景，如果产生错误，后续插入自增值不一定与MySQL完全一致。             <ul style="list-style-type: none"> <li>- 提供 auto_increment_cache参数，可以控制预留自增值的数量。</li> </ul> </li> <li>● 并行导入或插入自动增长列触发自增时，每个并行线程预留的缓存值</li> </ul> |

| 序号 | MySQL数据库功能概述 | 详细语法说明 | GaussDB数据库实现差异                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|----|--------------|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|    |              |        | <p>也只在主线程中使用，未完全使用完毕的话，也会出现表中自动增长列的值不连续的情况。并行插入产生的自增值结果无法保证与MySQL完全一致。</p> <ul style="list-style-type: none"> <li>● 本地临时表中的自动增长列批量插入时不会预留自增值，正常场景不会产生不连续的自增值。MySQL临时表与普通表中的自动增长列自增结果一致。</li> <li>● SERIAL数据类型为原有的自增列，与 AUTO_INCREMENT自增列有差异。MySQL的 SERIAL数据类型就是 AUTO_INCREMENT自增列。</li> <li>● 不允许 auto_increment_offset 的值大于 auto_increment_increment 的值，会产生错误。MySQL允许，并说明 auto_increment_offset 会被忽略。</li> <li>● 在表有主键或索引的情况下，ALTER TABLE命令重写表数据的顺序与MySQL不一定相同，GaussDB按照表数据存储顺序重写，MySQL会按主键或索引顺序重写，导致自增值的顺序可能不同。</li> <li>● ALTER TABLE命令添加或修改自增列时，第一次预留自增值的数量是表统计信息中的行数，统计信息的行数不一定与MySQL一致。</li> <li>● last_insert_id函数返回值为128位的整型。</li> </ul> |

| 序号 | MySQL数据库功能概述 | 详细语法说明                                              | GaussDB数据库实现差异                                                                                                                                                                                                                                                                                                                                                                             |
|----|--------------|-----------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|    |              |                                                     | <ul style="list-style-type: none"> <li>在触发器或用户自定义函数中自增时，刷新 last_insert_id 返回值。MySQL 不刷新。</li> <li>对 GUC 参数 auto_increment_offset 和 auto_increment_increment 设置超出范围的值会产生错误。MySQL 会自动改为边界值。</li> </ul>                                                                                                                                                                                       |
| 3  | 支持前缀索引       | CREATE INDEX、ALTER TABLE、CREATE TABLE               | <ul style="list-style-type: none"> <li>前缀长度不得超过 2676，键值的实际长度受内部页面限制，若字段中含有多字节字符或者一个索引上有多个键，索引行长度可能会超限报错。</li> <li>CREATE INDEX 语法中，不支持以下关键字作为前缀键的字段名称：COALESCE、EXTRACT、GREATEST、LEAST、LNNVL、NULLIF、NVL、NVL2、OVERLAY、POSITION、REGEXP_LIKE、SUBSTRING、TIMESTAMPDIFF、TREAT、TRIM、XMLCONCAT、XMLELEMENT、XMLEXISTS、XMLFOREST、XMLPARSE、XMLPI、XMLROOT、XMLSERIALIZE。</li> <li>主键索引中不支持前缀键。</li> </ul> |
| 4  | 支持指定字符集与排序规则 | ALTER SCHEMA、ALTER TABLE、CREATE SCHEMA、CREATE TABLE | -                                                                                                                                                                                                                                                                                                                                                                                          |

| 序号 | MySQL数据库功能概述             | 详细语法说明                                                      | GaussDB数据库实现差异 |
|----|--------------------------|-------------------------------------------------------------|----------------|
| 5  | 修改表时支持在表第一列前面或者在指定列后面添加列 | ALTER TABLE                                                 | -              |
| 6  | 修改列名称/定义语法兼容             | ALTER TABLE                                                 | -              |
| 7  | 定时任务EVENT语法兼容            | ALTER EVENT、<br>CREATE EVENT、<br>DROP EVENT、<br>SHOW EVENTS | -              |
| 8  | 创建分区表语法兼容                | CREATE TABLE<br>PARTITION、CREATE<br>TABLE SUBPARTITION      | -              |
| 9  | 建表和修改表时支持指定表级和列级comment  | CREATE TABLE、<br>ALTER TABLE                                | -              |
| 10 | 创建索引时支持指定索引级comment      | CREATE INDEX                                                | -              |



| 序号 | MySQL数据库功能概述   | 详细语法说明                | GaussDB数据库实现差异                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|----|----------------|-----------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 11 | 交换普通表和分区表分区的数据 | ALTER TABLE PARTITION | <p>ALTER TABLE EXCHANGE PARTITION的差异点:</p> <ul style="list-style-type: none"> <li>• 对于自增列，MySQL执行alter exchange partition后，自增列会被重置；GaussDB 则不会被重置，自增列则按照旧的自增值递增。</li> <li>• MySQL表或分区使用 tablespace时，则无法进行分区和普通表数据的交换；GaussDB表或分区使用不同的 tablespace时，仍可进行分区和普通表数据的交换。</li> <li>• 对于列默认值，MySQL不会校验默认值，因此默认值不同时也可进行分区和普通表数据的交换；GaussDB会校验默认值，如果默认值不同，则无法进行分区和普通表数据的交换。</li> <li>• MySQL在分区表或普通表上进行DROP列操作后，表结构仍然一致，则可进行分区和普通表数据的交换；GaussDB需要保证普通表和分区表的被删除列严格对齐才能进行分区和普通表数据的交换。</li> <li>• MySQL和GaussDB的哈希算法不同，所以两者在相同的hash分区存储的数据可能不一致，导致最后交换的数据也可能不一致。</li> <li>• MySQL的分区表不支持外键，普通表包含外键或其他表引用普通表的外键，则无法进行分区和普通表数据的交换；GaussDB的分区表支持外键，在两个表的外键约束一致时，则可进行</li> </ul> |

| 序号 | MySQL数据库功能概述 | 详细语法说明                                  | GaussDB数据库实现差异                                                      |
|----|--------------|-----------------------------------------|---------------------------------------------------------------------|
|    |              |                                         | 分区和普通表数据的交换，GaussDB的分区表不带外键，普通表有其他表引用，如果分区表和普通表表一致，则可进行分区和普通表数据的交换。 |
| 12 | 支持删除表的主键外键约束 | ALTER TABLE DROP [PRIMARY   FOREIGN]KEY | -                                                                   |

| 序号 | MySQL数据库功能概述                | 详细语法说明                | GaussDB数据库实现差异                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|----|-----------------------------|-----------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 13 | 支持CREATE TABLE ... LIKE语法兼容 | CREATE TABLE ... LIKE | <ul style="list-style-type: none"> <li>● 在MySQL 8.0.16 之前的版本中，CHECK约束会被语法解析但功能会被忽略，表现为不复制CHECK约束，GaussDB支持复制CHECK约束。</li> <li>● 对于主键约束名称，在建表时，MySQL所有主键约束名称固定为PRIMARY KEY，GaussDB不支持复制。</li> <li>● 对于唯一键约束名称，在建表时，MySQL支持复制，GaussDB不支持复制。</li> <li>● 对于CHECK约束名称，在建表时，MySQL 8.0.16 之前的版本无CHECK约束信息，GaussDB支持复制。</li> <li>● 对于索引名称，在建表时，MySQL支持复制，GaussDB不支持复制。</li> <li>● 在跨sql_mode模式建表时，MySQL受宽松模式和严格模式控制，GaussDB可能存在严格模式失效的情况。<br/>例如：源表存在默认值“0000-00-00”，在“no_zero_date”严格模式下，GaussDB建表成功，且包含默认值“0000-00-00”，严格模式失效；而MySQL建表失败，受严格模式控制。</li> <li>● 针对跨数据库创建表，MySQL支持，GaussDB不支持。</li> </ul> |

| 序号 | MySQL数据库功能概述 | 详细语法说明                                                                                                                                                    | GaussDB数据库实现差异                                                                                                                                                                                                                                                                                                     |
|----|--------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 14 | 支持更改表名兼容语法   | <pre>ALTER TABLE tbl_name RENAME [TO   AS   =] new_tbl_name;  RENAME {TABLE   TABLES} tbl_name TO new_tbl_name [, tbl_name2 TO new_tbl_name2, ...];</pre> | <ul style="list-style-type: none"> <li>• GaussDB的alter rename语法仅支持修改表名称功能操作，不能耦合其它功能操作。</li> <li>• GaussDB仅旧表名字段支持如 schema.table_name用法；且新表名与旧表名将属于同一schema下。</li> <li>• GaussDB不支持新旧表跨schema重命名操作；但如有权限，则可在当前schema下修改其它schema下表名称。</li> <li>• GaussDB的rename多组表的语法支持全为本地临时表的重命名，不支持本地临时表和非本地临时表组合的场景。</li> </ul> |

| 序号 | MySQL数据库功能概述 | 详细语法说明                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 | GaussDB数据库实现差异                                                                                                                                                                                                                                                                                                                         |
|----|--------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 15 | 支持增加子分区语法兼容  | <p>ALTER TABLE [ IF EXISTS ] { table_name [*]   ONLY table_name   ONLY ( table_name ) }</p> <p>action [, ... ];</p> <p>action:</p> <p>move_clause  <br/>exchange_clause  <br/>row_clause  <br/>merge_clause  <br/>modify_clause  <br/>split_clause  <br/>add_clause  <br/>drop_clause  <br/>ilm_clause</p> <p>add_clause:</p> <p>ADD<br/>{partition_less_than_item  <br/>partition_start_end_item   partition_list_item}<br/> <br/>PARTITION({partition_less_than_item  <br/>partition_start_end_item  <br/>partition_list_item})}</p> | <ul style="list-style-type: none"> <li>不支持如下语法添加多分区：<br/>ALTER TABLE table_name<br/>ADD PARTITION<br/>(partition_definition1,<br/>partition_definition1,...);</li> <li>仅支持原有添加多分区语法：<br/>ALTER TABLE table_name<br/>ADD PARTITION<br/>(partition_definition1), ADD<br/>PARTITION<br/>(partition_definition2[y1] ),<br/>...;</li> </ul> |

### 8.2.2.8.2 DML

| 序号 | MySQL数据库功能概述            | 详细语法说明 | GaussDB数据库实现差异 |
|----|-------------------------|--------|----------------|
| 1  | DELETE支持从多个表中删除数据       | DELETE | -              |
| 2  | DELETE支持ORDER BY和LIMIT  | DELETE | -              |
| 3  | DELETE支持从指定分区（或子分区）删除数据 | DELETE | -              |

| 序号 | MySQL数据库功能概述           | 详细语法说明 | GaussDB数据库实现差异                                                                                                                                          |
|----|------------------------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------|
| 4  | UPDATE支持从多个表中更新数据      | UPDATE | -                                                                                                                                                       |
| 5  | UPDATE支持ORDER BY和LIMIT | UPDATE | -                                                                                                                                                       |
| 6  | SELECT INTO语法兼容        | SELECT | <ul style="list-style-type: none"><li>• GaussDB可以使用SELECT INTO根据查询结果创建一个新表，MySQL不支持。</li><li>• GaussDB的SELECT INTO语法不支持将多个查询进行集合运算后的结果作为查询结果。</li></ul> |

| 序号 | MySQL数据库功能概述     | 详细语法说明  | GaussDB数据库实现差异                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|----|------------------|---------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7  | REPLACE INTO语法兼容 | REPLACE | <ul style="list-style-type: none"> <li>● 时间类型初始值的差异。例如： <ul style="list-style-type: none"> <li>- MySQL不受严格模式和宽松模式的影响，可向表中插入时间0值，即： <pre>mysql&gt; CREATE TABLE test(f1 TIMESTAMP NOT NULL, f2 DATETIME NOT NULL, f3 DATE NOT NULL); Query OK, 1 row affected (0.00 sec)  mysql&gt; REPLACE INTO test VALUES(f1, f2, f3); Query OK, 1 row affected (0.00 sec)  mysql&gt; SELECT * FROM test; +-----+-----+   f1            f2              f3                          +-----+-----+   0000-00-00 00:00:00   0000-00-00 00:00:00   0000-00-00   +-----+-----+ 1 row in set (0.00 sec)</pre> </li> <li>- GaussDB在宽松模式下才可以成功插入时间0值，即 <pre>gaussdb=# SET b_format_version = '5.7'; SET gaussdb=# SET b_format_dev_version = 's1'; SET gaussdb=# SET sql_mode = ""; SET gaussdb=# CREATE TABLE test(f1 TIMESTAMP NOT NULL, f2 DATETIME NOT NULL, f3 DATE NOT NULL); CREATE TABLE gaussdb=# REPLACE INTO test VALUES(f1, f2, f3); REPLACE 0 1 gaussdb=# SELECT * FROM test; f1            f2            f3 ----- 0000-00-00 00:00:00   0000-00-00 00:00:00   0000-00-00 (1 row)</pre> 在严格模式下，则报错date/time field value out of range: "0000-00-00 00:00:00"。 </li> </ul> </li> <li>● 位串类型初始值的差异。例如：</li> </ul> |

| 序号 | MySQL数据库功能概述    | 详细语法说明 | GaussDB数据库实现差异                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
|----|-----------------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|    |                 |        | <ul style="list-style-type: none"> <li>- MySQLBIT类型的初始值为空串", 即:<br/> <pre>mysql&gt; CREATE TABLE test(f1 BIT(3) NOT NULL); Query OK, 0 rows affected (0.01 sec)  mysql&gt; REPLACE INTO test VALUES(f1); Query OK, 1 row affected (0.00 sec)  mysql&gt; SELECT f1, f1 IS NULL FROM test; +----+-----+   f1   f1 is null   +----+-----+            0                0     +----+-----+ 2 rows in set (0.00 sec)</pre> </li> <li>- GaussDB位串类型BIT的初始值为NULL, 则报错。<br/> <pre>gaussdb=# CREATE TABLE test(f1 BIT(3) NOT NULL); CREATE TABLE gaussdb=# REPLACE INTO test VALUES(f1); ERROR: null value in column "f1" violates not-null constraint DETAIL: Failing row contains (null).</pre> </li> </ul> |
| 8  | SELECT支持指定多分区查询 | SELECT | -                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| 9  | UPDATE支持指定多分区更新 | UPDATE | -                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |



| 序号 | MySQL数据库功能概述    | 详细语法说明    | GaussDB数据库实现差异                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|----|-----------------|-----------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 10 | LOAD DATA导入数据功能 | LOAD DATA | <ul style="list-style-type: none"> <li>● LOAD DATA语法执行结果与MySQL严格模式一致，宽松模式暂未适配。</li> <li>● IGNORE与LOCAL参数功能仅为当导入数据与表中数据存在冲突时，忽略当前冲突行数据功能和当文件中字段数小于指定表中列数时自动为其余列填充默认值功能，其余功能暂未适配。</li> <li>● 指定LOCAL关键字，且文件路径为相对路径时，文件从二进制目录下搜索；不指定LOCAL关键字，且文件路径为相对路径时，文件从数据目录下搜索。</li> <li>● 语法中指定分隔符，转义字符，分行符等符号时，若指定为单引号，将导致词法解析错误。</li> <li>● [(col_name_or_user_var [, col_name_or_user_var] ...)]指定列参数不支持重复指定列。</li> <li>● [FIELDS TERMINATED BY 'string']指定换行符不能与[LINES TERMINATED BY 'string']分隔符相同。</li> <li>● 执行LOAD DATA语法写入表中的数据若无法转换为表中数据类型格式时报错。</li> <li>● LOAD DATA SET表达式中不支持指定列名计算。</li> <li>● 若set表达式返回值类型与对应列类型之间不存在隐式转换函数则报错。</li> <li>● LOAD DATA只能用于表，不能用于视图。</li> <li>● windows下的文件与linux环境下文件默认换行符存在差异，LOAD DATA无法识别此场景会报错，建议用户导入时检查导入文件行尾换行符。</li> </ul> |

| 序号 | MySQL数据库功能概述    | 详细语法说明        | GaussDB数据库实现差异                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|----|-----------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 11 | INSERT IGNORE兼容 | INSERT IGNORE | <ul style="list-style-type: none"> <li>● GaussDB会返回降级后的错误信息，MySQL则会将降级后的错误信息记录到错误堆栈中，然后调用show warnings;命令查看。</li> <li>● 时间类型的差异。例如：             <ul style="list-style-type: none"> <li>- GaussDB中date、datetime、timestamp默认零值。<br/> <pre>gaussdb=# CREATE TABLE test(f1 DATE NOT NULL, f2 DATETIME NOT NULL, f3 TIMESTAMP NOT NULL); CREATE TABLE gaussdb=# INSERT IGNORE INTO test VALUES(NULL, NULL, NULL); WARNING: null value in column "f1" violates not-null constraint DETAIL: Failing row contains (null, null, null, null). WARNING: null value in column "f2" violates not-null constraint DETAIL: Failing row contains (null, null, null, null). WARNING: null value in column "f3" violates not-null constraint DETAIL: Failing row contains (null, null, null, null). INSERT 0 1 gaussdb=# SELECT * FROM test;    f1       f2        f3 -----+----- 1970-01-01   1970-01-01 00:00:00   1970-01-01 00:00:00 (1 row)</pre> </li> <li>- MySQL中date、datetime、timestamp默认零值。<br/> <pre>mysql&gt; CREATE TABLE test(f1 DATE NOT NULL, f2 DATETIME NOT NULL, f3 TIMESTAMP NOT NULL); Query OK, 0 rows affected (0.00 sec)  mysql&gt; INSERT IGNORE INTO test VALUES(NULL, NULL, NULL); Query OK, 1 row affected, 3 warnings (0.00 sec)  mysql&gt; show warnings; +-----+-----+   Level   Code   Message   +-----+-----+ </pre> </li> </ul> </li> </ul> |

| 序号 | MySQL数据库功能概述 | 详细语法说明 | GaussDB数据库实现差异                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|----|--------------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|    |              |        | <pre>   Warning   1048   Column 'f1' cannot be null     Warning   1048   Column 'f2' cannot be null     Warning   1048   Column 'f3' cannot be null   +-----+----- +-----+ 3 rows in set (0.00 sec)  mysql&gt; SELECT * FROM test; +-----+-----+ +-----+   f1        f2        f3        +-----+-----+ +-----+   0000-00-00   0000-00-00 00:00:00     0000-00-00 00:00:00   +-----+-----+ +-----+ 1 row in set (0.00 sec) </pre> <ul style="list-style-type: none"> <li>• 由于GaussDB不支持MySQL的bit类型，因此忽略bit类型NOT NULL约束和插入的bit类型长度与定义不同的场景下不支持INSERT IGNORE错误降级。 <ul style="list-style-type: none"> <li>- GaussDB中bit类型 <pre> gaussdb=# CREATE TABLE test(f1 BIT(10) NOT NULL); CREATE TABLE gaussdb=# INSERT IGNORE INTO test VALUES(NULL); ERROR: Un-support feature DETAIL: ignore null for insert statement is not supported in column f1. gaussdb=# INSERT IGNORE INTO test VALUES('1010'); ERROR: bit string length 4 does not match type bit(10) CONTEXT: referenced column: f1 </pre> </li> <li>- MySQL中bit类型 <pre> mysql&gt; CREATE TABLE test(f1 BIT(10) NOT NULL); Query OK, 0 rows affected (0.00 sec)  mysql&gt; INSERT IGNORE INTO test VALUES(NULL); Query OK, 1 row affected, 1 warning (0.00 sec)  mysql&gt; INSERT IGNORE INTO test VALUES('1010'); Query OK, 1 row affected, 1 warning (0.01 sec) </pre> </li> </ul> </li> <li>• MySQL数据库时间类型指定精度时，插入时间零值会显示精</li> </ul> |

| 序号 | MySQL数据库功能概述 | 详细语法说明 | GaussDB数据库实现差异                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
|----|--------------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|    |              |        | <p>度，GaussDB则不显示，例如：</p> <ul style="list-style-type: none"> <li>- GaussDB指定时间精度 <pre> gaussdb=# CREATE TABLE test(f1 TIME(3) NOT NULL, f2 DATETIME(3) NOT NULL, f3 TIMESTAMP(3) NOT NULL); CREATE TABLE gaussdb=# INSERT IGNORE INTO test VALUES(NULL,NULL,NULL); WARNING: null value in column "f1" violates not-null constraint DETAIL: Failing row contains (null, null, null). WARNING: null value in column "f2" violates not-null constraint DETAIL: Failing row contains (null, null, null). WARNING: null value in column "f3" violates not-null constraint DETAIL: Failing row contains (null, null, null). INSERT 0 1 gaussdb=# SELECT * FROM test;  f1        f2             f3 -----+-----+----- 00:00:00   1970-01-01 00:00:00   1970-01-01 00:00:00 (1 row) </pre> </li> <li>- MySQL指定时间精度 <pre> mysql&gt; CREATE TABLE test(f1 TIME(3) NOT NULL, f2 DATETIME(3) NOT NULL, f3 TIMESTAMP(3) NOT NULL); Query OK, 0 rows affected (0.00 sec)  mysql&gt; INSERT IGNORE INTO test VALUES(NULL,NULL,NULL); Query OK, 1 row affected, 3 warnings (0.00 sec)  mysql&gt; SELECT * FROM test; +-----+-----+-----+   f1        f2        f3        +-----+-----+-----+   00:00:00.000   0000-00-00 00:00:00.000   0000-00-00 00:00:00.000   +-----+-----+-----+ 1 row in set (0.00 sec) </pre> </li> </ul> <p>● 由于MySQL数据库和GaussDB执行过程的差异，因此，产生</p> |

| 序号 | MySQL数据库功能概述 | 详细语法说明 | GaussDB数据库实现差异                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|----|--------------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|    |              |        | <p>的warnings条数可能不同，例如：</p> <ul style="list-style-type: none"> <li>- GaussDB产生的warnings条数           <pre>gaussdb=# CREATE TABLE test(f1 INT, f2 INT not null); CREATE TABLE gaussdb=# INSERT INTO test VALUES(1,0),(3,0),(5,0); INSERT 0 3 gaussdb=# INSERT IGNORE INTO test SELECT f1+1, f1/f2 FROM test; WARNING: division by zero CONTEXT: referenced column: f2 WARNING: null value in column "f2" violates not-null constraint DETAIL: Failing row contains (2, null). WARNING: division by zero CONTEXT: referenced column: f2 WARNING: null value in column "f2" violates not-null constraint DETAIL: Failing row contains (4, null). WARNING: division by zero CONTEXT: referenced column: f2 WARNING: null value in column "f2" violates not-null constraint DETAIL: Failing row contains (6, null). INSERT 0 3</pre> </li> <li>- MySQL产生的warnings条数           <pre>mysql&gt; CREATE TABLE test(f1 INT, f2 INT not null); Query OK, 0 rows affected (0.01 sec)  mysql&gt; INSERT INTO test VALUES(1,0),(3,0),(5,0); Query OK, 3 rows affected (0.00 sec) Records: 3 Duplicates: 0 Warnings: 0  mysql&gt; INSERT IGNORE INTO test SELECT f1+1, f1/f2 FROM test; Query OK, 3 rows affected, 4 warnings (0.00 sec) Records: 3 Duplicates: 0 Warnings: 4</pre> </li> <li>• MySQL数据库和GaussDB INSERT IGNORE在触发器中的差异，例如：           <ul style="list-style-type: none"> <li>- GaussDB触发器中使用 INSERT IGNORE               <pre>gaussdb=# CREATE TABLE test1(f1 INT NOT NULL); CREATE TABLE gaussdb=# CREATE TABLE test2(f1</pre> </li> </ul> </li> </ul> |

| 序号 | MySQL数据库功能概述 | 详细语法说明 | GaussDB数据库实现差异                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
|----|--------------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|    |              |        | <pre> INT); CREATE TABLE gaussdb=# CREATE OR REPLACE FUNCTION trig_test() RETURNS TRIGGER AS \$\$ gaussdb\$\$ BEGIN gaussdb\$\$ INSERT IGNORE INTO test1 VALUES(NULL); gaussdb\$\$ RETURN NEW; gaussdb\$\$ END; gaussdb\$\$ \$\$ LANGUAGE plpgsql; CREATE FUNCTION gaussdb=# CREATE TRIGGER trig2 BEFORE INSERT ON test2 FOR EACH ROW EXECUTE PROCEDURE trig_test(); CREATE TRIGGER gaussdb=# INSERT INTO test2 VALUES(NULL); WARNING: null value in column "f1" violates not-null constraint DETAIL: Failing row contains (null). CONTEXT: SQL statement "INSERT IGNORE INTO test1 VALUES(NULL)" PL/pgSQL function trig_test() line 3 at SQL statement INSERT 0 1 gaussdb=# SELECT * FROM test1; f1 ---- 0 (1 rows)  gaussdb=# SELECT * FROM test2; f1 ---- (1 rows) </pre> <p>- MySQL触发器中使用INSERT IGNORE</p> <pre> mysql&gt; CREATE TABLE test1(f1 INT NOT NULL); Query OK, 0 rows affected (0.01 sec)  mysql&gt; CREATE TABLE test2(f1 INT); Query OK, 0 rows affected (0.00 sec)  mysql&gt; DELIMITER    mysql&gt; CREATE TRIGGER trig2 BEFORE INSERT ON test2 FOR EACH ROW -&gt; BEGIN -&gt; INSERT IGNORE into test1 values(NULL); -&gt; END   Query OK, 0 rows affected (0.01 sec)  mysql&gt; DELIMITER ; </pre> |

| 序号 | MySQL数据库功能概述 | 详细语法说明 | GaussDB数据库实现差异                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|----|--------------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|    |              |        | <pre>mysql&gt; INSERT INTO test2 VALUES(NULL); ERROR 1048 (23000): Column 'f1' cannot be null mysql&gt; INSERT IGNORE INTO test2 VALUES(NULL); Query OK, 1 row affected (0.00 sec)</pre> <pre>mysql&gt; SELECT * FROM test1; +----+   f1   +----+   0   +----+ 1 row in set (0.00 sec)</pre> <pre>mysql&gt; SELECT * FROM test2; +-----+   f1   +-----+   NULL   +-----+ 1 row in set (0.00 sec)</pre> <ul style="list-style-type: none"> <li>● GaussDB的bool、serial的实现机制与MySQL不同，因此其默认零值与MySQL不同，例如：             <ul style="list-style-type: none"> <li>- GaussDB的行为                     <pre>gaussdb=# CREATE TABLE test(f1 SERIAL, f2 BOOL NOT NULL); NOTICE: CREATE TABLE will create implicit sequence "test_f1_seq" for serial column "test.f1" CREATE TABLE gaussdb=# INSERT IGNORE INTO test values(NULL,NULL); WARNING: null value in column "f1" violates not-null constraint DETAIL: Failing row contains (null, null). WARNING: null value in column "f2" violates not-null constraint DETAIL: Failing row contains (null, null). INSERT 0 1 gaussdb=# SELECT * FROM test; f1   f2 -----+----- 0   f (1 row)</pre> </li> <li>- MySQL的行为                     <pre>mysql&gt; CREATE TABLE test(f1 SERIAL, f2 BOOL NOT NULL); Query OK, 0 rows affected (0.00 sec)  mysql&gt; INSERT IGNORE INTO test values(NULL,NULL); Query OK, 1 row affected, 1 warning (0.00 sec)  mysql&gt; SELECT * FROM test;</pre> </li> </ul> </li> </ul> |

| 序号 | MySQL数据库功能概述 | 详细语法说明 | GaussDB数据库实现差异                                                                               |
|----|--------------|--------|----------------------------------------------------------------------------------------------|
|    |              |        | <pre>+----+----+   f1   f2   +----+----+   1   0   +----+----+ 1 row in set (0.00 sec)</pre> |

### 8.2.2.8.3 DCL

| 序号 | 概述                    | 详细语法说明                                                                               | 差异                                                                                                                                                                                       |
|----|-----------------------|--------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1  | 支持SET用户自定义变量          | SET                                                                                  | <ul style="list-style-type: none"> <li>自定义变量长度的差异。例如： <ul style="list-style-type: none"> <li>MySQL自定义变量名长度没有约束。</li> <li>GaussDB自定义变量名长度不超过64字节，超过部分的变量名会截断并提示告警。</li> </ul> </li> </ul> |
| 2  | SET TRANSACTION 语法兼容  | SET TRANSACTION                                                                      | MySQL可以设置当前会话（session）和全局（global）的事务隔离级别和读写模式，GaussDB设置当前会话需要设置参数 b_format_behavior_compat_options 包含 set_session_transaction，设置全局只对当前数据库生效。                                             |
| 3  | SET NAMES指定 COLLATE字句 | SET [ SESSION   LOCAL ] NAMES {'charset_name' [COLLATE 'collation_name']   DEFAULT}; | GaussDB暂不支持指定 charset_name 与数据库字符集不同。具体请参考《开发者指南》中“SQL 参考 > SQL语法 > S > SET ” 章节。                                                                                                        |

### 8.2.2.9 驱动

#### 8.2.2.9.1 JDBC



## JDBC 接口参考

### 获取结果集中的数据

ResultSet对象提供了丰富的方法，以获取结果集中的数据。获取数据常用的方法如表1所示，其他方法请参考JDK官方文档。

表 8-45 ResultSet 对象的常用方法

| 方法                                   | 描述                | 差异                                                                       |
|--------------------------------------|-------------------|--------------------------------------------------------------------------|
| int getInt(int columnIndex)          | 按列标获取 int 型数据。    | -                                                                        |
| int getInt(String columnLabel)       | 按列名获取 int 型数据。    | -                                                                        |
| String getString(int columnIndex)    | 按列标获取 String 型数据。 | 字段类型为整型且带有ZEROFILL属性时，GaussDB按照ZEROFILL属性要求的宽度信息用0进行补位后输出结果，MySQL直接输出结果。 |
| String getString(String columnLabel) | 按列名获取 String 型数据。 | 字段类型为整型且带有ZEROFILL属性时，GaussDB按照ZEROFILL属性要求的宽度信息用0进行补位后输出结果，MySQL直接输出结果。 |
| Date getDate(int columnIndex)        | 按列标获取 Date 型数据    | -                                                                        |
| Date getDate(String columnLabel)     | 按列名获取 Date 型数据。   | -                                                                        |

## 8.2.3 MySQL 数据库兼容性 M-Compatibility 模式

### 8.2.3.1 MySQL 数据库兼容性概述

本章节主要介绍GaussDB数据库的M-Compatibility兼容模式与MySQL5.7数据库的兼容性对比信息。仅介绍505.1版本后新增的兼容性特性，特性的相关规格和约束建议在开发者指南中查看。

为了适配MySQL语句及pymysql、JDBC等多种连接方式，GaussDB新增了M-Compatibility协议。

GaussDB数据库在数据类型、SQL功能和数据库对象等基本功能上与MySQL数据库兼容。

GaussDB的执行计划和优化、explain显示结果与MySQL不同。

由于GaussDB数据库与MySQL数据库底层框架实现存在差异，GaussDB数据库与MySQL数据库仍存在部分差异。

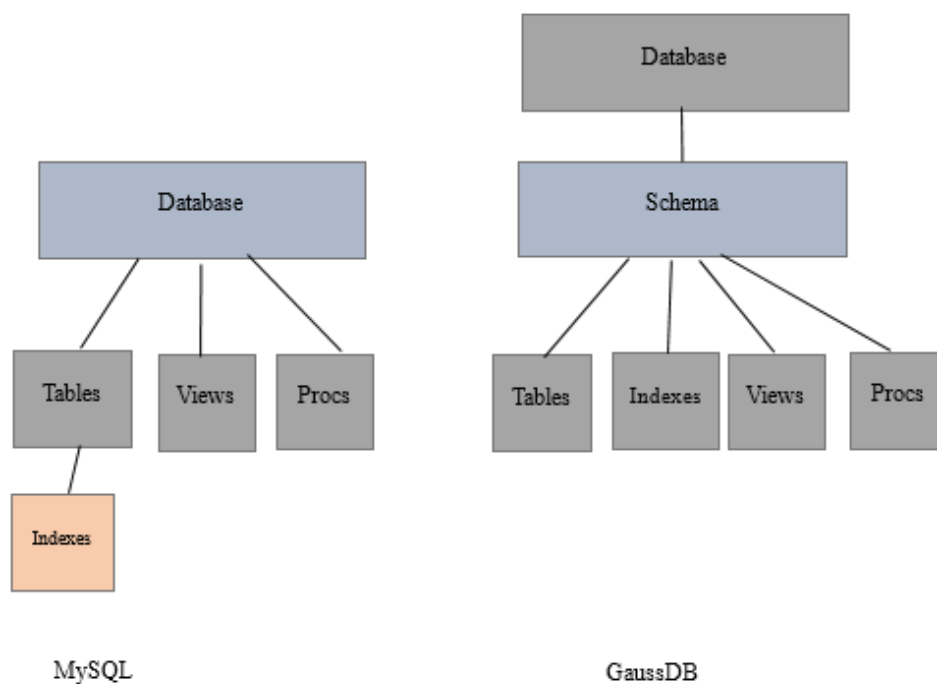
## 说明

由于M-Compatibility的底层架构与MySQL存在差异，对于information\_schema和m\_schema下与MySQL名称相同的Schema（具体参见M-Compatibility开发指南第六章），其查询性能可能存在差异。例如count函数无法做执行上的优化，表现为select \*和select count(\*)语句耗时近似。

## Database 和 Schema 设计

MySQL的数据对象包括DATABASE、TABLE、INDEX、VIEW、TRIGGER、PROC等，MySQL的对象层次跟GaussDB的对应关系是从上至下且1对多包含关系。如下图所示：

图 8-1 MySQL 和 GaussDB 中 Database 和 Schema 之间的差异



- 在MySQL中Database和Schema是同义词；而在GaussDB中，一个Database下可以有多个Schema。在该特性中，每个MySQL中的Database都被映射到GaussDB的一个Schema。
- 在MySQL中，INDEX从属于一个TABLE，但在GaussDB中，INDEX从属于一个Schema。这个差异导致INDEX名在GaussDB中要求在Schema内唯一，但在MySQL中仅要在在一个表内唯一。这个差异将作为当前约束予以保留。

### 8.2.3.2 数据类型

GaussDB的数据类型的大部分功能场景与MySQL一致，但存在部分差异。

- 除特别说明，部分数据类型精度、标度、位数大小等不支持用浮点型数值定义，建议使用合法的整型数值定义。

### 8.2.3.2.1 数值数据类型

表 8-46 整数类型

| 序号 | MySQL数据库                     | GaussDB数据库 | 差异                                                                                                                                                                                                                                                                                                                       |
|----|------------------------------|------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1  | BOOL                         | 支持, 存在差异   | 输出格式: GaussDB中SELECT TRUE/FALSE输出结果为t/f, MySQL为1/0。                                                                                                                                                                                                                                                                      |
| 2  | BOOLEAN                      | 支持, 存在差异   | MySQL: BOOL/BOOLEAN类型实际映射为TINYINT类型。                                                                                                                                                                                                                                                                                     |
| 3  | TINYINT[(M)]<br>[UNSIGNED]   | 支持, 存在差异   | 详细请参见说明。                                                                                                                                                                                                                                                                                                                 |
| 4  | SMALLINT[(M)]<br>[UNSIGNED]  | 支持, 存在差异   | 详细请参见说明。                                                                                                                                                                                                                                                                                                                 |
| 5  | MEDIUMINT[(M)]<br>[UNSIGNED] | 支持, 存在差异   | MySQL存储MEDIUMINT数据需要3字节。<br><ul style="list-style-type: none"> <li>带符号的范围是-8,388,608 ~ +8,388,607。</li> <li>无符号的范围是0 ~ +16,777,215。</li> </ul> GaussDB映射为INT类型, 存储需要4字节, 通过边界值判断限制取值范围。<br><ul style="list-style-type: none"> <li>带符号的范围是-8,388,608 ~ +8,388,607。</li> <li>无符号的范围是0 ~ +16,777,215。</li> </ul> 其他差异请参见说明。 |
| 6  | INT[(M)]<br>[UNSIGNED]       | 支持, 存在差异   | 详细请参见说明。                                                                                                                                                                                                                                                                                                                 |
| 7  | INTEGER[(M)]<br>[UNSIGNED]   | 支持, 存在差异   | 详细请参见说明。                                                                                                                                                                                                                                                                                                                 |
| 8  | BIGINT[(M)]<br>[UNSIGNED]    | 支持, 存在差异   | 详细请参见说明。                                                                                                                                                                                                                                                                                                                 |

 说明

输入格式:

- MySQL:  
整型类型对于类似” 1.2.3.4.5” 有多个小数点的字符串形式输入，在宽松模式下MySQL会发生错误解析，抛出WARNING并插表成功，例如将” 1.2.3.4.5” 插入表后值为12。
- GaussDB:  
整型类型对于类似” 1.2.3.4.5” 有多个小数点的的字符串形式输入，在宽松模式下，会将第二个小数点后的字符当作非法字符全部截断，抛出WARNING并插表成功，例如将” 1.2.3.4.5” 插入表后值为1，” 1.6.3.4.5” 插入表后值为2。

表 8-47 任意精度类型

| 序号 | MySQL数据库         | GaussDB数据库 | 差异                                                                             |
|----|------------------|------------|--------------------------------------------------------------------------------|
| 1  | DECIMAL[(M[,D])] | 支持，存在差异    | MySQL decimal用一个9*9的数组存储数值，整数部分和小数部分分开存储，超过该长度时优先截小数部分。GaussDB只会在整数位数超过81位时截断。 |
| 2  | NUMERIC[(M[,D])] | 支持，存在差异    |                                                                                |
| 3  | DEC[(M[,D])]     | 支持，存在差异    |                                                                                |
| 4  | FIXED[(M[,D])]   | 支持，存在差异    |                                                                                |

表 8-48 浮点类型

| 序号 | MySQL数据库                | GaussDB数据库 | 差异                                   |
|----|-------------------------|------------|--------------------------------------|
| 1  | FLOAT[(M,D)]            | 支持，存在差异    | FLOAT数据类型不支持KEY键值分区策略分区表。            |
| 2  | FLOAT(p)                | 支持，存在差异    | FLOAT数据类型不支持KEY键值分区策略分区表。            |
| 3  | DOUBLE[(M,D)]           | 支持，存在差异    | DOUBLE数据类型不支持KEY键值分区策略分区表。           |
| 4  | DOUBLE PRECISION[(M,D)] | 支持，存在差异    | DOUBLE PRECISION数据类型不支持KEY键值分区策略分区表。 |
| 5  | REAL[(M,D)]             | 支持，存在差异    | REAL数据类型不支持KEY值分区策略分区表。              |

## 8.2.3.2.2 日期与时间数据类型

表 8-49 日期与时间数据类型

| 序号 | MySQL数据库         | GaussDB数据库 | 差异                                                                                                                                                                                                                                                                                       |
|----|------------------|------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1  | DATE             | 支持，存在差异。   | GaussDB支持date数据类型，与MySQL相比规格上存在如下差异： <ul style="list-style-type: none"><li>● 输入格式：<ul style="list-style-type: none"><li>- 分隔符：反斜杠\在MySQL和GaussDB中都视为转义，但MySQL支持\0，GaussDB暂不支持，因此反斜杠作为分隔符且分隔符后为字符0时GaussDB会报错。</li></ul></li></ul>                                                        |
| 2  | DATETIME[(fsp)]  | 支持，存在差异。   | GaussDB支持datetime数据类型，与MySQL相比规格上存在如下差异： <ul style="list-style-type: none"><li>● 输入格式：<ul style="list-style-type: none"><li>- 分隔符：反斜杠\在MySQL和GaussDB中都视为转义，但MySQL支持\0，GaussDB暂不支持，因此反斜杠作为分隔符且分隔符后为字符0时GaussDB会报错。</li></ul></li></ul>                                                    |
| 3  | TIMESTAMP[(fsp)] | 支持，存在差异。   | GaussDB支持timestamp数据类型，与MySQL相比规格上存在如下差异： <ul style="list-style-type: none"><li>● 输入格式：<ul style="list-style-type: none"><li>- 分隔符：反斜杠\在MySQL和GaussDB中都视为转义，但MySQL支持\0，GaussDB暂不支持，因此反斜杠作为分隔符且分隔符后为字符0时GaussDB会报错。</li></ul></li></ul>                                                   |
| 4  | TIME[(fsp)]      | 支持，存在差异。   | GaussDB支持time数据类型，与MySQL相比规格上存在如下差异： <ul style="list-style-type: none"><li>● 输入格式：<ul style="list-style-type: none"><li>- 分隔符：反斜杠\在MySQL和GaussDB中都视为转义，但MySQL支持\0，GaussDB暂不支持，因此反斜杠作为分隔符且分隔符后为字符0时GaussDB会报错。</li></ul></li><li>● 当时间类型的时、分、秒、纳秒为0时，GaussDB和MySQL可能存在符号位不同的情况。</li></ul> |
| 5  | YEAR[(4)]        | 支持。        | -                                                                                                                                                                                                                                                                                        |

 说明

- GaussDB不支持ODBC语法的字面量：
  - { d 'str' }
  - { t 'str' }
  - { ts 'str' }
- GaussDB支持标准SQL字面量，且类型关键字后面可选择添加精度，MySQL不支持：
  - DATE[(n)] 'str'
  - TIME[(n)] 'str'
  - TIMESTAMP[(n)] 'str'

### 8.2.3.2.3 字符串数据类型

表 8-50 字符串数据类型

| 序号 | MySQL数据库   | GaussDB数据库 | 差异                                                                                                                                                                                                                                                                                                                                                                          |
|----|------------|------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1  | CHAR(M)    | 支持，存在差异    | <ul style="list-style-type: none"> <li>• 输入格式：                             <ul style="list-style-type: none"> <li>- 输入二进制或十六进制字符串，GaussDB输出为十六进制，MySQL中根据ASCII码表转义，无法转义的输出为空。</li> <li>- GaussDB不支持转义字符输入，MySQL支持。</li> </ul> </li> </ul>                                                                                                                                   |
| 2  | VARCHAR(M) | 支持，存在差异    | <ul style="list-style-type: none"> <li>• 输入格式：                             <ul style="list-style-type: none"> <li>- GaussDB的自定义函数参数和返回值不支持长度校验，存储过程参数不支持长度校验，MySQL支持。</li> <li>- GaussDB的自定义函数和存储过程中的临时变量支持长度校验以及严格宽松模式下的报错和截断告警，MySQL不支持。</li> <li>- GaussDB不支持转义字符输入，MySQL支持。</li> <li>- 输入二进制或十六进制字符串，GaussDB输出为十六进制，MySQL中根据ASCII码表转义，无法转义的输出为空。</li> </ul> </li> </ul> |

| 序号 | MySQL数据库   | GaussDB数据库 | 差异                                                                                                                                                                                                                                                                                                                                                                                                  |
|----|------------|------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 3  | TINYTEXT   | 支持，存在差异    | <ul style="list-style-type: none"> <li>● 输入格式：                             <ul style="list-style-type: none"> <li>- GaussDB不支持转义字符输入，MySQL支持。</li> <li>- 默认值：创建表列时语法上允许设置默认值，MySQL不允许设置默认值。</li> <li>- 输入二进制或十六进制字符串，GaussDB输出为十六进制，MySQL中根据ASCII码表转义，无法转义的输出为空。</li> </ul> </li> <li>● 主键：MySQL中TINYTEXT类型不支持主键，GaussDB支持。</li> <li>● 索引：MySQL中TINYTEXT类型不支持除前缀索引外其他索引方法，GaussDB支持。</li> </ul>     |
| 4  | TEXT       | 支持，存在差异    | <ul style="list-style-type: none"> <li>● 输入格式：                             <ul style="list-style-type: none"> <li>- GaussDB不支持转义字符输入，MySQL支持。</li> <li>- 默认值：创建表列时语法上允许设置默认值，MySQL不允许设置默认值。</li> <li>- 输入二进制或十六进制字符串，GaussDB输出为十六进制，MySQL中根据ASCII码表转义，无法转义的输出为空。</li> </ul> </li> <li>● 主键：MySQL中TEXT类型不支持主键，GaussDB支持。</li> <li>● 索引：MySQL中TEXT类型不支持除前缀索引外其他索引方法，GaussDB支持。</li> </ul>             |
| 5  | MEDIUMTEXT | 支持，存在差异    | <ul style="list-style-type: none"> <li>● 输入格式：                             <ul style="list-style-type: none"> <li>- GaussDB不支持转义字符输入，MySQL支持。</li> <li>- 默认值：创建表列时语法上允许设置默认值，MySQL不允许设置默认值。</li> <li>- 输入二进制或十六进制字符串，GaussDB输出为十六进制，MySQL中根据ASCII码表转义，无法转义的输出为空。</li> </ul> </li> <li>● 主键：MySQL中MEDIUMTEXT类型不支持主键，GaussDB支持。</li> <li>● 索引：MySQL中MEDIUMTEXT类型不支持除前缀索引外其他索引方法，GaussDB支持。</li> </ul> |

| 序号 | MySQL数据库 | GaussDB数据库 | 差异                                                                                                                                                                                                                                                                                                                                                                                                       |
|----|----------|------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 6  | LONGTEXT | 支持，存在差异    | <ul style="list-style-type: none"><li>● 输入格式：<ul style="list-style-type: none"><li>- GaussDB只支持不超过1G字节长度，MySQL支持4G-1字节长度。</li><li>- GaussDB不支持转义字符输入，MySQL支持。</li><li>- 默认值：创建表列时语法上允许设置默认值，MySQL不允许设置默认值。</li><li>- 输入二进制或十六进制字符串，GaussDB输出为十六进制，MySQL中根据ASCII码表转义，无法转义的输出为空。</li></ul></li><li>● 主键：MySQL中LONGTEXT类型不支持主键，GaussDB支持。</li><li>● 索引：MySQL中LONGTEXT类型不支持除前缀索引外其他索引方法，GaussDB支持。</li></ul> |



## 8.2.3.2.4 二进制数据类型

表 8-51 二进制数据类型

| 序号 | MySQL数据库     | GaussDB数据库 | 差异                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|----|--------------|------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1  | BINARY[(M)]  | 支持, 存在差异   | <ul style="list-style-type: none"><li>● 输入格式:<ul style="list-style-type: none"><li>- GaussDB不支持转义字符输入, MySQL支持。</li><li>- 输入二进制或十六进制字符串, GaussDB输出为十六进制, MySQL中根据ASCII码表转义, 无法转义的输出为空。</li><li>- 插入字符串长度小于目标长度时, GaussDB填充符是0x20, MySQL是0x00。</li></ul></li><li>● 字符集: 默认字符集为数据库初始化字符集, MySQL默认类型字符集为BINARY字符集。</li><li>● 输出格式:<ul style="list-style-type: none"><li>- JDBC协议输出时BINARY类型的末尾空格显示为空格, MySQL末尾空格显示为\x00。</li><li>- 宽松模式下, BINARY类型面对输入超过n的字节数的字符输入(例如中文字符), 会将超限的整个字符截断。MySQL中会将超限的整个字符的前几位满足n范围内的字节信息保留, 但输出时字符信息显示乱码。</li></ul></li></ul> <p><b>说明</b><br/>GaussDB中, 由于BINARY类型填充符和\0截断与MySQL的差异, 在操作符比较计算, 字符串相关系统函数计算, 索引匹配, 数据导入导出等场景下与MySQL的表现会存在差异。差异场景请见本节示例。</p> |
| 2  | VARBINARY(M) | 支持, 存在差异   | <ul style="list-style-type: none"><li>● 输入格式:<ul style="list-style-type: none"><li>- GaussDB不支持转义字符输入, MySQL支持。</li><li>- 输入二进制或十六进制字符串, GaussDB输出为十六进制, MySQL中根据ASCII码表转义, 无法转义的输出为空。</li></ul></li><li>● 字符集: 默认字符集为数据库初始化字符集, MySQL默认类型字符集为BINARY字符集。</li><li>● 输出格式: JDBC协议输出时BINARY类型的末尾空格显示为空格, MySQL末尾空格显示为\x00。</li></ul>                                                                                                                                                                                                                                                                                                                                                      |

| 序号 | MySQL数据库   | GaussDB数据库 | 差异                                                                                                                                                                                                                                                                                                                                                                                                  |
|----|------------|------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 3  | TINYBLOB   | 支持，存在差异    | <ul style="list-style-type: none"> <li>● 输入格式：                             <ul style="list-style-type: none"> <li>- GaussDB不支持转义字符输入，MySQL支持。</li> <li>- 默认值：创建表列时语法上允许设置默认值，MySQL不允许设置默认值。</li> <li>- 输入二进制或十六进制字符串，GaussDB输出为十六进制，MySQL中根据ASCII码表转义，无法转义的输出为空。</li> </ul> </li> <li>● 主键：MySQL中TINYBLOB类型不支持主键，GaussDB支持。</li> <li>● 索引：MySQL中TINYBLOB类型不支持除前缀索引外其他索引方法，GaussDB支持。</li> </ul>     |
| 4  | BLOB       | 支持，存在差异    | <ul style="list-style-type: none"> <li>● 输入格式：                             <ul style="list-style-type: none"> <li>- GaussDB不支持转义字符输入，MySQL支持。</li> <li>- 默认值：创建表列时语法上允许设置默认值，MySQL不允许设置默认值。</li> <li>- 输入二进制或十六进制字符串，GaussDB输出为十六进制，MySQL中根据ASCII码表转义，无法转义的输出为空。</li> </ul> </li> <li>● 主键：MySQL中BLOB类型不支持主键，GaussDB支持。</li> <li>● 索引：MySQL中BLOB类型不支持除前缀索引外其他索引方法，GaussDB支持。</li> </ul>             |
| 5  | MEDIUMBLOB | 支持，存在差异    | <ul style="list-style-type: none"> <li>● 输入格式：                             <ul style="list-style-type: none"> <li>- GaussDB不支持转义字符输入，MySQL支持。</li> <li>- 默认值：创建表列时语法上允许设置默认值，MySQL不允许设置默认值。</li> <li>- 输入二进制或十六进制字符串，GaussDB输出为十六进制，MySQL中根据ASCII码表转义，无法转义的输出为空。</li> </ul> </li> <li>● 主键：MySQL中MEDIUMBLOB类型不支持主键，GaussDB支持。</li> <li>● 索引：MySQL中MEDIUMBLOB类型不支持除前缀索引外其他索引方法，GaussDB支持。</li> </ul> |

| 序号 | MySQL数据库 | GaussDB数据库 | 差异                                                                                                                                                                                                                                                                                                                                                                                                                              |
|----|----------|------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 6  | LONGBLOB | 支持，存在差异    | <ul style="list-style-type: none"> <li>取值范围：只支持不超过1G字节长度，MySQL支持4G-1字节长度。</li> <li>输入格式：                             <ul style="list-style-type: none"> <li>GaussDB不支持转义字符输入，MySQL支持。</li> <li>默认值：创建表列时语法上允许设置默认值，MySQL不允许设置默认值。</li> <li>输入二进制或十六进制字符串，GaussDB输出为十六进制，MySQL中根据ASCII码表转义，无法转义的输出为空。</li> </ul> </li> <li>主键：MySQL中LONGBLOB类型不支持主键，GaussDB支持。</li> <li>索引：MySQL中LONGBLOB类型不支持除前缀索引外其他索引方法，GaussDB支持。</li> </ul> |
| 7  | BIT[(M)] | 支持，存在差异    | <ul style="list-style-type: none"> <li>输出格式：所有输出按照二进制字符串形式输出。MySQL中根据ASCII码表转义，无法转义的输出为空。</li> </ul>                                                                                                                                                                                                                                                                                                                            |

示例：

```

-- GaussDB场景
m_db=# create table test(a binary(10));
CREATE TABLE
m_db=# insert into test values(0x8000);
INSERT 0 1
m_db=# select hex(a) from test;
      hex
-----
80202020202020202020
(1 row)

m_db=# select * from test where hex(a) = 80000000000000000000;
 a
--
(0 rows)

m_db=# CREATE TABLE test2(a binary(10));
CREATE TABLE
m_db=# INSERT INTO test2 VALUES(0x80008000);
INSERT 0 1
m_db=# SELECT hex(a) FROM test2;
      hex
-----
80202020202020202020
(1 row)

-- Mysql场景
mysql> create table test(a binary(10));
Query OK, 0 rows affected (0.01 sec)

mysql> insert into test values(0x8000);

```

```

Query OK, 1 row affected (0.00 sec)

mysql> select hex(a) from test;
+-----+
| hex(a) |
+-----+
| 80000000000000000000 |
+-----+
1 row in set (0.00 sec)

mysql> select * from test where hex(a) = 80000000000000000000;
+-----+
| a |
+-----+
|  |
+-----+
1 row in set (0.00 sec)

mysql> CREATE TABLE test2(a binary(10));
Query OK, 0 rows affected (0.00 sec)

mysql> INSERT INTO test2 VALUES(0x80008000);
Query OK, 1 row affected (0.00 sec)

mysql> SELECT hex(a) FROM test2;
+-----+
| hex(a) |
+-----+
| 80008000000000000000 |
+-----+
1 row in set (0.00 sec)
    
```

### 8.2.3.2.5 数据类型支持的属性

表 8-52 数据类型支持的属性

| 序号 | MySQL数据库           | GaussDB数据库 |
|----|--------------------|------------|
| 1  | NULL               | 支持         |
| 2  | NOT NULL           | 支持         |
| 3  | DEFAULT            | 支持         |
| 4  | ON UPDATE          | 支持         |
| 4  | PRIMARY KEY        | 支持         |
| 5  | AUTO_INCREMENT     | 支持         |
| 6  | CHARACTER SET name | 支持         |
| 7  | COLLATE name       | 支持         |

### 8.2.3.2.6 数据类型转换

不同的数据类型之间支持转换。有如下场景涉及到数据类型转换：

- 操作符（比较操作符、运算操作符等）的操作数的数据类型不一致。常见于查询条件或者关联条件中的比较运算。
- 函数调用时实参和形参的数据类型不一致。

- DML语句要更新（包括INSERT、UPDATE、MERGE、REPLACE等）的目标列，数据的类型和列的定义类型不一致。
- 显式的类型转换：CAST(expr AS datatype)，将expr表达式类型转换为datatype类型。
- 集合运算（UNION、MINUS、EXCEPT、INTERSECT）确定最终投影列的目标数据类型后，各个SELECT查询的投影列的类型和目标数据类型不一致。
- 其他表达式计算场景，根据不同表达式的数据类型，来决定用于比较或者最终结果的目标数据类型。

数据类型转换差异点主要分为三类：隐式转换，显式转换和UNION/CASE。

#### 双冒号转换差异点：

- GaussDB中使用双冒号将函数入参转换为期望类型可能导致结果超出预期，且MySQL中无双冒号功能。

示例：

```
m_db=# SELECT POW("12"::VARBINARY,"12"::VARBINARY);
ERROR: value out of range: overflow
CONTEXT: referenced column: pow

varbinary col
m_db=# CREATE TABLE test_varbinary (
    A VARBINARY(10)
);
m_db=# INSERT INTO test_varbinary VALUES ('12');
m_db=# SELECT POW(A, A) FROM test_varbinary;
    pow
-----
8916100448256
(1 row)
```

#### 隐式类型转换差异点：

- GaussDB中统一平铺成小类型到小类型的转换规则，MySQL中使用小类型转大类型，大类型转小类型的转换规则。
- GaussDB中隐式转换因数据类型本身差异点，输出格式存在部分行为不一致。
- GaussDB中的隐式转换，BIT数据类型到字符数据类型和二进制数据类型转换，输出存在部分行为不一致。GaussDB输出为十六进制，MySQL中根据ASCII码表转义，无法转义的输出为空。

示例：

```
m_db=# CREATE TABLE bit_storage (
    VS_COL1 BIT(4),
    VS_COL2 BIT(4),
    VS_COL3 BIT(4),
    VS_COL4 BIT(4),
    VS_COL5 BIT(4),
    VS_COL6 BIT(4),
    VS_COL7 BIT(4),
    VS_COL8 BIT(4)
);
m_db=# CREATE TABLE string_storage (
    VS_COL1 BLOB,
    VS_COL2 TINYBLOB,
    VS_COL3 MEDIUMBLOB,
    VS_COL4 LONGBLOB,
    VS_COL5 TEXT,
    VS_COL6 TINYTEXT,
    VS_COL7 MEDIUMTEXT,
    VS_COL8 LONGTEXT
);
m_db=# INSERT INTO bit_storage VALUES(B'101', B'101', B'101', B'101', B'101', B'101', B'101', B'101');
```

```

m_db=# INSERT INTO string_storage SELECT * FROM bit_storage;
m_db=# SELECT * FROM string_storage;
VS_COL1 | VS_COL2 | VS_COL3 | VS_COL4 | VS_COL5 | VS_COL6 | VS_COL7 | VS_COL8
-----+-----+-----+-----+-----+-----+-----+-----
\x05   |\x05   |\x05   |\x05   |\x05   |\x05   |\x05   |\x05
(1 row)
m_db=# DROP TABLE bit_storage, string_storage;

mysql> CREATE TABLE bit_storage (
  VS_COL1 BIT(4),
  VS_COL2 BIT(4),
  VS_COL3 BIT(4),
  VS_COL4 BIT(4),
  VS_COL5 BIT(4),
  VS_COL6 BIT(4),
  VS_COL7 BIT(4),
  VS_COL8 BIT(4)
);
mysql> CREATE TABLE bit_storage (
  VS_COL1 BIT(4),
  VS_COL2 BIT(4),
  VS_COL3 BIT(4),
  VS_COL4 BIT(4),
  VS_COL5 BIT(4),
  VS_COL6 BIT(4),
  VS_COL7 BIT(4),
  VS_COL8 BIT(4)
);
mysql> INSERT INTO bit_storage VALUES(B'101', B'101', B'101', B'101', B'101', B'101', B'101', B'101');
mysql> INSERT INTO string_storage SELECT * FROM bit_storage;
mysql> SELECT * FROM string_storage;
-----+-----+-----+-----+-----+-----+-----+-----
| VS_COL1 | VS_COL2 | VS_COL3 | VS_COL4 | VS_COL5 | VS_COL6 | VS_COL7 | VS_COL8 |
-----+-----+-----+-----+-----+-----+-----+-----
|         |         |         |         |         |         |         |         |
-----+-----+-----+-----+-----+-----+-----+-----
1 row in set (0.00 sec)
mysql> DROP TABLE bit_storage, string_storage;

```

- 中间带有0x00的二进制或十六进制字符串插入二进制数据类型时，GaussDB部分插入，0x00后的字符会截断，MySQL可以完整插入。

示例：

```

m_db=# CREATE TABLE blob_storage (
  A BLOB
);
m_db=# INSERT INTO blob_storage VALUES (0xBB00BB);
m_db=# SELECT hex(A) FROM blob_storage;
hex
----
BB
(1 row)
m_db=# DROP TABLE blob_storage;

mysql> CREATE TABLE blob_storage (
  A TEXT
);
mysql> INSERT INTO blob_storage VALUES (0xBB00BB);
mysql> SELECT hex(A) FROM blob_storage;
-----+
| hex(a) |
-----+
| BB00BB |
-----+
1 row in set (0.01 sec)
mysql> DROP TABLE blob_storage;

```

- 中间带有0x00的二进制或十六进制字符串插入字符串数据类型时，GaussDB部分插入，0x00后的字符会截断，MySQL严格模式不允许插入，宽松模式插入空字符串。

示例：

```
m_db=# CREATE TABLE text_storage (  
    A TEXT  
);  
m_db=# INSERT INTO text_storage VALUES (b'101110110000000010111011');  
m_db=# SELECT hex(A) FROM text_storage;  
hex  
-----  
BB  
(1 row)  
m_db=# DROP TABLE text_storage;  
  
mysql> CREATE TABLE text_storage (  
    A TEXT  
);  
mysql> INSERT INTO text_storage VALUES (b'101110110000000010111011');  
ERROR 1366 (HY000): Incorrect string value: '\xBB\x00\xBB' for column 'A' at row 1  
mysql> SELECT hex(A) FROM text_storage;  
Empty set (0.00 sec)  
mysql> SET SQL_MODE="";  
mysql> INSERT INTO text_storage VALUES (b'101110110000000010111011');  
mysql> SELECT hex(A) FROM text_storage;  
+-----+  
| hex(A) |  
+-----+  
|      |  
+-----+  
1 row in set (0.01 sec)  
mysql> DROP TABLE text_storage;
```

- WHERE子句中只带有普通字符串，GaussDB中't'、'true'、'yes'、'y'、'on'返回TRUE，'no'、'f'、'off'、'false'、'n'返回FALSE，其余字符串报错。MySQL通过字符串转换为INT1判断返回TRUE/FALSE。

示例：

```
m_db=# CREATE TABLE test_where (  
    A INT  
);  
m_db=# INSERT INTO test_where VALUES (1);  
m_db=# SELECT * FROM test_where WHERE '111';  
ERROR: invalid input syntax for type boolean: "111"  
LINE 1: select * from test_where where '111';  
m_db=# DROP TABLE test_where;  
  
mysql> CREATE TABLE test_where (  
    A INT  
);  
mysql> INSERT INTO test_where VALUES (1);  
mysql> SELECT * FROM test_where WHERE '111';  
+-----+  
| a  |  
+-----+  
|  1 |  
+-----+  
1 row in set (0.01 sec)  
mysql> DROP TABLE test_where;
```

- 对于YEAR类型的输入，在将字符串转换为整型的过程中，MySQL考虑科学计数法，GaussDB暂不支持，统一做截断处理。

示例：

```
m_db=# CREATE TABLE test_year (  
    A YEAR  
);
```

```
m_db=# INSERT INTO test_year VALUES ('2E3x');
WARNING: Data truncated for column.
LINE 1: insert into t1 values ('2E3x');
      ^
CONTEXT: referenced column: a
m_db=# SELECT * FROM test_year ORDER BY A;
a
-----
2002
(1 row)
m_db=# DROP TABLE test_year;

mysql> CREATE TABLE test_year (
      A YEAR
);
mysql> INSERT INTO test_year VALUES ('2E3x');
mysql> SELECT * FROM test_year ORDER BY A;
+-----+
| a |
+-----+
| 2000 |
+-----+
1 row in set (0.01 sec)
mysql> DROP TABLE test_year;
```

### 显式类型转换差异点:

GaussDB中平铺成对各目标类型的转换规则，MySQL中使用C++多态重载函数，在嵌套场景中存在不一致行为。

#### 示例:

```
m_db=# SELECT CAST(GREATEST(date'2023-01-01','2023-01-01') AS SIGNED);
WARNING: Truncated incorrect INTEGER value: '2023-01-01'
CONTEXT: referenced column: cast
cast
-----
2023
(1 row)

mysql> SELECT CAST(GREATEST(date'2023-01-01','2023-01-01') AS SIGNED);
+-----+
| CAST(GREATEST(date'2023-01-01','2023-01-01') AS SIGNED) |
+-----+
| 20230101 |
+-----+
```

### UNION, CASE和相关构造差异点:

- POLYGON + NULL、POINT + NULL、POLYGON + POINT组合在MySQL中均返回GEOMETRY类型，GaussDB中未涉及，暂时当做报错处理。
- SET和ENUM两种类型暂未支持，暂时当做报错处理。
- 常量类型和其他类型做类型聚合的时候，输出类型的精度为其他类型的精度。如“select "helloworld" union select p from t;”的结果的精度为属性p的精度。
- 定点常量和不带精度约束的类型（非字符串类型如int、bool、year等，聚合结果类型为定点类型）聚合时，精度约束会按照定点数默认精度31输出。
- merge rule差异：  
MySQL 5.7中YEAR和TINYINT、INT、MEDIUMINT、BIGINT、BOOL聚合的结果类型为带UNSIGNED的类型，GaussDB的结果类型为不带UNSIGNED的类型；MySQL中BIT和INT、NUMERIC、FLOAT、DOUBLE等数值类型的聚合类型为VARBINARY类型，GaussDB中BIT和INT|NUMERIC聚合为NUMERIC类型，和FLOAT或者DOUBLE类型聚合时，结果为DOUBLE类型，和无符号整型聚合时为UINT8类型。



- MySQL中BINARY和CHAR填充字符不相同，BINARY填充'\0'，CHAR填充空格，GaussDB中BINARY和CHAR都是填充空格。

### 8.2.3.3 系统函数

GaussDB数据库兼容绝大多数MySQL的系统函数，但存在部分差异。只能使用M-Compatibility兼容模式下的系统函数，不能使用原GaussDB的系统函数，避免产生非预期的结果。当前存在原GaussDB的系统函数和MySQL系统函数同名，但是M-Compatibility兼容模式下尚未支持这些函数的情况；一部分未支持的同名函数会提示用户在M-Compatibility兼容模式下不支持，另外一部分同名函数仍然保持原GaussDB系统函数的行为。用户禁止使用这些同名函数，避免产生非预期的结果。具体同名函数列表如下：

**表 8-53** M-Compatibility 兼容模式下提示不支持的同名函数

|                     |               |                    |              |                   |
|---------------------|---------------|--------------------|--------------|-------------------|
| cot                 | isEmpty       | json_append        | json_array   | json_array_append |
| json_array_insert   | json_contains | json_contains_path | json_depth   | json_extract      |
| json_insert         | json_keys     | json_length        | json_merge   | json_merge_patch  |
| json_merge_preserve | json_object   | json_quote         | json_remove  | json_valid        |
| json_replace        | json_search   | json_set           | json_type    | json_unquote      |
| last_insert_id      | md5           | mod                | octet_length | overlaps          |
| point               | radians       | stddev_pop         | stddev_samp  | var_pop           |
| var_samp            | variance      | lag                | lead         | -                 |

**表 8-54** M-Compatibility 兼容模式下保持原 GaussDB 系统函数行为的同名函数

|                |               |         |              |             |
|----------------|---------------|---------|--------------|-------------|
| ceil           | decode        | encode  | format       | instr       |
| position       | round         | stddev  | regexp_instr | regexp_like |
| regexp_replace | regexp_substr | row_num | -            | -           |

 说明

M-Compatibility兼容模式下，系统函数存在以下公共差异：

1. 系统函数的返回值类型仅考虑入参node类型为Var（表中数据）和Const（常量输入）类型时的情况与MySQL保持一致，其他情况（如入参为运算表达式、函数表达式等）可能返回值的类型与MySQL有差异。
2. 当聚合函数以其他函数、操作符或SELECT子句等表达式作为入参时（如SELECT sum(abs(n)) FROM t;），聚合函数将获取不到入参表达式传递的精度信息，导致函数的结果精度与MySQL有差异。

## 8.2.3.3.1 流程控制函数

表 8-55 流程控制函数列表

| 序号 | MySQL数据库 | GaussDB数据库 | 差异                                                                                               |
|----|----------|------------|--------------------------------------------------------------------------------------------------|
| 1  | IF()     | 支持。        | 当第一个参数为TRUE且第三个参数表达式中存在隐式类型转换错误，或者第一个参数为FALSE且第二个参数表达式中存在隐式类型转换错误时，MySQL会忽略该错误，GaussDB会提示类型转换错误。 |
| 2  | IFNULL() | 支持。        | 第一个参数不为NULL且第二个参数表达式中存在隐式类型转换错误时，MySQL会忽略该错误，GaussDB会提示类型转换错误。                                   |
| 3  | NULLIF() | 支持，有差异。    | 函数返回值类型在MySQL5.7和MySQL8.0中存在差异，考虑到MySQL8.0更合理，因此函数返回值类型兼容MySQL8.0。                               |

## 8.2.3.3.2 日期和时间函数

 说明

以下为GaussDB M-Compatibility兼容性日期时间函数公共说明。

- 《M-Compatibility开发指南》中“函数和操作符”章节函数入参为时间类型表达式的情况：  
时间类型表达式主要包括TEXT、DATETIME、DATE或TIME，但所有可以隐式转换为时间表达式的类型都可以作为入参，比如数字类型可以通过先隐式转化为TEXT，再作为时间类型表达式生效。  
生效的情况根据函数有所不同，比如：DATEDIFF函数由于只计算日期的差值，所以时间表达式会当做日期来解析。TIMESTAMPDIFF函数由于计算时间差值时是根据UNIT来决定的，所以会根据UNIT选择将时间表达式当做DATE或者TIME或者DATETIME来解析。
- 函数入参为无效日期的情况：  
一般而言，日期时间函数支持DATE、DATETIME的范围和MySQL保持一致。DATE支持的范围为'0000-01-01'到'9999-12-31'，DATETIME支持的范围为'0000-01-01 00:00:00'到'9999-12-31 23:59:59'。虽然GaussDB支持的DATE、DATETIME范围大于MySQL，但是越界仍然算无效日期。

大部分时间函数会告警并返回NULL，只有能通过cast正常转换的日期，才是正常合理的日期。

新框架下GaussDB的大部分日期时间函数与MySQL一致，一些函数的差异如下表所示：

**表 8-56** 日期与和时间函数列表

| 序号 | MySQL数据库                                      | GaussDB数据库 | 差异                                                                                                                       |
|----|-----------------------------------------------|------------|--------------------------------------------------------------------------------------------------------------------------|
| 1  | ADDDATE()                                     | 支持         | -                                                                                                                        |
| 2  | ADDTIME()                                     | 支持         | -                                                                                                                        |
| 3  | CONVERT_TZ()                                  | 支持         | -                                                                                                                        |
| 4  | CURDATE()                                     | 支持         | -                                                                                                                        |
| 5  | CURRENT_DATE()/<br>CURRENT_DATE               | 支持         | -                                                                                                                        |
| 6  | CURRENT_TIME()/<br>CURRENT_TIME               | 支持，<br>有差异 | MySQL入参整型值会按照一字节最大值255整数回绕（例：SELECT CURRENT_TIME(257) == SELECT CURRENT_TIME(1)）。<br>GaussDB只支持[0,6]合法值，其他值报错。           |
| 7  | CURRENT_TIMESTAMP(<br>)/<br>CURRENT_TIMESTAMP | 支持，<br>有差异 | MySQL入参整型值会按照一字节最大值255整数回绕（例：SELECT CURRENT_TIMESTAMP(257) == SELECT CURRENT_TIMESTAMP(1)）。<br>GaussDB只支持[0,6]合法值，其他值报错。 |
| 8  | CURTIME()                                     | 支持，<br>有差异 | MySQL入参整型值会按照一字节最大值255整数回绕（例：SELECT CURTIME(257) == SELECT CURTIME(1)）。<br>GaussDB只支持[0,6]合法值，其他值报错。                     |
| 9  | DATE()                                        | 支持         | -                                                                                                                        |
| 10 | DATE_ADD()                                    | 支持         | -                                                                                                                        |
| 11 | DATE_FORMAT()                                 | 支持         | -                                                                                                                        |
| 12 | DATE_SUB()                                    | 支持         | -                                                                                                                        |
| 13 | DATEDIFF()                                    | 支持         | -                                                                                                                        |
| 14 | DAY()                                         | 支持         | -                                                                                                                        |
| 15 | DAYNAME()                                     | 支持         | -                                                                                                                        |
| 16 | DAYOFMONTH()                                  | 支持         | -                                                                                                                        |
| 17 | DAYOFWEEK()                                   | 支持         | -                                                                                                                        |

| 序号 | MySQL数据库                            | GaussDB数据库 | 差异                                                                                                                |
|----|-------------------------------------|------------|-------------------------------------------------------------------------------------------------------------------|
| 18 | DAYOFYEAR()                         | 支持         | -                                                                                                                 |
| 19 | EXTRACT()                           | 支持         | -                                                                                                                 |
| 20 | FROM_DAYS()                         | 支持         | -                                                                                                                 |
| 21 | FROM_UNIXTIME()                     | 支持         | -                                                                                                                 |
| 22 | GET_FORMAT()                        | 支持         | -                                                                                                                 |
| 23 | HOUR()                              | 支持         | -                                                                                                                 |
| 24 | LAST_DAY()                          | 支持         | -                                                                                                                 |
| 25 | LOCALTIME()/<br>LOCALTIME           | 支持,<br>有差异 | MySQL入参整型值会按照一字节最大值255整数回绕（例SELECT LOCALTIME(257) == SELECT LOCALTIME(1)）。<br>GaussDB只支持[0,6]合法值，其他值报错。           |
| 26 | LOCALTIMESTAMP/<br>LOCALTIMESTAMP() | 支持,<br>有差异 | MySQL入参整型值会按照一字节最大值255整数回绕（例SELECT LOCALTIMESTAMP(257) == SELECT LOCALTIMESTAMP(1)）。<br>GaussDB只支持[0,6]合法值，其他值报错。 |
| 27 | MAKEDATE()                          | 支持         | -                                                                                                                 |
| 28 | MAKETIME()                          | 支持         | -                                                                                                                 |
| 29 | MICROSECOND()                       | 支持         | -                                                                                                                 |
| 30 | MINUTE()                            | 支持         | -                                                                                                                 |
| 31 | MONTH()                             | 支持         | -                                                                                                                 |
| 32 | MONTHNAME()                         | 支持         | -                                                                                                                 |
| 33 | NOW()                               | 支持,<br>有差异 | MySQL入参整型值会按照一字节最大值255整数回绕（例SELECT NOW(257)==SELECT NOW(1)）。<br>GaussDB只支持[0,6]合法值，其他值报错。                         |

| 序号 | MySQL数据库     | GaussDB数据库 | 差异                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|----|--------------|------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 34 | PERIOD_ADD() | 支持，有差异     | <ol style="list-style-type: none"><li>1. 整数溢出处理的行为。<br/>MySQL在5.7版本，此函数入参结果最大值为<math>2^{32}=4294967296</math>，在入参或结果的period对应的月份累加值以及month_number超过uint32范围时存在整数回绕问题；在MySQL8.0中已修复此问题，对越界月份报错。GaussDB下此函数的表现与MySQL8.0版本保持一致。</li><li>2. 负数period的表现。<br/>MySQL在5.7版本，会将负数年份解析为异常值而不是报错。GaussDB入参或结果（如100年1月减去10000月）出现负数时报错。在MySQL8.0中已修复此问题，对越界月份报错。GaussDB下此函数的表现与MySQL8.0版本保持一致。</li><li>3. period月份越界的表现。<br/>MySQL在5.7版本中，若月份大于12或等于0，例如200013、199900，会将其顺延到之后的年份，或者将0月作为上一年12月处理。在MySQL8.0中已修复此问题，对越界月份报错。GaussDB下此函数的表现与MySQL8.0版本保持一致。</li></ol> |

| 序号 | MySQL数据库      | GaussDB数据库 | 差异                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|----|---------------|------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 35 | PERIOD_DIFF() | 支持，有差异     | <ol style="list-style-type: none"> <li>1. 整数溢出处理的行为。<br/>MySQL在5.7版本，此函数入参结果最大值为<math>2^{32}=4294967296</math>，在入参或结果的period对应的月份累加值以及month_number超过uint32范围时存在整数回绕问题；在MySQL8.0中已修复此问题，对越界月份报错。GaussDB下此函数的表现与MySQL8.0版本保持一致。</li> <li>2. 负数period的表现。<br/>MySQL在5.7版本，会将负数年份解析为异常值而不是报错。GaussDB入参或结果（如100年1月减去10000月）出现负数时报错。在MySQL8.0中已修复此问题，对越界月份报错。GaussDB下此函数的表现与MySQL8.0版本保持一致。</li> <li>3. period月份越界的表现。<br/>MySQL在5.7版本中，若月份大于12或等于0，例如200013、199900，会将其顺延到之后的年份，或者将0月作为上一年12月处理。在MySQL8.0中已修复此问题，对越界月份报错。GaussDB下此函数的表现与MySQL8.0版本保持一致。</li> </ol> |
| 36 | QUARTER()     | 支持         | -                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| 37 | SEC_TO_TIME() | 支持         | -                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| 38 | SECOND()      | 支持         | -                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| 39 | STR_TO_DATE() | 支持         | 返回值与MySQL有差异，GaussDB返回的是text，MySQL返回的是datetime、date。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| 40 | SUBDATE()     | 支持         | -                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| 41 | SUBTIME()     | 支持         | -                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| 42 | SYSDATE()     | 支持，有差异     | MySQL入参整型值会按照一字节最大值255整数回绕。GaussDB不回绕。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| 43 | TIME()        | 支持         | -                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| 44 | TIME_FORMAT() | 支持         | -                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| 45 | TIME_TO_SEC() | 支持         | -                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| 46 | TIMEDIFF()    | 支持         | -                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |

| 序号 | MySQL数据库         | GaussDB数据库 | 差异                                                                                                                                 |
|----|------------------|------------|------------------------------------------------------------------------------------------------------------------------------------|
| 47 | TIMESTAMP()      | 支持         | -                                                                                                                                  |
| 48 | TIMESTAMPADD()   | 支持         | -                                                                                                                                  |
| 49 | TIMESTAMPDIFF()  | 支持         | -                                                                                                                                  |
| 50 | TO_DAYS()        | 支持         | -                                                                                                                                  |
| 51 | TO_SECONDS()     | 支持         | -                                                                                                                                  |
| 52 | UNIX_TIMESTAMP() | 支持         | MySQL会根据入参是否存在小数位，决定返回定点型还是整型。当前GaussDB在内层嵌套操作符或函数时，返回的类型与MySQL可能存在不同。当内层节点返回定点、浮点、字符型、时间类型（不包括DATE类型）时，MySQL可能返回整型，GaussDB会返回定点型。 |
| 53 | UTC_DATE()       | 支持         | -                                                                                                                                  |
| 54 | UTC_TIME()       | 支持，有差异     | MySQL入参整型值会按照一字节最大值255整数回绕，GaussDB只支持[0,6]合法值，其他值报错。                                                                               |
| 55 | UTC_TIMESTAMP()  | 支持，有差异     | MySQL入参整型值会按照一字节最大值255整数回绕，GaussDB只支持[0,6]合法值，其他值报错。                                                                               |
| 56 | WEEK()           | 支持         | -                                                                                                                                  |
| 57 | WEEKDAY()        | 支持         | -                                                                                                                                  |
| 58 | WEEKOFYEAR()     | 支持         | -                                                                                                                                  |
| 59 | YEAR()           | 支持         | -                                                                                                                                  |
| 60 | YEARWEEK()       | 支持         | -                                                                                                                                  |

### 8.2.3.3.3 字符串函数

表 8-57 字符串函数列表

| 序号 | MySQL数据库      | GaussDB数据库 | 差异                                                     |
|----|---------------|------------|--------------------------------------------------------|
| 1  | ASCII()       | 支持。        | -                                                      |
| 2  | BIT_LENGTH()  | 支持。        | -                                                      |
| 3  | CHAR_LENGTH() | 支持，有差异。    | GaussDB此函数如果数据库字符集是SQL_ASCII，CHAR_LENGTH()会返回字节数而非字符数。 |

| 序号 | MySQL数据库           | GaussDB数据库 | 差异                                                                                                                                                                                                                                                                                                                                            |
|----|--------------------|------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 4  | CHARACTER_LENGTH() | 支持，有差异。    | GaussDB此函数如果数据库字符集是SQL_ASCII，CHARACTER_LENGTH()会返回字节数而非字符数。                                                                                                                                                                                                                                                                                   |
| 5  | CONCAT()           | 支持。        | 当MySQL返回值类型为二进制字符串类型（BINARY、VARBINARY、BLOB等）时，GaussDB对应的返回值类型为LONGBLOB；当MySQL返回值类型为非二进制字符串类型（CHAR、VARCHAR、TEXT等）时，GaussDB对应的返回值类型为TEXT。                                                                                                                                                                                                       |
| 6  | CONCAT_WS()        | 支持。        | 当MySQL返回值类型为二进制字符串类型（BINARY、VARBINARY、BLOB等）时，GaussDB对应的返回值类型为LONGBLOB；当MySQL返回值类型为非二进制字符串类型（CHAR、VARCHAR、TEXT等）时，GaussDB对应的返回值类型为TEXT。                                                                                                                                                                                                       |
| 7  | HEX()              | 支持。        | -                                                                                                                                                                                                                                                                                                                                             |
| 8  | LENGTH()           | 支持。        | -                                                                                                                                                                                                                                                                                                                                             |
| 9  | LPAD()             | 支持，有差异。    | <ul style="list-style-type: none"> <li>MySQL默认最大填充长度为1398101，GaussDB默认最大长度为1048576。在不同字符集下，最大填充长度会有差异，例如字符集为'GBK'时，GaussDB默认最大长度为2097152。</li> <li>如果数据库字符集是SQL_ASCII，可能产生未预期的结果。</li> <li>当MySQL返回值类型为二进制字符串类型（BINARY、VARBINARY、BLOB等）时，GaussDB对应的返回值类型为LONGBLOB；当MySQL返回值类型为非二进制字符串类型（CHAR、VARCHAR、TEXT等）时，GaussDB对应的返回值类型为TEXT。</li> </ul> |
| 10 | REPEAT()           | 支持。        | 当MySQL返回值类型为二进制字符串类型（BINARY、VARBINARY、BLOB等）时，GaussDB对应的返回值类型为LONGBLOB；当MySQL返回值类型为非二进制字符串类型（CHAR、VARCHAR、TEXT等）时，GaussDB对应的返回值类型为TEXT。                                                                                                                                                                                                       |



| 序号 | MySQL数据库          | GaussDB数据库 | 差异                                                                                                                                                                                                                                                                                                                                            |
|----|-------------------|------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 11 | REPLACE()         | 支持。        | 当MySQL返回值类型为二进制字符串类型（BINARY、VARBINARY、BLOB等）时，GaussDB对应的返回值类型为LONGBLOB；当MySQL返回值类型为非二进制字符串类型（CHAR、VARCHAR、TEXT等）时，GaussDB对应的返回值类型为TEXT。                                                                                                                                                                                                       |
| 12 | RPAD()            | 支持，有差异。    | <ul style="list-style-type: none"> <li>MySQL默认最大填充长度为1398101，GaussDB默认最大长度为1048576。在不同字符集下，最大填充长度会有差异，例如字符集为'GBK'时，GaussDB默认最大长度为2097152。</li> <li>如果数据库字符集是SQL_ASCII，可能产生未预期的结果。</li> <li>当MySQL返回值类型为二进制字符串类型（BINARY、VARBINARY、BLOB等）时，GaussDB对应的返回值类型为LONGBLOB；当MySQL返回值类型为非二进制字符串类型（CHAR、VARCHAR、TEXT等）时，GaussDB对应的返回值类型为TEXT。</li> </ul> |
| 13 | SPACE()           | 支持。        | -                                                                                                                                                                                                                                                                                                                                             |
| 14 | STRCMP()          | 支持，有差异。    | 如果数据库字符集是SQL_ASCII，可能产生未预期的结果。                                                                                                                                                                                                                                                                                                                |
| 15 | FIND_IN_SET()     | 支持，有差异。    | <p>当指定数据库使用的字符编码是SQL_ASCII时，服务器把字节值0~127根据ASCII标准解释，而字节值128~255则当作无法解析的字符；如果该函数的输入输出包含了任何非ASCII数据，数据库将无法帮助你转换或者校验非ASCII字符。</p> <p>当MySQL返回值类型为二进制字符串类型（BINARY、VARBINARY、BLOB等）时，GaussDB对应的返回值类型为LONGBLOB；当MySQL返回值类型为非二进制字符串类型（CHAR、VARCHAR、TEXT等）时，GaussDB对应的返回值类型为TEXT。</p>                                                                 |
| 16 | LCASE()           |            |                                                                                                                                                                                                                                                                                                                                               |
| 17 | LEFT()            |            |                                                                                                                                                                                                                                                                                                                                               |
| 18 | LOWER()           |            |                                                                                                                                                                                                                                                                                                                                               |
| 19 | LTRIM()           |            |                                                                                                                                                                                                                                                                                                                                               |
| 20 | REVERSE()         |            |                                                                                                                                                                                                                                                                                                                                               |
| 21 | RIGHT()           |            |                                                                                                                                                                                                                                                                                                                                               |
| 22 | RTRIM()           |            |                                                                                                                                                                                                                                                                                                                                               |
| 23 | SUBSTR()          |            |                                                                                                                                                                                                                                                                                                                                               |
| 24 | SUBSTRING()       |            |                                                                                                                                                                                                                                                                                                                                               |
| 25 | SUBSTRING_INDEX() |            |                                                                                                                                                                                                                                                                                                                                               |
| 26 | TRIM()            |            |                                                                                                                                                                                                                                                                                                                                               |
| 27 | UCASE()           |            |                                                                                                                                                                                                                                                                                                                                               |

| 序号 | MySQL数据库 | GaussDB数据库 | 差异                                                                             |
|----|----------|------------|--------------------------------------------------------------------------------|
| 28 | UPPER()  |            |                                                                                |
| 29 | UNHEX()  | 支持。        | MySQL的返回值类型为BINARY、VARBINARY、BLOB、MEDIUMBLOB、LONGBLOB，GaussDB返回值类型固定为LONGBLOB。 |
| 30 | FIELD()  | 支持。        | -                                                                              |

### 8.2.3.3.4 强制转换函数

表 8-58 强制转换函数列表

| 序号 | MySQL数据库  | GaussDB数据库 | 差异                                                                                                                                                                                                                                                                                    |
|----|-----------|------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1  | CAST()    | 支持         | <ul style="list-style-type: none"> <li>GaussDB不支持使用CAST(expr AS CHAR[(N)] charset_info或者CAST(expr AS NCHAR[(N)])转换字符集。</li> <li>GaussDB不支持使用CAST(expr AS JSON)将表达式转换为JSON。</li> <li>GaussDB支持使用CAST(expr AS FLOAT[(p)])或CAST(expr AS DOUBLE)将表达式转换为浮点类型，MySQL 5.7版本不支持此转换。</li> </ul> |
| 2  | CONVERT() | 支持         | <ul style="list-style-type: none"> <li>GaussDB不支持使用CONVERT(expr, CHAR[(N)] charset_info或者CAST(expr, NCHAR[(N)])转换字符集。</li> <li>GaussDB不支持使用CAST(expr, JSON)将表达式转换为JSON。</li> <li>GaussDB支持使用CONVERT(expr, FLOAT[(p)])或CONVERT(expr, DOUBLE)将表达式转换为浮点类型，MySQL 5.7版本不支持此转换。</li> </ul>  |

### 8.2.3.3.5 加密函数

表 8-59 加密函数列表

| 序号 | MySQL数据库      | GaussDB数据库 | 差异                                                                                                                                                                                                                                                              |
|----|---------------|------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1  | AES_DECRYPT() | 支持         | 1. ecb为不安全加密模式，GaussDB不支持，默认为cbc模式。<br>2. GaussDB中，当指定数据库使用的字符编码是SQL_ASCII时，服务器把字节值0-127根据ASCII标准解释，而字节值128-255则当作无法解析的字符；如果该函数的输入输出包含了任何非ASCII数据，数据库将无法帮助你转换或者校验非ASCII字符。<br>3. MySQL的返回值类型为BINARY、VARBINARY、BLOB、MEDIUMBLOB、LONGBLOB，GaussDB返回值类型固定为LONGBLOB。 |
| 2  | AES_ENCRYPT() | 支持         |                                                                                                                                                                                                                                                                 |
| 3  | SHA()/SHA1()  | 支持         | -                                                                                                                                                                                                                                                               |
| 4  | SHA2()        | 支持         | -                                                                                                                                                                                                                                                               |

### 8.2.3.3.6 比较函数

表 8-60 比较函数列表

| 序号 | MySQL数据库   | GaussDB数据库 | 差异                                                                                                     |
|----|------------|------------|--------------------------------------------------------------------------------------------------------|
| 1  | COALESCE() | 支持，有差异。    | union distinct场景下，返回值精度与MySQL不完全一致。<br>当第一个不为NULL的参数的后续参数表达式中存在隐式类型转换错误时，MySQL会忽略该错误，GaussDB会提示类型转换错误。 |
| 2  | INTERVAL() | 支持。        | -                                                                                                      |

| 序号 | MySQL数据库   | GaussDB数据库 | 差异                                                                                                                                                                                                                                                        |
|----|------------|------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 3  | GREATEST() | 支持，有差异。    | <p>当MySQL返回值类型为二进制字符串类型（BINARY、VARBINARY、BLOB等）时，GaussDB对应的返回值类型为LONGBLOB；当MySQL返回值类型为非二进制字符串类型（CHAR、VARCHAR、TEXT等）时，GaussDB对应的返回值类型为TEXT。</p> <p>当该函数入参含有NULL且在WHERE关键字之后调用，返回结果与MySQL5.7不一致，此处为MySQL5.7存在的问题，MySQL8.0修复了该问题，目前GaussDB和MySQL8.0保持一致。</p> |
| 4  | LEAST()    | 支持，有差异。    | <p>当MySQL返回值类型为二进制字符串类型（BINARY、VARBINARY、BLOB等）时，GaussDB对应的返回值类型为LONGBLOB；当MySQL返回值类型为非二进制字符串类型（CHAR、VARCHAR、TEXT等）时，GaussDB对应的返回值类型为TEXT。</p> <p>当该函数入参含有NULL且在WHERE关键字之后调用，返回结果与MySQL5.7不一致，此处为MySQL5.7存在的问题，MySQL8.0修复了该问题，目前GaussDB和MySQL8.0保持一致。</p> |
| 5  | ISNULL()   | 支持。        | -                                                                                                                                                                                                                                                         |

### 8.2.3.3.7 聚合函数

表 8-61 聚合函数列表

| 序号 | MySQL数据库       | GaussDB数据库 | 差异                                                                                                                                                                                                                                                                                                                                                                              |
|----|----------------|------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1  | AVG()          | 支持，有差异     | <ul style="list-style-type: none"> <li>GaussDB中指定DISTINCT且SQL语句包含GROUP BY子句时，不保证结果的顺序。</li> <li>GaussDB中当expr中的列为BIT、BOOL、整数类型，且所有行的和超过BIGINT的范围时，会发生溢出导致整数翻转。</li> </ul>                                                                                                                                                                                                       |
| 2  | BIT_AND()      | 支持         | -                                                                                                                                                                                                                                                                                                                                                                               |
| 3  | BIT_OR()       | 支持         | -                                                                                                                                                                                                                                                                                                                                                                               |
| 4  | BIT_XOR()      | 支持         | -                                                                                                                                                                                                                                                                                                                                                                               |
| 5  | COUNT()        | 支持，有差异     | GaussDB中指定DISTINCT且SQL语句包含GROUP BY子句时，不保证结果的顺序。                                                                                                                                                                                                                                                                                                                                 |
| 6  | GROUP_CONCAT() | 支持，有差异     | <ul style="list-style-type: none"> <li>GaussDB中指定DISTINCT且SQL语句包含GROUP BY子句时，不保证结果的顺序。</li> <li>GaussDB中当GROUP_CONCAT参数中同时有DISTINCT和ORDER BY语法时，所有ORDER BY后的表达式必须也在DISTINCT的表达式之中。</li> <li>GaussDB中GROUP_CONCAT(... ORDER BY 数字)不代表按照第几个参数的顺序，数字只是一个常量表达式，相当于不排序。</li> <li>GaussDB中使用参数group_concat_max_len限制GROUP_CONCAT最大返回长度，超长截断，目前能返回的最大长度是1073741823，小于MySQL。</li> </ul> |
| 7  | MAX()          | 支持，有差异     | GaussDB中指定DISTINCT且SQL语句包含GROUP BY子句时，不保证结果的顺序。                                                                                                                                                                                                                                                                                                                                 |

| 序号 | MySQL数据库 | GaussDB数据库 | 差异                                                                                                                                                                         |
|----|----------|------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 8  | MIN()    | 支持，有差异     | GaussDB中指定DINCTINCT且SQL语句包含GROUP BY子句时，不保证结果的顺序。                                                                                                                           |
| 9  | SUM()    | 支持，有差异     | <ul style="list-style-type: none"> <li>GaussDB中指定DINCTINCT且SQL语句包含GROUP BY子句时，不保证结果的顺序。</li> <li>GaussDB中当expr中的列为BIT、BOOL、整数类型，且所有行的和超过BIGINT的范围时，会发生溢出导致整数翻转。</li> </ul> |

### 8.2.3.3.8 数字操作函数

表 8-62 数字操作函数列表

| 序号 | MySQL数据库  | GaussDB数据库 | 差异                                                                   |
|----|-----------|------------|----------------------------------------------------------------------|
| 1  | ABS()     | 支持。        | -                                                                    |
| 2  | ACOS()    | 支持。        | -                                                                    |
| 3  | ASIN()    | 支持。        | -                                                                    |
| 4  | ATAN()    | 支持。        | -                                                                    |
| 5  | ATAN2()   | 支持。        | -                                                                    |
| 6  | CEILING() | 支持。        | -                                                                    |
| 7  | COS()     | 支持。        | -                                                                    |
| 8  | DEGREES() | 支持。        | -                                                                    |
| 9  | EXP()     | 支持。        | -                                                                    |
| 10 | FLOOR()   | 支持，有差异。    | FLOOR函数的返回值类型与MySQL的有差异：入参类型为INT，GaussDB返回值类型为BIGINT，MySQL返回值类型为INT。 |
| 11 | LN()      | 支持。        | -                                                                    |
| 12 | LOG()     | 支持。        | -                                                                    |
| 13 | LOG10()   | 支持。        | -                                                                    |
| 14 | LOG2()    | 支持。        | -                                                                    |

| 序号 | MySQL数据库   | GaussDB数据库 | 差异                                                                             |
|----|------------|------------|--------------------------------------------------------------------------------|
| 15 | PI()       | 支持。        | PI函数的返回值精度与MySQL的有差异：MySQL中PI函数的结果仅保留四舍五入之后的小数后6位，而GaussDB的结果会保留四舍五入之后的小数后15位。 |
| 16 | POW()      | 支持。        | -                                                                              |
| 17 | POWER()    | 支持。        | -                                                                              |
| 18 | RAND()     | 支持。        | -                                                                              |
| 19 | SIGN()     | 支持。        | -                                                                              |
| 20 | SIN()      | 支持。        | -                                                                              |
| 21 | SQRT()     | 支持。        | -                                                                              |
| 22 | TAN()      | 支持。        | -                                                                              |
| 23 | TRUNCATE() | 支持。        | -                                                                              |

### 8.2.3.3.9 其他函数

表 8-63 其他函数列表

| 序号 | MySQL数据库     | GaussDB数据库 | 差异 |
|----|--------------|------------|----|
| 1  | DATABASE()   | 支持         | -  |
| 2  | UUID()       | 支持         | -  |
| 3  | UUID_SHORT() | 支持         | -  |

### 8.2.3.4 操作符

GaussDB数据库兼容绝大多数MySQL的操作符，但存在部分差异。如未列出，操作符行为默认为GaussDB原生行为，目前存在MySQL不支持但是GaussDB支持的语句，不建议使用这类语句。

#### 操作符差异

- ORDER BY排序对NULL值处理的差异。MySQL在排序时会把NULL值排在前面；GaussDB默认将NULL值默认排在最后面。GaussDB可以通过nulls first和nulls last设置NULL值排序顺序。
- 有ORDER BY时，GaussDB输出顺序与MySQL一致。没有ORDER BY时，GaussDB不保证结果有序。
- MySQL操作符要使用括号来保证表达的结合性与优先级无误，否则执行报错。例如：SELECT 1 regexp ('12345' regexp '123');

GaussDB操作符可以接受不用括号严格表述的表达式，也能成功执行。

- NULL值显示不同。MySQL会将NULL显示为“NULL”；GaussDB将NULL值显示为空值。

MySQL输出结果：

```
mysql> Select NULL;
+-----+
| NULL |
+-----+
| NULL |
+-----+
1 row in set (0.00 sec)
```

GaussDB输出结果：

```
m_db=# select NULL;
?column?
-----
(1 row)
```

- 操作符执行后，列名显示不一致。MySQL会将NULL显示为“NULL”；GaussDB将NULL值显示为空值。
- 字符串转double遇到非法字符串时，告警信息不一致。MySQL在常量非法字符串报错，字段非法字符串不报错；GaussDB在常量非法字符串和字段非法字符串都报错。
- 比较操作符返回结果显示不同。MySQL返回1/0；GaussDB返回t/f。

表 8-64 操作符

| 序号 | MySQL数据库 | GaussDB数据库 | 差异                                        |
|----|----------|------------|-------------------------------------------|
| 1  | <>       | 支持，但有差异。   | MySQL支持索引，GaussDB不支持索引。                   |
| 2  | <=>      | 支持，但有差异。   | MySQL支持索引，GaussDB不支持索引、hash连接和合并连接。       |
| 3  | 行表达式     | 不支持。       | MySQL支持<=>，GaussDB不支持<=>。                 |
| 4  | --       | 支持。        | MySQL表示对一个操作数进行两次取反，结果等于原操作数；GaussDB表示注释。 |



| 序号 | MySQL数据库 | GaussDB数据库 | 差异                                                                                                                                                                                                                                                                                           |
|----|----------|------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 5  | !!       | 支持，但有差异。   | <p>MySQL: !!含义同!, 表示取非。</p> <p>GaussDB: !表示取非操作, 当!与!中间存在空格时, 表示连续两次取非 (!!); 当!与!中间没有空格时, 表示阶乘 (!! )。</p> <p><b>说明</b></p> <ul style="list-style-type: none"> <li>GaussDB中, 当同时使用阶乘 (!! )和取非 (!) 时, 阶乘 (!! )和取非 (!) 中间需要添加空格, 否则会报错。</li> <li>GaussDB中, 当需要多次取非操作时, !与!之间需使用空格隔开。</li> </ul> |

| 序号 | MySQL数据库     | GaussDB数据库 | 差异                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|----|--------------|------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 6  | [NOT] REGEXP | 支持，但有差异。   | <ul style="list-style-type: none"> <li>• GaussDB和MySQL在正则表达式中支持的元字符有所不同。例如，GaussDB支持“\d”表示数字，“\w”表示字母、数字和下划线，“\s”表示空格，而MySQL不支持这些元字符，MySQL会把这些字符当成正常字符串。</li> <li>• GaussDB '\b' 可以与'\b'匹配，MySQL匹配失败。</li> <li>• GaussDB新框架下中使用“\”表示转义字符，而MySQL中使用“\\”。</li> <li>• MySQL不支持2个操作符连在一起使用。</li> <li>• 模式字符串pat非法入参，只存在右单括号')'时，GaussDB数据库报错。MySQL存在bug，后续版本已经修复此问题。</li> <li>• 在de abc匹配序列de或abc的匹配规则，当 左右存在空值时，MySQL存在bug，会报错，后续版本已经修复此问题。</li> <li>• 空白字符[\t]正则匹配字符类[:blank:]，GaussDB可匹配，MySQL\t不能匹配[:blank:]，MySQL存在bug，后续版本已经修复此问题。</li> <li>• GaussDB支持非贪婪模式匹配，即尽可能少的匹配字符，在部分特殊字符后加 '?' 问号字符，例如：“??, *?, +?, {n}?, {n}?, {n,m}?”。MySQL 5.7版本不支持非贪婪模式匹配，并报错：Got error 'repetition-operator</li> </ul> |

| 序号 | MySQL数据库             | GaussDB数据库 | 差异                                                                                                                                                                                       |
|----|----------------------|------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|    |                      |            | <p>operand invalid' from regexp。MySQL 8.0版本已经支持。</p> <ul style="list-style-type: none"> <li>在binary字符集下，text类型、blob类型均会转换成bytea类型，由于REGEXP操作符不支持bytea类型，因此无法匹配。</li> </ul>               |
| 7  | LIKE                 | 支持，但有差异。   | <p>MySQL: LIKE的左操作数只能是位运算或者算术运算或者由括号组成的表达式，LIKE的右操作数只能是单目运算符(不含NOT)或者括号组成的表达式。</p> <p>GaussDB: LIKE的左右操作数可以是任意表达式。</p>                                                                   |
| 8  | [NOT]<br>BETWEEN AND | 支持，但有差异。   | <p>MySQL: [NOT] BETWEEN AND嵌套使用时从右到左结合。[NOT] BETWEEN AND的第1个操作数和第2个操作数只能是位运算或者算术运算或者由括号组成的表达式。</p> <p>GaussDB: [NOT] BETWEEN AND嵌套使用时从左到右结合。[NOT] BETWEEN AND的第1个操作数和第2个操作数可以是任意表达式。</p> |
| 9  | IN                   | 支持，但有差异。   | <p>MySQL: IN的左操作数只能是位运算或者算术运算或者由括号组成的表达式。</p> <p>GaussDB: IN的左操作数可以是任意表达式。</p>                                                                                                           |
| 10 | !                    | 支持，但有差异。   | <p>MySQL: !的操作数只能是单目运算符(不含not)或者括号组成的表达式。</p> <p>GaussDB: !的操作数可以是任意表达式。</p>                                                                                                             |
| 11 | #                    | 不支持。       | <p>MySQL支持#注释，GaussDB不支持#注释。</p>                                                                                                                                                         |

表 8-65 操作符组合存在差异

| 操作符组合示例                                   | MySQL 数据库 | GaussDB 数据库 | 说明                                       |
|-------------------------------------------|-----------|-------------|------------------------------------------|
| SELECT 1 LIKE 3 & 1;                      | 不支持       | 支持          | LIKE的右操作数不能是位运算符组成的表达式。                  |
| SELECT 1 LIKE 1 +1;                       | 不支持       | 支持          | LIKE的右操作数不能是算术运算符组成的表达式。                 |
| SELECT 1 LIKE NOT 0;                      | 不支持       | 支持          | LIKE的右操作数只能是+、-、!等单目操作符或者括号组成的表达式，NOT除外。 |
| SELECT 1 BETWEEN 1 AND 2 BETWEEN 2 AND 3; | 从右到左结合    | 从左到右结合      | 建议加上括号明确优先级。                             |
| SELECT 2 BETWEEN 1=1 AND 3;               | 不支持       | 支持          | BETWEEN的第2个操作数不能是比较操作符组成的表达式。            |
| SELECT 0 LIKE 0 BETWEEN 1 AND 2;          | 不支持       | 支持          | BETWEEN的第1个操作数不能是模式匹配操作符组成的表达式。          |
| SELECT 1 IN (1) BETWEEN 0 AND 3;          | 不支持       | 支持          | BETWEEN的第1个操作数不能是IN操作符组成的表达式。            |
| SELECT 1 IN (1) IN (1);                   | 不支持       | 支持          | 第2个IN表达式左操作数不能是IN组成的表达式。                 |
| SELECT ! NOT 1;                           | 不支持       | 支持          | !的操作数只能是+、-、!等单目操作符或者括号组成的表达式，NOT除外。     |

### 说明

在GaussDB中支持，但在MySQL中不支持的操作符组合不建议使用。建议按照MySQL中的使用规则对操作符进行组合使用。

## 索引差异

- GaussDB当前仅支持UBTree和B-tree索引。
- 针对模糊匹配（LIKE操作符），MySQL创建默认索引可以走索引；GaussDB默认的索引不走索引，需要用户使用以下语法指定opclass，比如指定为text\_pattern\_ops，LIKE操作符才可以走索引。  
CREATE INDEX indexname ON tablename(col [opclass]);
- B-tree/UBTree索引场景保持原生GaussDB原有逻辑，即同一操作符族内的类型比较，支持索引扫描，其余索引类型暂未支持。
- where子句中，索引字段类型和常量类型操作场景下，GaussDB中索引与MySQL索引支持存在差异，如下表所示。例如以下语句GaussDB不支持索引：  
create table t(\_int int);  
create index idx on t(\_int) using BTREE;  
select \* from t where \_int > 2.0;

 说明

where子句里索引字段类型和常量类型操作场景中，可以使用cast函数将常数类型显示转为字段类型，以便实现索引。

```
select * from t where _int > cast(2.0 as signed);
```

表 8-66 索引支持存在差异

| 索引字段类型   | 常量类型                      | GaussDB | MySQL |
|----------|---------------------------|---------|-------|
| 整型       | 整型                        | 是       | 是     |
| 浮点型      | 浮点型                       | 是       | 是     |
| 定点型      | 定点型                       | 是       | 是     |
| 字符串类型    | 字符串类型                     | 是       | 是     |
| 二进制类型    | 二进制类型                     | 是       | 是     |
| 带日期的时间类型 | 带日期的时间类型                  | 是       | 是     |
| TIME类型   | TIME类型                    | 是       | 是     |
| 带日期的时间类型 | 可转为带日期的时间类型（如20231130等整型） | 是       | 是     |
| 带日期的时间类型 | TIME类型                    | 是       | 是     |
| TIME类型   | 可转为TIME类型的常量（如203008等整型）  | 是       | 是     |
| 浮点型      | 整型                        | 是       | 是     |
| 浮点型      | 定点型                       | 是       | 是     |
| 浮点型      | 字符串类型                     | 是       | 是     |
| 浮点型      | 二进制类型                     | 是       | 是     |
| 浮点型      | 带日期的时间类型                  | 是       | 是     |
| 浮点型      | TIME类型                    | 是       | 是     |
| 定点型      | 整型                        | 是       | 是     |
| 字符串类型    | 带日期的时间类型                  | 是       | 否     |
| 字符串类型    | TIME类型                    | 是       | 否     |
| 二进制类型    | 字符串类型                     | 是       | 是     |

| 索引字段类型   | 常量类型              | GaussDB | MySQL |
|----------|-------------------|---------|-------|
| 二进制类型    | 带日期的时间类型          | 是       | 否     |
| 二进制类型    | TIME类型            | 是       | 否     |
| 整型       | 浮点型               | 否       | 是     |
| 整型       | 定点型               | 否       | 是     |
| 整型       | 字符串类型             | 否       | 是     |
| 整型       | 二进制类型             | 否       | 是     |
| 整型       | 带日期的时间类型          | 否       | 是     |
| 整型       | TIME类型            | 否       | 是     |
| 定点型      | 浮点型               | 否       | 是     |
| 定点型      | 字符串类型             | 否       | 是     |
| 定点型      | 二进制类型             | 否       | 是     |
| 定点型      | 带日期的时间类型          | 否       | 是     |
| 定点型      | TIME类型            | 否       | 是     |
| 字符串类型    | 二进制类型             | 否       | 是     |
| 带日期的时间类型 | 整型（不可转为带日期的时间类型）  | 否       | 是     |
| 带日期的时间类型 | 浮点型（不可转为带日期的时间类型） | 否       | 是     |
| 带日期的时间类型 | 定点型（不可转为带日期的时间类型） | 否       | 是     |
| TIME类型   | 整型（不可转为TIME类型）    | 否       | 是     |
| TIME类型   | 字符串类型（不可转为TIME类型） | 否       | 是     |
| TIME类型   | 二进制类型（不可转为TIME类型） | 否       | 是     |
| TIME类型   | 带日期的时间类型          | 否       | 是     |

表 8-67 是否支持走索引

| 索引字段类型   | 常量类型                | 是否走索引 | MySQL |
|----------|---------------------|-------|-------|
| 字符串类型    | 整型                  | 否     | 否     |
| 字符串类型    | 浮点型                 | 否     | 否     |
| 字符串类型    | 定点型                 | 否     | 否     |
| 二进制类型    | 整型                  | 否     | 否     |
| 二进制类型    | 浮点型                 | 否     | 否     |
| 二进制类型    | 定点型                 | 否     | 否     |
| 带日期的时间类型 | 字符串类型（不可转为带日期的时间类型） | 否     | 否     |
| 带日期的时间类型 | 二进制类型（不可转为带日期的时间类型） | 否     | 否     |
| TIME类型   | 浮点型（不可转为TIME类型）     | 否     | 否     |
| TIME类型   | 定点型（不可转为TIME类型）     | 否     | 否     |

### 8.2.3.5 字符集

GaussDB数据库支持指定数据库、模式、表或列的字符集，支持的范围如下。

表 8-68 字符集列表

| 序号 | MySQL数据库 | GaussDB数据库 |
|----|----------|------------|
| 1  | utf8mb4  | 支持         |
| 2  | utf8     | 支持         |
| 3  | gbk      | 支持         |
| 4  | gb18030  | 支持         |
| 5  | binary   | 支持         |

**说明**

- utf8和utf8mb4在GaussDB中为同一个字符集，编码最大长度为4字节。由此可导致如下差异：当前字符串字符集为utf8，指定其字符序为utf8mb4\_bin/utf8mb4\_general\_ci/utf8mb4\_unicode\_ci/utf8mb4\_0900\_ai\_ci时（例如select \_utf8'a' collate utf8mb4\_bin），MySQL报错，GaussDB不报错。当字符串字符集为utf8mb4，指定其字符序为utf8\_bin/utf8\_general\_ci/utf8\_unicode\_ci时也存在同样差异。
- 词法语法解析按照字节流解析，当多字节字符中包含与'\', '\', '\\等符号一致的编码时，会导致与mysql行为不一致，建议暂时关闭转义符开关进行规避。

**8.2.3.6 排序规则**

GaussDB数据库支持指定模式、表或列的排序规则，支持的范围如下。

**说明**

排序规则差异说明：

- 当前仅有字符串类型、部分二进制类型支持指定排序规则，其他类型不支持指定排序规则，可以通过查询pg\_type系统表中类型的typcollation属性不为0来判断该类型支持字符序。MySQL中所有类型可以指定字符序，但除字符串、二进制类型其他排序规则无实际意义。
- 当前排序规则（除binary外）仅支持在其对应字符集与库级字符集一致时可以指定，GaussDB数据库中，字符集必须与数据库的字符集一致，且不支持表内多种字符集混合使用。
- utf8mb4字符集下默认字符序为utf8mb4\_general\_ci，与MySQL5.7保持一致。

**表 8-69** 排序规则列表

| 序号 | MySQL数据库           | GaussDB数据库 |
|----|--------------------|------------|
| 1  | utf8mb4_general_ci | 支持         |
| 2  | utf8mb4_unicode_ci | 支持         |
| 3  | utf8mb4_bin        | 支持         |
| 4  | gbk_chinese_ci     | 支持         |
| 5  | gbk_bin            | 支持         |
| 6  | gb18030_chinese_ci | 支持         |
| 7  | gb18030_bin        | 支持         |
| 8  | binary             | 支持         |
| 9  | utf8mb4_0900_ai_ci | 支持         |
| 10 | utf8_general_ci    | 支持         |
| 11 | utf8_bin           | 支持         |
| 12 | utf8_unicode_ci    | 支持         |



### 8.2.3.7 事务

GaussDB数据库兼容MySQL的事务，但存在部分差异。本章节介绍GaussDB的M-Compatibility数据库中事务相关的差异。

#### 事务默认隔离级别

M-Compatibility默认隔离级别为READ COMMITTED，MySQL默认隔离级别为REPEATABLE-READ。

```
-- 查看当前事务隔离级别。  
m_db=# SHOW transaction_isolation;
```

#### 子事务

M-Compatibility中，通过SAVEPOINT用于在当前事务里建立一个新的保存点（子事务），使用ROLLBACK TO SAVEPOINT回滚到一个保存点（子事务），子事务回滚后父事务可以继续运行，子事务的回滚不影响父事务的事务状态。

MySQL不存在创建保存点（子事务）。

#### 嵌套事务

嵌套事务指在事务块中开启新事务。

M-Compatibility中，正常事务块中开启新事务会警告存在一个进行中的事务，忽略开启命令；异常事务块中开启新事务将报错，必须在执行ROLLBACK/COMMIT之后才可以执行，执行ROLLBACK/COMMIT会回滚之前语句。

MySQL中，正常事务块中开启新事务会先把之前事务提交，然后开启新事务；异常事务块中开启新事务会忽略错误，提交之前无错误的语句并开启新事务。

```
-- M-Compatibility正常事务块中，开启新事务会警告并忽略。  
m_db=# DROP TABLE IF EXISTS test_t;  
m_db=# CREATE TABLE test_t(a int, b int);  
m_db=# BEGIN;  
m_db=# INSERT INTO test_t values(1, 2);  
m_db=# BEGIN; -- 会警告there is already a transaction in progress。  
m_db=# SELECT * FROM test_t ORDER BY 1;  
m_db=# COMMIT;  
  
-- M-Compatibility异常事务块中，开启新事务会报错，必须ROLLBACK/COMMIT之后才可以执行。  
m_db=# BEGIN;  
m_db=# ERROR sql; -- 错误语句。  
m_db=# BEGIN; -- 报错。  
m_db=# COMMIT; -- ROLLBACK/COMMIT之后才可以执行。
```

#### 隐式提交的语句

M-Compatibility使用GaussDB存储，继承GaussDB事务机制，事务中执行DDL、DCL不会自动提交。

MySQL在DDL、DCL、管理类语句，锁相关语句会自动提交。

```
-- M-Compatibility创建表和设置GUC参数可以回滚掉。  
m_db=# DROP TABLE IF EXISTS test_table_rollback;  
m_db=# BEGIN;  
m_db=# CREATE TABLE test_table_rollback(a int, b int);  
m_db=# \d test_table_rollback;  
m_db=# ROLLBACK;  
m_db=# \d test_table_rollback; -- 不存在该表。
```

## SET TRANSACTION 差异

M-Compatibility中，SET TRANSACTION同时设置多次隔离级别/事务访问模式时，只有最后一个会生效；多个事务特性支持使用空格和逗号分割。

MySQL中SET TRANSACTION不允许设置多次隔离级别/事务访问模式；多个事务特性只支持使用逗号分割。

表 8-70 SET TRANSACTION 差异

| 序号 | 语法                      | 功能                               | 差异                                                                         |
|----|-------------------------|----------------------------------|----------------------------------------------------------------------------|
| 1  | SET TRANSACTION         | 设置事务特性。                          | M-Compatibility中，SET TRANSACTION在会话级别生效；MySQL中SET TRANSACTION在下一个事务生效。     |
| 2  | SET SESSION TRANSACTION | 设置会话级事务特性。                       | -                                                                          |
| 3  | SET GLOBAL TRANSACTION  | 设置全局会话级事务特性，该特性适用于后续会话，对当前会话无影响。 | M-Compatibility中，GLOBAL是全局会话级别生效，只针对当前数据库实例，其它数据库不影响。<br>MySQL中，会使所有数据库生效。 |

```
-- SET TRANSACTION会话级生效。
m_db=# SET TRANSACTION ISOLATION LEVEL READ COMMITTED READ WRITE;
m_db=# SHOW transaction_isolation;
m_db=# SHOW transaction_read_only;
-- M-Compatibility同时设置多次隔离级别/事务访问模式，最后一个生效。
m_db=# SET SESSION TRANSACTION ISOLATION LEVEL READ COMMITTED, ISOLATION LEVEL
REPEATABLE READ, READ WRITE, READ ONLY;
m_db=# SHOW transaction_isolation; -- repeatable read
m_db=# SHOW transaction_read_only; -- on
```

## START TRANSACTION 差异

M-Compatibility中，START TRANSACTION开启事务时，同时支持设置隔离级别；同时设置多次隔离级别/事务访问模式时，只有最后一个会生效；当前版本不支持立即开启一致性快照；多个事务特性支持空格和逗号分割。

MySQL的start transaction 开启事务时，不支持设置隔离级别，不支持设置多次事务访问模式；多个事务特性只支持逗号分割。

```
-- 开启事务设置隔离级别。
m_db=# START TRANSACTION ISOLATION LEVEL READ COMMITTED;
m_db=# COMMIT;
-- 多次设置访问模式。
m_db=# START TRANSACTION READ ONLY, READ WRITE;
m_db=# COMMIT;
```

## 事务相关的 GUC 参数

表 8-71 事务相关的 GUC 参数差异

| 序号 | GUC参数                         | 功能                                                      | 差异                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|----|-------------------------------|---------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1  | autocommit                    | 设置事务自动提交模式。                                             | M-Compatibility中当前版本不支持SET autocommit = off。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| 2  | transaction_isolation         | 在M-Compatibility中是设置当前事务的隔离级别。<br>在MySQL中是设置会话级事务的隔离级别。 | <ol style="list-style-type: none"> <li>在M-Compatibility中，通过使用SET命令，只能改变当前事务的隔离级别。如果想要改变会话级的隔离级别，可以使用 default_transaction_isolation。<br/>在MySQL中，通过使用SET命令，可以改变会话级的事务隔离级别。</li> <li>支持范围差异，M-Compatibility中当前版本支持以下隔离级别设置，对大小写和空格敏感： <ul style="list-style-type: none"> <li>• READ COMMITTED</li> <li>• READ UNCOMMITTED</li> <li>• REPEATABLE READ</li> <li>• SERIALIZABLE</li> <li>• DEFAULT（设置和会话中默认隔离级别一样）</li> </ul> MySQL中支持以下隔离级别设置，对大小写不敏感，对空格敏感： <ul style="list-style-type: none"> <li>• READ-UNCOMMITTED</li> <li>• READ-COMMITTED</li> <li>• REPEATABLE-READ</li> <li>• SERIALIZABLE</li> </ul> </li> <li>在GaussDB中，新事务的 transaction_isolation值将被初始化为 default_transaction_isolation的值。</li> </ol> |
| 3  | tx_isolation                  | 设置事务的隔离级别；<br>tx_isolation和 transaction_isolation是同义词。  | M-Compatibility中当前不支持。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| 4  | default_transaction_isolation | 设置事务的隔离级别。                                              | M-Compatibility中通过SET设置会改变会话级事务隔离级别。<br>MySQL中不支持该系统参数。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |

| 序号 | GUC参数                         | 功能                                                     | 差异                                                                                                                                                                                                                                                                 |
|----|-------------------------------|--------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 5  | transaction_read_only         | 设置事务的访问模式。                                             | <ol style="list-style-type: none"> <li>在M-Compatibility中，通过使用SET命令，只能改变当前事务的访问模式。如果想要改变会话级的访问模式，可以使用default_transaction_read_only。在MySQL中，通过使用SET命令，可以改变会话级的事务隔离级别。</li> <li>在GaussDB中，新事务的transaction_read_only值将被初始化为default_transaction_read_only的值。</li> </ol> |
| 6  | tx_read_only                  | 设置事务的访问模式。tx_read_only和transaction_read_only是 synonym。 | M-Compatibility中当前不支持。                                                                                                                                                                                                                                             |
| 7  | default_transaction_read_only | 设置事务的访问模式。                                             | M-Compatibility中通过SET设置会改变会话级事务访问模式；MySQL中不支持该系统参数。                                                                                                                                                                                                                |

### 8.2.3.8 SQL

GaussDB数据库兼容绝大多数MySQL语法，但存在部分差异。本章节介绍GaussDB数据库当前支持的MySQL语法。

部分关键字在MySQL可以做标识符但M-Compatibility不可以或存在限制，以下为**限制做标识符列表**

表 8-72 限制做标识符列表

| 关键字类型         | 关键字                                                                                    | 约束                    |
|---------------|----------------------------------------------------------------------------------------|-----------------------|
| 保留（可以是类型或函数）  | COLLATION、COMPACT                                                                      | 除函数和变量，不可以作为其他数据库标识符。 |
| 非保留（不能是类型或函数） | BIT、BOOLEAN、COALESCE、DATE、NATIONAL、NCHAR、NONE、NUMBER、TEXT、TIME、TIMESTAMP、TIMESTAMPDIFF | 不可以作为函数或变量的标识符。       |

| 关键字类型 | 关键字                                                                  | 约束             |
|-------|----------------------------------------------------------------------|----------------|
| 保留    | ANY、ARRAY、BUCKETS、DO、END、LESS、MODIFY、OFFSET、ONLY、RETURNING、SOME、USER | 不可以作为任意数据库标识符。 |

### 8.2.3.8.1 关键字

约束差异：

- 当关键字在M-Compatibility模式下为保留关键字，在MySQL中为非保留关键字，其差异为：在M-Compatibility模式下不可作为表名、列名、列别名、AS列别名、AS表别名、表别名、函数名和变量名，在MySQL中支持。
- 当关键字在M-Compatibility模式下为非保留关键字，在MySQL中为保留关键字，其差异为：在M-Compatibility模式下可作为表名、列名、列别名、AS列别名、AS表别名、表别名、函数名和变量名，在MySQL中不支持。
- 当关键字在M-Compatibility模式下为保留关键字（可以是函数或类型），在MySQL中为保留关键字，其差异为：在M-Compatibility模式下可作为列别名、AS列别名、函数名和变量名，在MySQL中不支持。
- 当关键字在M-Compatibility模式下为保留关键字（可以是函数或类型），在MySQL中为非保留关键字，其差异为：在M-Compatibility模式下不可作为表名、列名、AS表别名和表别名，在MySQL中支持。
- 当关键字在M-Compatibility模式下为非保留关键字（不能是函数或类型），在MySQL中为保留关键字，其差异为：在M-Compatibility模式下可作为表名、列名、列别名、AS列别名、AS表别名、表别名和变量名，在MySQL中不支持。
- 当关键字在M-Compatibility模式下为非保留关键字（不能是函数或类型），在MySQL中为非保留关键字，其差异为：在M-Compatibility模式下不可作为函数名，在MySQL中支持。

#### 📖 说明

在M-Compatibility模式下的非保留关键字、保留关键字（可以是函数或类型）以及非保留关键字（不能是函数或类型）之中，以下关键字不能作为列别名进行使用：

BETWEEN、BIGINT、BLOB、CHAR、CHARACTER、CROSS、DEC、DECIMAL、DIV、DOUBLE、EXISTS、FLOAT、FLOAT4、FLOAT8、GROUPING、INNER、INOUT、INT、INT1、INT2、INT3、INT4、INT8、INTEGER、JOIN、LEFT、LIKE、LONGBLOB、LONGTEXT、MEDIUMBLOB、MEDIUMINT、MEDIUMTEXT、MOD、NATURAL、NUMERIC、OUT、OUTER、PRECISION、REAL、RIGHT、ROW、ROW\_NUMBER、SIGNED、SMALLINT、SOUNDS、TINYBLOB、TINYINT、TINYTEXT、VALUES、VARCHAR、VARYING、WITHOUT

其中，SIGNED和WITHOUT在MySQL中可以作为列别名进行使用。

### 8.2.3.8.2 标识符

M-Compatibility模式下标识符存在以下差异：

- GaussDB无引号标识符中不支持以美元符号（\$）开头，MySQL无引号标识符中支持。

- GaussDB无引号标识符中的支持大小写敏感的数据库对象。
- GaussDB标识符支持U+0080~U+00FF扩展字符，MySQL标识符支持U+0080~U+FFFF的扩展字符。
- 无引号标识符中，GaussDB不支持创建以数字开头包含一个e或E结尾作为标识符的表，例如：

```
-- GaussDB报错不支持，MySQL支持
m_db=# CREATE TABLE 23e(c1 int);
ERROR: syntax error at or near "23"
LINE 1: CREATE TABLE 23e(c1 int);
                        ^

m_db=# CREATE TABLE t1(23E int);
ERROR: syntax error at or near "23"
LINE 1: CREATE TABLE t1(23E int);
                        ^
```

- 有引号标识符中，GaussDB对于创建了列名为纯数字或科学计算法的表，不支持直接使用，需要在引号中使用；对于点操作符(.)场景，列名为纯数字或科学计算法的表也需要在引号中使用。例如：

```
-- 创建列名为纯数字或科学计算法的表
m_db=# CREATE TABLE t1(`123` int, `1e3` int, `1e` int);
CREATE TABLE

-- 向表中插入数据
m_db=# INSERT INTO t1 VALUES(7, 8, 9);
INSERT 0 1

-- 结果非预期，但与MySQL结果一致
m_db=# SELECT 123 FROM t1;
?column?
-----
    123
(1 row)

-- 结果非预期，但与MySQL结果一致
m_db=# SELECT 1e3 FROM t1;
?column?
-----
    1000
(1 row)

-- 结果非预期，并且与MySQL结果不一致
m_db=# SELECT 1e FROM t1;
e
---
    1
(1 row)

-- 正确用法
m_db=# SELECT `123` FROM t1;
123
-----
    7
(1 row)

m_db=# SELECT `1e3` FROM t1;
1e3
-----
    8
(1 row)

m_db=# SELECT `1e` FROM t1;
1e
---
    9
(1 row)

-- 点操作符的场景，GaussDB不支持，MySQL支持
```

```

m_db=# SELECT t1.123 FROM t1;
ERROR: syntax error at or near ".123"
LINE 1: SELECT t1.123 FROM t1;
      ^
m_db=# SELECT t1.1e3 FROM t1;
ERROR: syntax error at or near "1e3"
LINE 1: SELECT t1.1e3 FROM t1;
      ^
m_db=# SELECT t1.1e FROM t1;
ERROR: syntax error at or near "1"
LINE 1: SELECT t1.1e FROM t1;
      ^
-- 点操作符的场景，正确用法：
m_db=# SELECT t1.`123` FROM t1;
 123
-----
 7
(1 row)

m_db=# SELECT t1.`1e3` FROM t1;
 1e3
-----
 8
(1 row)

m_db=# SELECT t1.`1e` FROM t1;
 1e
-----
 9
(1 row)

m_db=# DROP TABLE t1;
DROP TABLE
    
```

- GaussDB分区名使用双引号（需要设置SQL\_MODE为ANSI\_QUOTES）或反引号是区分大小写的，MySQL不区分。

### 8.2.3.8.3 DDL

| 概述                          | 详细语法说明                   | 差异                                                                                                                                                                                                       |
|-----------------------------|--------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 建表和修改表时支持创建主键、UNIQUE索引、外键约束 | ALTER TABLE、CREATE TABLE | <ul style="list-style-type: none"> <li>• 在GaussDB中，当约束关联的表为ustore，且SQL语句中指定为using btree时，底层会建立为ubtree。</li> <li>• 在GaussDB中，允许将外键作为分区键。</li> <li>• 索引名、约束名、key名 GaussDB是SCHEMA下唯一，MySQL是表下唯一。</li> </ul> |

| 概述    | 详细语法说明                   | 差异                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|-------|--------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 支持自增列 | ALTER TABLE、CREATE TABLE | <ul style="list-style-type: none"> <li>● GaussDB的自动增长列建议为索引的第一个字段，否则建表时产生警告，MySQL自动增长列必须为索引第一个字段，否则建表时会报错。GaussDB含有自动增长列的表进行某些操作时会产生错误，例如：ALTER TABLE EXCHANGE PARTITION。</li> <li>● GaussDB的 AUTO_INCREMENT = value语法，value必须为小于<math>2^{127}</math>的正数。MySQL可以为0，GaussDB不可以。</li> <li>● GaussDB中当自增值已经达到字段数据类型的最大值时，继续自增将产生错误。MySQL有些场景产生错误或警告，有些场景仍自增为最大值。</li> <li>● 不支持 innodb_autoinc_lock_mode系统变量，GaussDB的 GUC参数 auto_increment_cache=0时，批量插入自动增长列的行为与MySQL系统变量 innodb_autoinc_lock_mode=1相似。</li> <li>● GaussDB的自动增长列在导入数据或者进行Batch Insert执行计划的插入操作时，对于混合0、NULL和确定值的场景，如果产生错误，后续插入自增值不一定与MySQL完全一致。             <ul style="list-style-type: none"> <li>- 提供 auto_increment_cache 参数，可以控制预留自增值的数量。</li> </ul> </li> <li>● GaussDB的并行导入或插入自动增长列触发自增时，每个并行线程预留的缓存值也只在其线程中使用，未完全使用完毕的话，也会出现表中自动增长列的值不连续的情况。并行插入产生的自增</li> </ul> |



| 概述 | 详细语法说明 | 差异                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|----|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|    |        | <p>值结果无法保证与MySQL完全一致。</p> <ul style="list-style-type: none"> <li>● GaussDB的本地临时表中的自动增长列批量插入时不会预留自增值，正常场景不会产生不连续的自增值。MySQL临时表与普通表中的自动增长列自增结果一致。</li> <li>● GaussDB的SERIAL数据类型为原有的自增列，与AUTO_INCREMENT自增列有差异。MySQL的SERIAL数据类型就是AUTO_INCREMENT自增列。</li> <li>● GaussDB的不允许auto_increment_offset的值大于auto_increment_increment的值，会产生错误。MySQL允许，并说明auto_increment_offset会被忽略。</li> <li>● 在表有主键或索引的情况下，ALTER TABLE命令重写表数据的顺序与MySQL不一定相同，GaussDB按表数据存储顺序重写，MySQL会按主键或索引顺序重写，导致自增值的顺序可能不同。</li> <li>● GaussDB的ALTER TABLE命令添加或修改自增列时，第一次预留自增值的数量是表统计信息中的行数，统计信息的行数不一定与MySQL一致。</li> <li>● GaussDB的last_insert_id函数返回值为128位的整型。</li> <li>● GaussDB在触发器或用户自定义函数中自增时，刷新last_insert_id返回值。MySQL不刷新。</li> <li>● GaussDB的对GUC参数auto_increment_offset和auto_increment_increment</li> </ul> |

| 概述                       | 详细语法说明                                              | 差异                                                                                                                                                                                                                                 |
|--------------------------|-----------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                          |                                                     | 设置超出范围的值会产生错误。MySQL会自动改为边界值。                                                                                                                                                                                                       |
| 支持前缀索引                   | CREATE INDEX、ALTER TABLE、CREATE TABLE               | <ul style="list-style-type: none"> <li>GaussDB中前缀长度不得超过2676，键值的实际长度受内部页面限制，若字段中含有多字节字符或者一个索引上有多个键，索引行长度可能会超限报错。</li> <li>GaussDB中主键索引中不支持前缀键，创建或添加主键时不支持指定前缀长度。</li> </ul>                                                         |
| 支持指定字符集与排序规则             | ALTER SCHEMA、ALTER TABLE、CREATE SCHEMA、CREATE TABLE | <ul style="list-style-type: none"> <li>指定库级字符集时，除BINARY字符集外，暂不支持创建新库/模式的字符集与数据库的server_encoding不同。</li> <li>指定表级、列级字符集和字符序时，MySQL支持指定与库级字符集、字符序不同的字符集和字符序。在GaussDB中，表级、列级字符集和字符序仅支持BINARY字符集、字符序或者与库级字符集、字符序相同的字符集、字符序。</li> </ul> |
| 修改表时支持在表第一列前面或者在指定列后面添加列 | ALTER TABLE                                         | -                                                                                                                                                                                                                                  |
| 修改列名称/定义语法兼容             | ALTER TABLE                                         | 暂不支持DROP INDEX   DROP KEY   ORDER BY子项。                                                                                                                                                                                            |
| 创建分区表语法兼容                | CREATE TABLE PARTITION、CREATE TABLE SUBPARTITION    | <ul style="list-style-type: none"> <li>对于partition_key，分区策略的分区键仅支持1列，且不支持表达式。</li> <li>对于subpartition_key，分区策略的分区键仅支持1列，且不支持表达式。</li> <li>分区表不支持用生成列做分区键。</li> </ul>                                                               |
| 建表和修改表时支持指定表级和列级comment  | CREATE TABLE、ALTER TABLE                            | -                                                                                                                                                                                                                                  |
| 创建索引时支持指定索引级comment      | CREATE INDEX                                        | -                                                                                                                                                                                                                                  |

| 概述                    | 详细语法说明                       | 差异                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|-----------------------|------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>交换普通表和分区表分区的数据</p> | <p>ALTER TABLE PARTITION</p> | <p>ALTER TABLE EXCHANGE PARTITION的差异点:</p> <ul style="list-style-type: none"> <li>• 对于自增列, MySQL执行 alter exchange partition 后, 自增列会被重置; GaussDB 则不会被重置, 自增列则按照旧的自增值递增。</li> <li>• MySQL表或分区使用 tablespace时, 则无法进行分区和普通表数据的交换; GaussDB表或分区使用不同的tablespace时, 仍可进行分区和普通表数据的交换。</li> <li>• 对于列默认值, MySQL不会校验默认值, 因此默认值不同时也可进行分区和普通表数据的交换; GaussDB会校验默认值, 如果默认值不同, 则无法进行分区和普通表数据的交换。</li> <li>• MySQL在分区表或普通表上进行DROP列操作后, 表结构仍然一致, 则可进行分区和普通表数据的交换; GaussDB需要保证普通表和分区表的被删除列严格对齐才能进行分区和普通表数据的交换。</li> <li>• MySQL和GaussDB的哈希算法不同, 所以两者在相同的hash分区存储的数据可能不一致, 导致最后交换的数据也可能不一致。</li> <li>• MySQL的分区表不支持外键, 普通表包含外键或其他表引用普通表的外键, 则无法进行分区和普通表数据的交换; GaussDB的分区表支持外键, 在两个表的外键约束一致时, 则可进行分区和普通表数据的交换, GaussDB的分区表不带外键, 普通表有其他表引用, 如果分区表和普通表表一致, 则可进行分区和普通表数据的交换。</li> </ul> |

| 概述                          | 详细语法说明                                                                                                                                                                                                                                                                                             | 差异                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|-----------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 修改分区表的分区键信息                 | ALTER TABLE                                                                                                                                                                                                                                                                                        | MySQL支持修改分区表的分区键信息，GaussDB中不支持。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| 支持CREATE TABLE ... LIKE语法兼容 | CREATE TABLE ... LIKE                                                                                                                                                                                                                                                                              | <ul style="list-style-type: none"> <li>在MySQL 8.0.16 之前的版本中，CHECK约束会被语法解析但功能会被忽略，表现为不复制CHECK约束，GaussDB支持复制CHECK约束。</li> <li>对于主键约束名称，在建表时，MySQL所有主键约束名称固定为PRIMARY KEY，GaussDB不支持复制。</li> <li>对于唯一键约束名称，在建表时，MySQL支持复制，GaussDB不支持复制。</li> <li>对于CHECK约束名称，在建表时，MySQL 8.0.16 之前的版本无CHECK约束信息，GaussDB支持复制。</li> <li>对于索引名称，在建表时，MySQL支持复制，GaussDB不支持复制。</li> <li>在跨sql_mode模式建表时，MySQL受宽松模式和严格模式控制，GaussDB可能存在严格模式失效的情况。<br/>例如：源表存在默认值“0000-00-00”，在“no_zero_date”严格模式下，GaussDB建表成功，且包含默认值“0000-00-00”，严格模式失效；而MySQL建表失败，受严格模式控制。</li> </ul> |
| 支持增加子分区语法兼容                 | <pre>ALTER TABLE [ IF EXISTS ] { table_name [*]   ONLY table_name   ONLY ( table_name )} add_clause; add_clause: ADD {{partition_less_than_item   partition_start_end_item   partition_list_item}   PARTITION({partition_less_than _item   partition_start_end_item   partition_list_item})}</pre> | <p>保留原分区表语法。</p> <p>不支持下述语法添加多分区：<br/>ALTER TABLE table_name ADD PARTITION (partition_definition1, partition_definition1,...);</p> <p>仅支持原有添加多分区语法：<br/>ALTER TABLE table_name ADD PARTITION (partition_definition1), ADD PARTITION (partition_definition2[y1] ), ...;</p>                                                                                                                                                                                                                                                                 |

| 概述                      | 详细语法说明                                                                                                                                                                                                  | 差异                                                                                                                                                                                                                                                                                         |
|-------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| TRUNCATE子分区语法兼容         | ALTER TABLE [ IF EXISTS ]<br>table_name<br>truncate_clause;                                                                                                                                             | 支持子项有差异，对于truncate_clause: <ul style="list-style-type: none"> <li>• M-Compatibility模式:<br/>TRUNCATE PARTITION { { ALL   partition_name [, ...] }   FOR ( partition_value [, ...] ) } [ UPDATE GLOBAL INDEX ]</li> <li>• MySQL支持:<br/>TRUNCATE PARTITION {partition_names   ALL}</li> </ul> |
| 主键索引名                   | CREATE TABLE table_name<br>( col_definitive ,PRIMARY KEY [index_name] [ USING method ] ( { column_name   ( expression ) } [ ASC   DESC ] } [, ... ] ) index_parameters [USING method  COMMENT 'string'] | GaussDB中的主键指定索引名后创建的索引名为用户所指定的索引名，MySQL索引名为PRIMARY。                                                                                                                                                                                                                                        |
| 删除有依赖的对象                | DROP drop_type name<br>CASCADE;                                                                                                                                                                         | 在GaussDB中，删除有依赖的对象需要加CASCADE，MySQL不需要。                                                                                                                                                                                                                                                     |
| NOT NULL约束不允许插入NULL值    | CREATE TABLE t1(id int NOT NULL DEFAULT 8);<br>INSERT INTO t1 VALUES(NULL);<br>INSERT INTO t1 VALUES(1), (NULL),(2);                                                                                    | 在MySQL宽松模式下，会将NULL进行类型转换，并成功插入数据；在MySQL严格模式下不允许插入NULL值。在GaussDB不支持此特性，在宽松模式和严格模式下均不允许插入NULL值。                                                                                                                                                                                              |
| CHECK约束生效               | CREATE TABLE                                                                                                                                                                                            | CREATE TABLE带CHECK约束的时候，MySQL8.0会生效，MySQL5.7只解析语法但不生效。GaussDB在此功能上同步MySQL8.0版本，且GaussDB CHECK约束可以引用其他列，而MySQL不能。<br>GaussDB 一个表中最多只能加32767个CHECK约束。                                                                                                                                        |
| 索引的algorithm和lock选项不起作用 | CREATE INDEX ...<br>DROP INDEX ...                                                                                                                                                                      | M-Compatibility模式的CREATE/DROP INDEX语句中INDEX选项algorithm_option和lock_option目前只在语法上支持，创建时不报错，但实际不起作用。                                                                                                                                                                                         |

| 概述                                   | 详细语法说明                                                                         | 差异                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|--------------------------------------|--------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| CREATE TABLE hash 分区和二级分区的存储与MySQL不同 | CREATE TABLE                                                                   | GaussDB的CREATE TABLE语句中hash分区表和二级分区表所使用的hash函数与MySQL不一致，因此hash分区表和二级分区表的存储与MySQL有区别。                                                                                                                                                                                                                                                                                                                                                                       |
| 分区表索引                                | CREATE INDEX                                                                   | <p>GaussDB的分区表索引分为LOCAL和GLOBAL两种。LOCAL索引与某个具体分区绑定，而GLOBAL索引则对应整个分区表。</p> <p>LOCAL和GLOBAL索引的创建方法和默认规则具体说明参见《开发者指南》中”SQL语法 &gt; SQL语句 &gt; C &gt; CREATE INDEX” 章节，例如：在非分区键上创建唯一索引，会默认创建为GLOBAL索引。</p> <p>MySQL无GLOBAL索引的概念。在GaussDB中，当分区表索引为GLOBAL索引时，对表分区进行DROP、TRUNCATE、EXCHANGE等操作不会默认更新GLOBAL索引，进而导致GLOBAL索引失效，导致后续语句无法选中该索引。为了避免这种场景，建议用户在使用分区操作语法时在最后显指定UPDATE GLOBAL INDEX子句，或配置全局GUC参数 enable_gpi_auto_update为 true（推荐），使得在进行分区操作时自定更新GLOBAL索引。</p> |
| COMMENT中不支持转义符(\)                    | <pre>CREATE TABLE ... COMMENT 'string'  CREATE INDEX ...COMMENT 'string'</pre> | GaussDB的COMMENT中不支持转义符(\)。                                                                                                                                                                                                                                                                                                                                                                                                                                 |

| 概述                                                             | 详细语法说明                           | 差异                                                                                                                                                                                                                                                                                                                                                           |
|----------------------------------------------------------------|----------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| CREATE/ALTER TABLE 语句中分区表为KEY分区，不支持指定algorithm。部分分区定义入参不支持表达式。 | CREATE TABLE、ALTER TABLE         | GaussDB的CREATE/ALTER TABLE语句中分区表为KEY分区，不支持指定algorithm。<br>不支持表达式入参的语法： <ul style="list-style-type: none"> <li>• PARTITION BY HASH()</li> <li>• PARTITION BY KEY()</li> <li>• VALUES LESS THAN()</li> </ul>                                                                                                                                                   |
| 分区表不支持LINEAR/KEY hash                                          | CREATE TABLE ... PARTITION ...   | GaussDB分区表不支持LINEAR/KEY hash。                                                                                                                                                                                                                                                                                                                                |
| check和auto_increment语法不能作用在同一字段                                | CREATE TABLE                     | 由于MySQL5.7的check字段不生效，check和auto_increment同时作用于同一字段只有auto_increment生效，但GaussDB报错。                                                                                                                                                                                                                                                                            |
| 删除存在依赖关系的表                                                     | DROP TABLE                       | GaussDB删除存在依赖的表必须加上CASCADE才能成功，MySQL不需要。                                                                                                                                                                                                                                                                                                                     |
| SET 不支持设置FOREIGN_KEY_CHECKS                                    | SET                              | GaussDB目前不支持SET FOREIGN_KEY_CHECKS=0 1。                                                                                                                                                                                                                                                                                                                      |
| 表定义相关选项                                                        | CREATE TABLE... 、ALTER TABLE ... | <ul style="list-style-type: none"> <li>• GaussDB不支持以下选项：AVG_ROW_LENGTH、CHECKSUM、COMPRESSION、CONNECTION、DATA DIRECTORY、INDEX DIRECTORY、DELAY_KEY_WRITE、ENCRYPTION、INSERT_METHOD、KEY_BLOCK_SIZE、MAX_ROWS、MIN_ROWS、PACK_KEYS、PASSWORD、STATS_AUTO_RECALC、STATS_PERSISTENT、STATS_SAMPLE_PAGES。</li> <li>• 以下选项在GaussDB中不报错，但实际上也不生效：ENGINE、ROW_FORMAT。</li> </ul> |

| 概述                                                                                 | 详细语法说明                          | 差异                               |
|------------------------------------------------------------------------------------|---------------------------------|----------------------------------|
| CMK密钥轮转，轮换加密COLUMN ENCRYPTION KEY的CLIENT MASTER KEY，对COLUMN ENCRYPTION KEY明文进行重加密。 | ALTER COLUMN ENCRYPTION KEY     | M-Compatibility模式不支持全密态。故不支持该语法。 |
| 密态等值查询特性使用多级加密模型，主密钥加密列密钥，列密钥加密数据。本语法用于创建主密钥对象。                                    | CREATE CLIENT MASTER KEY        | M-Compatibility模式不支持全密态。故不支持该语法。 |
| 创建一个列加密密钥，该密钥可用于加密表中的指定列。                                                          | CREATE COLUMN ENCRYPTION KEY    | M-Compatibility模式不支持全密态。故不支持该语法。 |
| 全密态功能，传输密钥到服务端缓存，只在开启内存解密逃生通道的情况下使用。                                               | \send_token                     | M-Compatibility模式不支持全密态。故不支持该语法。 |
| 全密态功能，传输密钥到服务端缓存，只在开启内存解密逃生通道的情况下使用。                                               | \st                             | M-Compatibility模式不支持全密态。故不支持该语法。 |
| 全密态功能，销毁服务端缓存的密钥，只在开启内存解密逃生通道的情况下使用。                                               | \clear_token                    | M-Compatibility模式不支持全密态。故不支持该语法。 |
| 全密态功能，销毁服务端缓存的密钥，只在开启内存解密逃生通道的情况下使用。                                               | \ct                             | M-Compatibility模式不支持全密态。故不支持该语法。 |
| 在全密态数据库特性中,用于设置访问外部密钥管理者的参数。                                                       | \key_info KEY_INFO              | M-Compatibility模式不支持全密态。故不支持该语法。 |
| 全密态功能，用于开启三方动态库功能与加载三方动态库时的参数设置。                                                   | \crypto_module_info MODULE_INFO | M-Compatibility模式不支持全密态。故不支持该语法。 |
| 全密态功能，用于开启三方动态库功能与加载三方动态库时的参数设置。                                                   | \cmi MODULE_INFO                | M-Compatibility模式不支持全密态。故不支持该语法。 |



| 概述                                                     | 详细语法说明                    | 差异                                                                                             |
|--------------------------------------------------------|---------------------------|------------------------------------------------------------------------------------------------|
| generated always as 语句不能再引用由 generated always as 生成的列。 | Generated Always AS       | GaussDB generated always as 语句不能再引用由 generated always as 生成的列，MySQL 可以。                        |
| ALTER TABLE 目前不支持 RENAME = 语法                          | ALTER TABLE               | GaussDB 的 ALTER TABLE 目前只支持 RENAME TO 和 RENAME AS 语法，MySQL 支持 RENAME TO、RENAME AS 和 RENAME = 。 |
| 禁用 GUC 参数 enable_expr_fusion                           | SET enable_expr_fusion=ON | M-Compatibility 模式暂不支持 GUC 参数 enable_expr_fusion 打开。                                           |

| 概述                                 | 详细语法说明                                      | 差异                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|------------------------------------|---------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>支持CREATE VIEW AS SELECT语法兼容</p> | <pre>CREATE VIEW table_name AS query;</pre> | <ul style="list-style-type: none"> <li>针对以下类型，不支持 CREATE VIEW view_name AS query语法中query包含计算操作（如函数调用、使用操作符计算）。             <ul style="list-style-type: none"> <li>- BINARY[(n)]</li> <li>- BOOLEAN/BOOL</li> <li>- VARBINARY(n)</li> <li>- CHAR[(n)]</li> <li>- VARCHAR(n)</li> <li>- TIME[(p)]</li> <li>- DATETIME[(p)]</li> <li>- TIMESTAMP[(p)]</li> <li>- BIT[(n)]</li> <li>- NUMERIC[(p[,s])]</li> <li>- DECIMAL[(p[,s])]</li> <li>- DEC[(p[,s])]</li> <li>- FIXED[(p[,s])]</li> <li>- FLOAT4[(p, s)]</li> <li>- FLOAT8[(p,s)]</li> <li>- FLOAT[(p)]</li> <li>- REAL[(p, s)]</li> <li>- FLOAT[(p, s)]</li> <li>- DOUBLE[(p,s)]</li> <li>- DOUBLE PRECISION[(p,s)]</li> <li>- TEXT</li> <li>- TINYTEXT</li> <li>- MEDIUMTEXT</li> <li>- LONGTEXT</li> <li>- BLOB</li> <li>- TINYBLOB</li> <li>- MEDIUMBLOB</li> <li>- LONGBLOB</li> </ul> </li> <li>在query为简单查询场景下，M-Compatibility模式针对上述类型的计算操作进行报错提示，例如：             <pre>m_db=# CREATE TABLE TEST (salary int(10)); CREATE TABLE m_db=# INSERT INTO TEST</pre> </li> </ul> |

| 概述       | 详细语法说明                    | 差异                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|----------|---------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|          |                           | <pre>VALUES(8000); INSERT 0 1 m_db=# CREATE VIEW view1 AS SELECT salary/10 as te FROM TEST; ERROR: Unsupported type numeric used with expression in CREATE VIEW statement. m_db=# CREATE TABLE TEST (salary int(10)); CREATE TABLE m_db=# INSERT INTO TEST VALUES(8000); INSERT 0 1 m_db=# CREATE VIEW view2 AS SELECT sec_to_time(salary) as te FROM TEST; ERROR: Unsupported type time used with expression in CREATE VIEW statement.</pre> <ul style="list-style-type: none"> <li>在query为复合查询，子查询等非简单查询场景下，M-Compatibility模式针对上述类型的计算操作与MySQL存在差异，M-Compatibility模式下新创建表的数据类型列精度属性不保留。</li> </ul> |
| 索引名可重名范围 | CREATE TABLE、CREATE INDEX | MySQL中索引名在一个表下唯一，在不同的表下可以有相同的索引名。M-Compatibility模式中的索引名在同一个SCHEMA下唯一，在同一的SCHEMA下不可用相同的索引名。在M-Compatibility模式下，针对会自动创建索引的约束和key，也会有相同的规则。                                                                                                                                                                                                                                                                                                                                                                                                                                                             |

| 概述      | 详细语法说明                  | 差异                                                                                                                                                                                                                                                                                                                                                                         |
|---------|-------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 视图依赖差异  | CREATE VIEW、ALTER TABLE | <p>MySQL中视图存储，只记录目标表的表名、列名、数据库名信息，不记录目标表的唯一标识；GaussDB会将创建视图时的SQL解析，并存储目标表的唯一标识。因此存在如下差异：</p> <ol style="list-style-type: none"> <li>1. 修改存在视图依赖的列的数据类型，MySQL中对应的视图不感知目标表的修改，因此可以修改成功；GaussDB视图中的列禁止修改数据类型，因此无法修改该列的数据类型。</li> <li>2. 重命名存在视图依赖的列，MySQL中对应的视图不感知目标表的修改，因此可以修改成功，但是后续无法查询该视图；GaussDB视图中，每个列精确存储了其对应的表和列的唯一标识，因此表中的列名可以修改成功，视图中的列名不被修改，且后续可以查询该视图。</li> </ol> |
| 外键差异    | CREATE TABLE            | <p>GaussDB外键约束对类型不敏感，如果主表和从表对应的字段数据类型存在隐式类型转换就可以建成。MySQL外键类型敏感。如果两个表对应的列类型不同外键无法建成。</p> <p>MySQL不支持通过MODIFY COLUMN或CHANGE COLUMN方式修改表列外键所在列的数据类型或列名等，GaussDB可以。</p>                                                                                                                                                                                                        |
| 索引升降序差异 | CREATE INDEX            | <p>在MySQL 5.7中，ASC   DESC被解析但是被忽略，默认行为为ASC；在MySQL 8.0及GaussDB中，ASC   DESC被解析且生效。</p>                                                                                                                                                                                                                                                                                       |

## 8.2.3.8.4 DML

| 序号 | 概述                      | 详细语法说明 | 差异                                                                                                                                                      |
|----|-------------------------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1  | DELETE支持从多个表中删除数据       | DELETE | -                                                                                                                                                       |
| 2  | DELETE支持ORDER BY和LIMIT  | DELETE | -                                                                                                                                                       |
| 3  | DELETE支持从指定分区（或子分区）删除数据 | DELETE | -                                                                                                                                                       |
| 4  | UPDATE支持从多个表中更新数据       | UPDATE | -                                                                                                                                                       |
| 5  | UPDATE支持ORDER BY和LIMIT  | UPDATE | -                                                                                                                                                       |
| 6  | SELECT INTO语法兼容         | SELECT | <ul style="list-style-type: none"><li>● GaussDB可以使用SELECT INTO根据查询结果创建一个新表，MySQL不支持。</li><li>● GaussDB的SELECT INTO语法不支持将多个查询进行集合运算后的结果作为查询结果。</li></ul> |

| 序号 | 概述               | 详细语法说明  | 差异                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|----|------------------|---------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7  | REPLACE INTO语法兼容 | REPLACE | <ul style="list-style-type: none"> <li>● 时间类型初始值的差异。例如： <ul style="list-style-type: none"> <li>- MySQL不受严格模式和宽松模式的影响，可向表中插入时间0值，即： <pre>mysql&gt; CREATE TABLE test(f1 TIMESTAMP NOT NULL, f2 DATETIME NOT NULL, f3 DATE NOT NULL); Query OK, 1 row affected (0.00 sec)  mysql&gt; REPLACE INTO test VALUES(f1, f2, f3); Query OK, 1 row affected (0.00 sec)  mysql&gt; SELECT * FROM test; +-----+-----+   f1            f2              f3                          +-----+-----+   0000-00-00 00:00:00   0000-00-00 00:00:00   0000-00-00   +-----+-----+ 1 row in set (0.00 sec)</pre> </li> <li>- GaussDB在宽松模式下才可以成功插入时间0值，即 <pre>gaussdb=# SET b_format_version = '5.7'; SET gaussdb=# SET b_format_dev_version = 's1'; SET gaussdb=# SET sql_mode = ""; SET gaussdb=# CREATE TABLE test(f1 TIMESTAMP NOT NULL, f2 DATETIME NOT NULL, f3 DATE NOT NULL); CREATE TABLE gaussdb=# REPLACE INTO test VALUES(f1, f2, f3); REPLACE 0 1 gaussdb=# SELECT * FROM test; f1            f2            f3 +-----+-----+ 0000-00-00 00:00:00   0000-00-00 00:00:00   0000-00-00 (1 row)</pre> 在严格模式下，则报错date/ time field value out of range: "0000-00-00 00:00:00"。 </li> </ul> </li> <li>● 位串类型初始值的差异。例 如：</li> </ul> |

| 序号 | 概述              | 详细语法说明 | 差异                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|----|-----------------|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|    |                 |        | <ul style="list-style-type: none"> <li>- MySQLBIT类型的初始值为空串", 长度为 ( M/8向上取整 ), 即:<br/> <pre>mysql&gt; CREATE TABLE test(f1 BIT(3) NOT NULL); Query OK, 0 rows affected (0.01 sec)  mysql&gt; REPLACE INTO test VALUES(f1); Query OK, 1 row affected (0.00 sec)  mysql&gt; SELECT f1, length(f1) FROM test; +----+-----+   f1   length(f1)   +----+-----+            1   +----+-----+ 1 row in set (0.00 sec)</pre> </li> <li>- GaussDB的BIT类型的初始值为空串", 长度为0, 则报错。<br/> <pre>gaussdb=# CREATE TABLE test(f1 BIT(3) NOT NULL); CREATE TABLE gaussdb=# REPLACE INTO test VALUES(f1); REPLACE 0 1 gaussdb=# SELECT f1, length(f1) FROM test; +----+-----+   f1   length(f1)   +----+-----+            0   +----+-----+ (1 row)</pre> </li> </ul> |
| 8  | SELECT支持指定多分区查询 | SELECT | -                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| 9  | UPDATE支持指定多分区更新 | UPDATE | -                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |

| 序号 | 概述              | 详细语法说明    | 差异                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|----|-----------------|-----------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 10 | LOAD DATA导入数据功能 | LOAD DATA | <p>在使用LOAD DATA导入数据功能时，GaussDB与MySQL相比有如下差异：</p> <ul style="list-style-type: none"> <li>● LOAD DATA语法执行结果与M*严格模式一致，宽松模式暂未适配。</li> <li>● IGNORE与LOCAL参数功能仅为当导入数据与表中数据存在冲突时，忽略当前冲突行数据功能和当文件中字段数小于指定表中列数时自动为其余列填充默认值功能，其余功能暂未适配。</li> <li>● 指定LOCAL关键字，且文件路径为相对路径时，文件从二进制目录下搜索；不指定LOCAL关键字，且文件路径为相对路径时，文件从数据目录下搜索。</li> <li>● 语法中指定分隔符，转义字符，分行符等符号时，若指定为单引号，将导致词法解析错误。</li> <li>● [(col_name_or_user_var [, col_name_or_user_var] ...)]指定列参数不支持重复指定列。</li> <li>● [FIELDS TERMINATED BY 'string']指定换行符不能与[LINES TERMINATED BY 'string']分隔符相同。</li> <li>● 执行LOAD DATA语法写入表中的数据若无法转换为表中数据类型格式时报错。</li> <li>● LOAD DATA SET表达式中不支持指定列名计算。</li> <li>● 若set表达式返回值类型与对应列类型之间不存在隐式转换函数则报错。</li> <li>● LOAD DATA只能用于表，不能用于视图。</li> <li>● Windows下的文件与Linux环境下文件默认换行符存在差异，LOAD DATA无法识别此场景会报错，建议用户导入时检查导入文件行结尾的换行符。</li> </ul> |



| 序号 | 概述                  | 详细语法说明                                                                                | 差异                                                                                                                                                                                                                                                                                       |
|----|---------------------|---------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 11 | INSERT支持VALUES引用列语法 | INSERT INTO tabname VALUES(1, 2,3) ON DUPLICAT E KEY UPDATE b = VALUES(co lumn_nam e) | GaussDB 的ON DUPLICATE KEY UPDATE子句中的VALUES()不支持表名.列名格式，MySQL支持。                                                                                                                                                                                                                          |
| 12 | LIMIT限制差异           | DELETE、SELECT、UPDATE                                                                  | <p>各个语句的limit子项与MySQL的limit项当前存在差异。</p> <p>GaussDB中limit参数最大值为BIG INT类型限制（超过9223372036854775807报错）。</p> <p>在MySQL中，limit最大值为unsigned LONGLONG类型限制（超过18446744073709551615报错）。</p> <p>limit可以设置小数值，实际执行时四舍五入。MySQL不能取小数。</p> <p>GaussDB的delete语句中，不允许limit 0。MySQL在delete语句中允许limit 0。</p> |

| 序号 | 概述                                      | 详细语法说明 | 差异                                                                                                                                                                                                                                                                                                                                                                                                      |
|----|-----------------------------------------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 13 | 反斜杠(\)用法差异                              | INSERT | <p>反斜杠(\)的用法在GaussDB和MySQL中都可以由参数控制但当前默认用法不同:</p> <p>MySQL中使用参数 NO_BACKSLASH_ESCAPES控制字符串和标识符中的反斜杠(\)被解析为普通字符还是转义字符, 默认反斜杠字符(\)作为字符串和标识符中的转义字符。设置set sql_mode='NO_BACKSLASH_ESCAPES';可以禁用反斜杠字符(\)作为字符串和标识符中的转义字符。</p> <p>GaussDB中使用参数 standard_conforming_strings控制字符串和标识符中的反斜杠\被解析为普通字符还是转义字符。默认值为on, 在普通字符串文本中按照SQL标准把反斜杠(\)当普通文本。使用set standard_conforming_strings=off;将反斜杠字符(\)作为字符串和标识符中的转义字符。</p> |
| 14 | 插入值少于字段数目时, MySQL报错, GaussDB补充空值。       | INSERT | GaussDB不指定列的列表时, 如果插入值少于字段数目, 默认按建表时的字段顺序赋值。字段上有非空约束时报错, 没有非空约束时, 如果指定了默认值则缺省部分补充默认值, 若未指定默认值则补充空。                                                                                                                                                                                                                                                                                                      |
| 15 | ORDER BY中排序的列必须包括在结果集的列中。               | SELECT | 在GaussDB中, 在与GROUP BY子句一起使用的情况下, ORDER BY中排序的列必须包括在SELECT语句所检索的结果集的列中。在与DISTINCT关键字一起使用的情况下, ORDER BY中排序的列必须包括在SELECT语句所检索的结果集的列中。                                                                                                                                                                                                                                                                      |
| 16 | 不允许对约束字段用 ON DUPLICATE KEY UPDATE 进行修改。 | INSERT | -                                                                                                                                                                                                                                                                                                                                                                                                       |
| 17 | SELECT结果允许存在重复列名。                       | SELECT | -                                                                                                                                                                                                                                                                                                                                                                                                       |

| 序号 | 概述                                            | 详细语法说明            | 差异                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|----|-----------------------------------------------|-------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 18 | NATURAL JOIN与MySQL有差异。                        | SELECT            | 在GaussDB中，NATURAL [ [LEFT   RIGHT] OUTER] JOIN允许不指定LEFT   RIGHT，不指定时NATURAL OUTER JOIN为NATURAL JOIN。允许连续使用多次JOIN。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| 19 | 外键数据类型是timestamp/datetime时，update/delete外表报错。 | UPDATE/<br>DELETE | 外键数据类型是timestamp/datetime时，update/delete外表报错，MySQL成功。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| 20 | nature join和using兼容。                          | SELECT            | <ul style="list-style-type: none"> <li>GaussDB join的顺序严格按照从左往右，MySQL可能会调整顺序。</li> <li>GaussDB和MySQL在natural join与using时均不允许左表或右表参与join的字段出现歧义（一般由左或右临时表中重名字段造成）。因为两者join的顺序有差别，故行为上可能有差别。</li> </ul> <p>- GaussDB的行为：</p> <pre>m_regression=# CREATE TABLE t1(a int,b int); CREATE TABLE m_regression=# CREATE TABLE t2(a int,b int); CREATE TABLE m_regression=# CREATE TABLE t3(a int,b int); CREATE TABLE m_regression=# SELECT * FROM t1 JOIN t2; a   b   a   b ----+----+--- (0 rows) m_regression=# SELECT * FROM t1 JOIN t2 natural join t3; -- failed, 因为:列a,b在t1 join t2 得到的临时表中存在重复, 故nature join存在歧义。 ERROR: common column name "a" appears more than once in left table</pre> <p>- MySQL的行为：</p> <pre>mysql&gt; SELECT * FROM t1 JOIN t2 NATURAL JOIN t3; Empty set (0.00 sec) mysql&gt; SELECT * FROM (t1 join t2) NATURAL JOIN t3; ERROR 1052 (23000): Column 'a' in from clause is ambiguous</pre> |

| 序号 | 概述                                               | 详细语法说明                                   | 差异                                                                                                                                                                                                                                                         |
|----|--------------------------------------------------|------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 21 | with clause兼容<br>MySQL8.0版本                      | SELECT、<br>INSERT、<br>UPDATE、<br>DELETE  | -                                                                                                                                                                                                                                                          |
| 22 | join兼容                                           | SELECT                                   | GaussDB join不支持使用逗号“,”的连接方式，MySQL支持。<br>GaussDB不支持use index for join。                                                                                                                                                                                      |
| 23 | SELECT语句中列表达式为函数表达式、算数表达式等情形时，查询结果显示的列名为?column? | SELECT                                   | GaussDB SELECT语句中列表达式为函数表达式、算数表达式等情形时，查询结果显示的列名为?column?，MySQL为对应表达式。                                                                                                                                                                                      |
| 24 | SELECT导出文件（into outfile）                         | SELECT ...<br>INTO<br>OUIFILE ...        | SELECT INTO OUTFILE语法，导出文件中FLOAT、DOUBLE、REAL类型的值显示精度和MySQL存在差异，不影响COPY导入和导入后的值。                                                                                                                                                                            |
| 25 | UPDATE/INSERT/<br>REPLACE ... SET指定模式名、表名        | UPDATE/<br>INSERT/<br>REPLACE ...<br>SET | UPDATE/REPLACE SET中，MySQL的三段式用法为database.table.column；GaussDB的三段式用法为table.column.filed，其中filed为指定复合类型中的属性。二者存在差异。<br>INSERT ... SET中，MySQL支持使用column、table.column和database.table.column；GaussDB只支持使用column，不支持使用table.column和database.table.column，二者存在差异。 |
| 26 | UPDATE SET执行顺序与MySQL不同                           | UPDATE ...<br>SET                        | MySQL中，UPDATE SET的顺序是从前往后依次UPDATE，前面UPDATE的结果会影响后面的结果，且允许多次设置同一列；GaussDB中为先取出原来的所有相关的数据，再一次性UPDATE，且不允许多次设置同一列，二者存在差异。                                                                                                                                     |
| 27 | IGNORE特性                                         | INSERT                                   | MySQL数据库和GaussDB执行过程的差异，因此产生的WARNING条数和WARNING信息可能存在不同。                                                                                                                                                                                                    |

| 序号 | 概述       | 详细语法说明 | 差异                                                                                            |
|----|----------|--------|-----------------------------------------------------------------------------------------------|
| 28 | HAVING语法 | SELECT | GaussDB的HAVING必须仅引用GROUP BY子句中的列或聚合函数中使用的列。但是，MySQL支持对此行为的扩展，并允许HAVING引用列表中的SELECT列和外部子查询中的列。 |

### 8.2.3.8.5 DCL

| 序号 | 概述                   | 详细语法说明                                                                                   | 差异                                                                                                  |
|----|----------------------|------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------|
| 1  | SET NAMES指定COLLATE字句 | SET [ SESSION   LOCAL ] NAMES { 'charset_name' [ COLLATE 'collation_name' ]   DEFAULT }; | GaussDB中暂不支持指定charset_name与数据库字符集不同。具体请参考《M-Compatibility开发指南》中“SQL参考 > SQL语法 > SQL语句 > S > SET”章节。 |

### 8.2.3.8.6 其它语句

| 序号 | 概述     | 详细语法说明       | 差异                                                                                                                          |
|----|--------|--------------|-----------------------------------------------------------------------------------------------------------------------------|
| 1  | 事务相关语法 | 数据库默认隔离级别    | M-Compatibility默认隔离级别为READ COMMITTED，MySQL默认隔离级别为REPEATABLE READ。<br>M-Compatibility隔离级别只有READ COMMITTED REPEATABLE READ生效。 |
| 2  | 事务相关语法 | 事务嵌套         | M-Compatibility中嵌套事务不会自动提交，MySQL会自动提交。                                                                                      |
| 3  | 事务相关语法 | 自动提交         | M-Compatibility使用GaussDB存储，继承GaussDB事务机制，事务中执行DDL、DCL不会自动提交。MySQL在DDL、DCL、管理类语句，锁相关语句会自动提交。                                 |
| 4  | 事务相关语法 | 报错后需rollback | M-Compatibility事务中报错，需要执行rollback，MySQL无限制。                                                                                 |
| 5  | 事务相关语法 | 锁机制          | M-Compatibility锁机制只能在事务块中使用，MySQL无限制。                                                                                       |

| 序号 | 概述  | 详细语法说明 | 差异                                                                                                                                                                                                                                                                                                               |
|----|-----|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 6  | 锁机制 | 锁机制    | <ul style="list-style-type: none"> <li>MySQL获取read锁后，当前会话无法进行写操作，M-Compatibility获取read锁后，当前会话可以进行写操作。</li> <li>MySQL给表上锁后，读取其他表报错，M-Compatibility无限制。</li> <li>MySQL同一会话中获取同一个表的锁，会自动释放上一个锁，并提交事务，M-Compatibility无该机制。</li> <li>M-Compatibility中LOCK TABLE只能在一个事务块的内部有用，且无UNLOCK TABLE命令，锁总是在事务结束时释放。</li> </ul> |

### 8.2.3.8.7 用户与权限

#### 概述

在M-Compatibility中，用户与权限管控相关的行为、语法整体沿用GaussDB的机制，暂不同步MySQL。

用户与权限的行为与GaussDB保持一致，具体行为说明请参见《开发者指南》中的“数据库安全 > 用户及权限”章节。

用户与权限的语法在原有GaussDB的基础上，裁剪了部分语法，具体语法说明请参见《M-Compatibility开发指南》中的“SQL参考 > SQL语法 > SQL语句”章节。M-Compatibility与GaussDB的语法差异请参见表8-73。

M-Compatibility创建USER时会自动创建与USER同名的Schema，MySQL不创建。

表 8-73 M-Compatibility 与 GaussDB 的语法差异

| 序号 | 语法说明         | 概述                                          | M-Compatibility与GaussDB的差异                                                                               |
|----|--------------|---------------------------------------------|----------------------------------------------------------------------------------------------------------|
| 1  | CREATE ROLE  | 创建一个角色。                                     | 在M-Compatibility中，不支持指定涉及以下关键字的选项：ENCRYPTED、UNENCRYPTED、RESOURCE POOL、PERM SPACE、TEMP SPACE、SPILL SPACE。 |
| 2  | CREATE USER  | 创建一个用户。                                     |                                                                                                          |
| 3  | CREATE GROUP | 创建一个新用户组。CREATE GROUP是CREATE ROLE的别名，不推荐使用。 |                                                                                                          |
| 4  | ALTER ROLE   | 修改角色属性。                                     |                                                                                                          |
| 5  | ALTER UER    | 修改用户属性。                                     |                                                                                                          |
| 6  | ALTER GROUP  | 修改一个用户组的属性。                                 |                                                                                                          |
|    |              |                                             | -                                                                                                        |

| 序号 | 语法说明                     | 概述                                | M-Compatibility与 GaussDB的差异                              |
|----|--------------------------|-----------------------------------|----------------------------------------------------------|
| 7  | DROP ROLE                | 删除角色。                             | -                                                        |
| 8  | DROP USER                | 删除用户。                             | -                                                        |
| 9  | DROP GROUP               | 删除用户组。                            | -                                                        |
| 10 | DROP OWNED               | 删除一个数据库角色所拥有的数据库对象。               | -                                                        |
| 11 | REASSIGN OWNED           | 修改数据库对象的属主。                       | M-Compatibility中不支持该语法。                                  |
| 12 | GRANT                    | 对角色和用户进行授权操作。                     | M-Compatibility中不支持授予或回收函数、存储过程、表空间、DATABASE LINK等对象的权限。 |
| 13 | REVOKE                   | 用于撤销一个或多个角色的权限。                   |                                                          |
| 14 | ALTER DEFAULT PRIVILEGES | 设置应用于将来创建的对象权限（这不会影响分配到已有对象中的权限）。 | M-Compatibility中不支持该语法。                                  |

## 差异说明

- 语法格式差异

M-Compatibility的授权语法请参见《M-Compatibility开发指南》中的“SQL参考 > SQL语法 > SQL语句 > G > GRANT”章节，MySQL中的授权语法如下：

```
-- 全局级、数据库级、表级、存储过程级赋权语法
GRANT
  priv_type [(column_list)]
  [, priv_type [(column_list)]] ...
  ON [object_type] priv_level
  TO user [auth_option] [, user [auth_option]] ...
  [REQUIRE {NONE | tls_option [[AND] tls_option] ...}]
  [WITH {GRANT OPTION | resource_option} ...]

-- 用户代理赋权语法
GRANT PROXY ON user
  TO user [, user] ...
  [WITH GRANT OPTION]

object_type: {
  TABLE
  | FUNCTION
  | PROCEDURE
}

priv_level: {
  *
  | *.*
  | db_name.*
  | db_name.tbl_name
  | tbl_name
  | db_name.routine_name
}
```

```

user:
  'user_name'@'host_name'

auth_option: {
  IDENTIFIED BY 'auth_string'
  | IDENTIFIED WITH auth_plugin
  | IDENTIFIED WITH auth_plugin BY 'auth_string'
  | IDENTIFIED WITH auth_plugin AS 'auth_string'
  | IDENTIFIED BY PASSWORD 'auth_string'
}

tls_option: {
  SSL
  | X509
  | CIPHER 'cipher'
  | ISSUER 'issuer'
  | SUBJECT 'subject'
}

resource_option: {
  | MAX_QUERIES_PER_HOUR count
  | MAX_UPDATES_PER_HOUR count
  | MAX_CONNECTIONS_PER_HOUR count
  | MAX_USER_CONNECTIONS count
}

```

- 赋权类型差异  
MySQL支持的赋权类型如下：

表 8-74 MySQL 支持的赋权类型

| 权限类型                           | 释义及权限级别                                                                                              |
|--------------------------------|------------------------------------------------------------------------------------------------------|
| <b>ALL [PRIVILEGES]</b>        | 授予指定访问级别的所有权限，除了 <b>GRANT OPTION</b> 和 <b>PROXY</b> 。                                                |
| <b>ALTER</b>                   | 启用 <b>ALTER TABLE</b> 。级别：全局、数据库、表。                                                                  |
| <b>ALTER ROUTINE</b>           | 允许更改或删除存储过程。级别：全局、数据库、例程。                                                                            |
| <b>CREATE</b>                  | 启用数据库和表创建。级别：全局、数据库、表。                                                                               |
| <b>CREATE ROUTINE</b>          | 启用存储过程创建。级别：全局、数据库。                                                                                  |
| <b>CREATE TABLESPACE</b>       | 允许创建、更改或删除表空间和日志文件组。级别：全局。                                                                           |
| <b>CREATE TEMPORARY TABLES</b> | 启用 <b>CREATE TEMPORARY TABLE</b> 。级别：全局、数据库。                                                         |
| <b>CREATE USER</b>             | 启用 <b>CREATE USER</b> 、 <b>DROP USER</b> 、 <b>RENAME USER</b> 和 <b>REVOKE ALL PRIVILEGES</b> 。级别：全局。 |
| <b>CREATE VIEW</b>             | 允许创建或更改视图。级别：全局、数据库、表。                                                                               |



| 权限类型                      | 释义及权限级别                                            |
|---------------------------|----------------------------------------------------|
| <b>DELETE</b>             | 启用 <b>DELETE</b> 。级别：全局、数据库、表。                     |
| <b>DROP</b>               | 允许删除数据库、表和视图。级别：全局、数据库、表。                          |
| <b>EVENT</b>              | 启用定时任务。级别：全局、数据库。                                  |
| <b>EXECUTE</b>            | 使用户能够执行存储过程。级别：全局、数据库、存储过程。                        |
| <b>FILE</b>               | 使用户能够使服务器读取或写入文件。级别：全局。                            |
| <b>GRANT OPTION</b>       | 允许向其他账户授予权限或从其他账户删除权限。级别：全局、数据库、表、存储过程、代理。         |
| <b>INDEX</b>              | 允许创建或删除索引。级别：全局、数据库、表。                             |
| <b>INSERT</b>             | 启用 <b>INSERT</b> 。级别：全局、数据库、表、列。                   |
| <b>LOCK TABLES</b>        | 在具有SELECT权限的表上启用LOCK TABLES。级别：全局、数据库。             |
| <b>PROCESS</b>            | 使用户能够通过 <b>SHOW PROCESSLIST</b> 查看所有正在运行的线程。级别：全局。 |
| <b>PROXY</b>              | 启用用户代理。级别：从用户到用户。                                  |
| <b>REFERENCES</b>         | 启用外键创建。级别：全局、数据库、表、列。                              |
| <b>RELOAD</b>             | 启用 <b>FLUSH</b> 操作的使用。级别：全局。                       |
| <b>REPLICATION CLIENT</b> | 使用户能够查询源服务器或副本服务器的位置。级别：全局。                        |
| <b>REPLICATION SLAVE</b>  | 允许副本从源读取二进制日志。级别：全局。                               |
| <b>SELECT</b>             | 启用使用 <b>SELECT</b> 。级别：全局、数据库、表、列。                 |
| <b>SHOW DATABASES</b>     | 启用 <b>SHOW DATABASES</b> 以显示所有数据库。级别：全局。           |
| <b>SHOW VIEW</b>          | 启用 <b>SHOW CREATE VIEW</b> 。级别：全局、数据库、表。           |

| 权限类型     | 释义及权限级别                                                                                  |
|----------|------------------------------------------------------------------------------------------|
| SHUTDOWN | 启用mysqladmin shutdown的使用。级别：全局。                                                          |
| SUPER    | 启用其他管理操作，例如 CHANGE MASTER TO、KILL、PURGE BINARY LOGS、SET GLOBAL和mysqladmin debug命令。级别：全局。 |
| TRIGGER  | 启用触发器操作。级别：全局、数据库、表。                                                                     |
| UPDATE   | 启用UPDATE。级别：全局、数据库、表、列。                                                                  |
| USAGE    | 等价于“没有特权”。                                                                               |

M-Compatibility以级别划分支持以下权限：

表 8-75 M-Compatibility 支持的赋权类型

| 授权对象 | 支持授予的权限                                                                                 |
|------|-----------------------------------------------------------------------------------------|
| 模式   | CREATE、USAGE、ALTER、DROP、COMMENT                                                         |
| 表、视图 | SELECT、INSERT、UPDATE、DELETE、TRUNCATE、REFERENCES、TRIGGER、ALTER、DROP、COMMENT、INDEX、VACUUM |
| 列    | SELECT、INSERT、UPDATE、REFERENCES、COMMENT                                                 |
| 序列   | SELECT、USAGE、UPDATE、ALTER、DROP、COMMENT                                                  |

- MySQL中通过'dbname.\*'表示模式层级的授权对象；在M-Compatibility中，使用'{DATABASE | SCHEMA} dbname'表示模式层级的授权对象。
- MySQL中用户名为两部分：用户名@主机名；M-Compatibility当前仅支持用户名。
- MySQL支持在GRANT赋权语法中修改用户验证，安全连接，资源参数属性，即auth\_option、tls\_option和resource option；M-Compatibility赋权语法中不支持以上特性，需使用CREATE USER、ALTER USER设置用户相关属性。
- MySQL支持用户代理赋权，GRANT PROXY ON主要用于对多个用户进行统一的权限管理。MySQL5.7未提供角色机制，而在MySQL8.0和M-Compatibility中都提供了角色机制。角色能满足用户对于多个用户权限统一管控的目标，可以替代GRANT PROXY ON。

- M-Compatibility拥有public的概念，所用用户都拥有public的权限，部分系统表、系统视图可供所有用户查询。用户可以对public所拥有的权限进行grant和revoke；MySQL中，新创建的用户只拥有全局的usage权限，这个权限很小，几乎为0，只有连接数据库和查询information\_schema 数据库的权限。
- M-Compatibility中，对象的所有者缺省具有该对象上的所有权限，出于安全考虑所有者可以舍弃部分权限，但ALTER、DROP、COMMENT、INDEX、VACUUM以及对象的可再授予权限属于所有者固有的权限，隐式拥有；MySQL中，没有owner的概念，即使用户创建了表，如果没赋予用户对应权限，那么用户也不能对其创建的表进行IUD等操作。
- 在MySQL中，USAGE实际上表示无权限，所用用户都拥有该权限，当执行revoke或grant usage时，实际上不会进行任何修改；在M-Compatibility中，USAGE权限如下：
  - 对于模式，USAGE允许访问包含在指定模式中的对象，若没有该权限，则只能看到这些对象的名称。
  - 对于序列，USAGE允许使用nextval函数。
- 在M-Compatibility中，支持给用户设置管理员角色，包括系统管理员（SYSADMIN）、安全管理员（CREATEROLE）、审计管理员（AUDITADMIN）、监控管理员（MONADMIN）、运维管理员（OPRADMIN）、安全策略管理员（POLADMIN）。默认情况下拥有SYSADMIN属性的系统管理员，具备系统最高权限。三权分立后，系统管理员将不再具有CREATEROLE属性（安全管理员）和AUDITADMIN属性（审计管理员）能力，即不再拥有创建角色和用户的权限，也不再拥有查看和维护数据库审计日志的权限；在MySQL中，不支持该用户设置管理员角色，也没有三权分立相关设计。
- 在M-Compatibility中，可以给用户赋予ANY权限，表示用户能够在非系统模式下拥有对应的权限，包括CREATE ANY TABLE、SELECT ANY TABLE、CREATE ANY INDEX等；在MySQL中，不支持ANY权限的赋予。
- MySQL中提供SHOW GRANTS查询用户权限；M-Compatibility中，可以通过gsql客户端元命令'\l+'、'\dn+'、'\dp'查询权限信息，也可以通过查询pg\_namespace、pg\_class、pg\_attribute等系统表的权限相关字段查询权限信息。
- MySQL中数据库、表、列被删除时，相关的授权信息在系统表中依然保留，如果重新创建同名对象用户依然拥有权限；M-Compatibility中当数据库、表、列被删除时，相关的授权信息会被删除，在重新创建同名对象后需要重新授权。
- MySQL在授予数据库层级的权限时，支持 '\_' 和 '%' 对数据库名进行模糊匹配；M-Compatibility不支持对象名模糊匹配， '\_' 或 '%' 等特殊字符被识别为普通字符。
- MySQL中，GRANT语句中指定用户不存在时默认会创建该账户（此特性已在MySQL8.0中移除）；M-Compatibility不支持给未创建用户赋权。

### 8.2.3.8.8 系统表和系统视图

| 序号 | 系统表或系统视图                   | 差异列                   | M-Compatibility与MySQL的差异                             |
|----|----------------------------|-----------------------|------------------------------------------------------|
| 1  | information_schema.columns | generation_expression | 该字段输出结果因涉及M-Compatibility与MySQL的表达式的字符串拼接逻辑的不同而存在差异。 |

| 序号 | 系统表或系统视图                      | 差异列             | M-Compatibility与MySQL的差异                                                                                |
|----|-------------------------------|-----------------|---------------------------------------------------------------------------------------------------------|
| 2  | information_schema.columns    | data_type       | 该字段输出结果因涉及M-Compatibility的数据类型format_type输出，目前未修改，与MySQL存在差异。                                           |
| 3  | information_schema.columns    | column_type     | 该字段输出结果因涉及M-Compatibility的数据类型format_type输出，目前未修改，与MySQL存在差异。                                           |
| 4  | information_schema.tables     | engine          | M-Compatibility中不支持该字段。                                                                                 |
| 5  | information_schema.tables     | version         | M-Compatibility中不支持该字段。                                                                                 |
| 6  | information_schema.tables     | row_format      | M-Compatibility中不支持该字段。                                                                                 |
| 7  | information_schema.tables     | avg_row_length  | M-Compatibility下表示使用数据文件除以所有元组数（包括活元组和死元组）的结果。表中没有元组，值为null。                                            |
| 8  | information_schema.tables     | max_data_length | M-Compatibility中不支持该字段。                                                                                 |
| 9  | information_schema.tables     | data_free       | M-Compatibility中表示死元组在总元组中的比例乘以数据文件大小。如果表中没有元组，则为null。                                                  |
| 10 | information_schema.tables     | check_time      | M-Compatibility中不支持该字段。                                                                                 |
| 11 | information_schema.tables     | create_time     | 在M-Compatibility下，此字段与MySQL行为表现有差异，对于创建视图的情形MySQL中该字段置null，M-Compatibility则显示实际的创建表时间。数据库自带的表，视图设置null。 |
| 12 | information_schema.tables     | update_time     | M-Compatibility数据库自带的表，视图设置null。                                                                        |
| 13 | information_schema.statistics | collation       | M-Compatibility只有值A、D，不会是NULL。                                                                          |

| 序号 | 系统表或系统视图                      | 差异列                           | M-Compatibility与MySQL的差异                                                                                                                                                          |
|----|-------------------------------|-------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 14 | information_schema.statistics | packed                        | M-Compatibility中不支持该字段。                                                                                                                                                           |
| 15 | information_schema.statistics | sub_part                      | M-Compatibility中不支持该字段。                                                                                                                                                           |
| 16 | information_schema.statistics | comment                       | M-Compatibility中不支持该字段。                                                                                                                                                           |
| 17 | information_schema.partitions | subpartition_name             | M-Compatibility中，如果分区不是子分区，则为null。                                                                                                                                                |
| 18 | information_schema.partitions | subpartition_ordinal_position | M-Compatibility中，如果分区不是子分区，则为null。                                                                                                                                                |
| 19 | information_schema.partitions | partition_method              | M-Compatibility中，分区策略。如果分区不是一级分区，则为null。<br><ul style="list-style-type: none"> <li>• 'r': 范围分区。</li> <li>• 'l': list分区。</li> <li>• 'h': hash分区。</li> <li>• 'i': 间隔分区。</li> </ul>  |
| 20 | information_schema.partitions | subpartition_method           | M-Compatibility中，子分区策略。如果分区不是二级分区，则为null。<br><ul style="list-style-type: none"> <li>• 'r': 范围分区。</li> <li>• 'l': list分区。</li> <li>• 'h': hash分区。</li> <li>• 'i': 间隔分区。</li> </ul> |
| 21 | information_schema.partitions | partition_description         | M-Compatibility中，是区分一级分区和二级分区的。                                                                                                                                                   |
| 22 | information_schema.partitions | partition_expression          | M-Compatibility中不支持该字段。                                                                                                                                                           |
| 23 | information_schema.partitions | subpartition_expression       | M-Compatibility中不支持该字段。                                                                                                                                                           |
| 24 | information_schema.partitions | data_length                   | M-Compatibility中不支持该字段。                                                                                                                                                           |
| 25 | information_schema.partitions | max_data_length               | M-Compatibility中不支持该字段。                                                                                                                                                           |

| 序号 | 系统表或系统视图                          | 差异列               | M-Compatibility与MySQL的差异 |
|----|-----------------------------------|-------------------|--------------------------|
| 26 | information_schema.<br>partitions | index_length      | M-Compatibility中不支持该字段。  |
| 27 | information_schema.<br>partitions | data_free         | M-Compatibility中不支持该字段。  |
| 28 | information_schema.<br>partitions | create_time       | M-Compatibility中不支持该字段。  |
| 29 | information_schema.<br>partitions | update_time       | M-Compatibility中不支持该字段。  |
| 30 | information_schema.<br>partitions | check_time        | M-Compatibility中不支持该字段。  |
| 31 | information_schema.<br>partitions | checksum          | M-Compatibility中不支持该字段。  |
| 32 | information_schema.<br>partitions | partition_comment | M-Compatibility中不支持该字段。  |
| 33 | information_schema.<br>partitions | nodegroup         | M-Compatibility中不支持该字段。  |

## 📖 说明

- 视图中对于整型的类型回显，不支持指定精度范围。如MySQL的bigint(1)，M-Compatibility下对应的是bigint类型，MySQL中bigint(21) unsigned，在M-Compatibility下对应的是bigint unsigned类型。
- MySQL中int类型，在M-Compatibility中是integer类型。
- MySQL中的set、enum类型的字段，M-Compatibility中都不支持。  
m\_schema.columns\_priv视图的Column\_priv字段、m\_schema.tables\_priv视图的Table\_priv、Column\_priv字段、m\_schema.procs\_priv视图的Routine\_type、Proc\_priv字段、m\_schema.func视图的type、language、sql\_data\_access、is\_deterministic、security\_type、sql\_mode字段、m\_schema.func视图的type字段均不支持，在此版本中不予显示。
- 由于information\_schema.tables、information\_schema.statistics中table\_rows、avg\_row\_length、data\_length、data\_free、index\_length、cardinality基于统计信息获取，查看前请先执行ANALYZE，更新统计信息后再查看（如果数据库中更新数据，建议延迟执行ANALYZE）。
- information\_schema.statistics包含的索引列需要是创建索引中索引列是完整的表列，如果索引列是表达式，不在这个视图中。
- information\_schema.partitions 中table\_row、avg\_row\_length基于统计信息获取，查看前请先执行ANALYZE，更新统计信息后再查看（如果数据库中更新数据，建议延迟执行ANALYZE）。
- information\_schema.partitions中一级分区和二级分区是分开呈现。
- 对于支持的grantee字段，MySQL的格式是'*user\_name*@'*host\_name*'，在M-Compatibility数据库，是被授予权限的用户或角色的名称。
- 对于支持的host字段，在M-Compatibility数据库，返回当前节点的hostname。
- m\_schema.tables\_priv、information\_schema.user\_privileges、information\_schema.schema\_privileges、information\_schema.table\_privileges、information\_schema.column\_privileges、m\_schema.columns\_priv、m\_schema.func、m\_schema.procs\_priv 在MySQL下需要授权后才能查看视图内容，M-Compatibility数据库可以根据默认权限查看到对应的内容。如对于表t1，在MySQL下需要先对t1给对应的用户授权，才能在权限视图中看到对应的权限信息，M-Compatibility数据库下则可以直接在视图中看到t1表相关的权限信息。

# 9 安全

## 9.1 责任共担

华为云秉承“将对网络和业务安全性保障的责任置于公司的商业利益之上”。针对层出不穷的云安全挑战和无孔不入的云安全威胁与攻击，华为云在遵从法律法规业界标准的基础上，以安全生态圈为护城河，依托华为独有的软硬件优势，构建面向不同区域和行业的完善云服务安全保障体系。

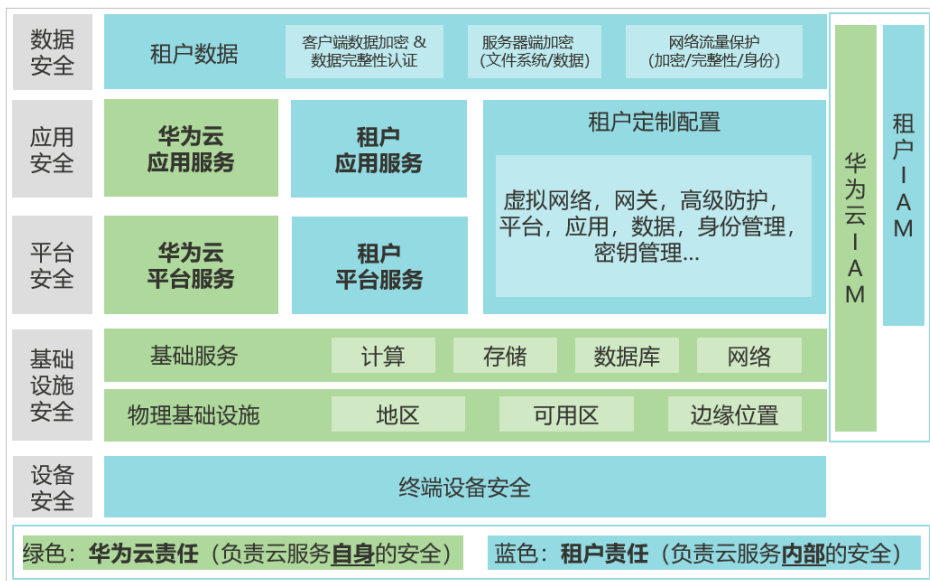
安全性是华为云与您的共同责任，如图9-1所示。

- **华为云**：负责云服务自身的安全，提供安全的云。华为云的安全责任在于保障其所提供的 IaaS、PaaS 和 SaaS 类云服务自身的安全，涵盖华为云数据中心的物理环境设施和运行其上的基础服务、平台服务、应用服务等。这不仅包括华为云基础设施和各项云服务技术的安全功能和性能本身，也包括运维运营安全，以及更广义的安全合规遵从。
- **租户**：负责云服务内部的安全，安全地使用云。华为云租户的安全责任在于对使用的 IaaS、PaaS 和 SaaS 类云服务内部的安全以及对租户定制配置进行安全有效的管理，包括但不限于虚拟网络、虚拟主机和访客虚拟机的操作系统，虚拟防火墙、API 网关和高级安全服务，各项云服务，租户数据，以及身份账号和密钥管理等方面的安全配置。

《[华为云安全白皮书](#)》详细介绍华为云安全性的构建思路与措施，包括云安全战略、责任共担模型、合规与隐私、安全组织与人员、基础设施安全、租户服务与租户安全、工程安全、运维运营安全、生态安全。



图 9-1 华为云安全责任共担模型



## 9.2 身份认证与访问控制

### 9.2.1 服务的访问控制

#### 身份认证

用户访问云数据库 GaussDB时支持对数据库用户进行身份验证，包含密码验证和IAM验证两种方式。

- 密码验证**  
 您需要对数据库实例进行管理，使用数据管理服务（Data Admin Service）登录数据库时，需要对账号密码进行验证，验证成功后方可进行操作。
- IAM验证**  
 您可以使用[统一身份认证服务](#)（Identity and Access Management，IAM）进行精细的权限管理。该服务提供用户身份认证、权限分配、访问控制等功能，可以帮助您安全地控制华为云资源的访问。您创建的IAM用户，需要通过验证用户和密码才可以使用GaussDB资源。具体请参见[创建IAM用户并登录](#)。

#### 访问控制

- 权限控制**  
 购买实例之后，您可以使用IAM为企业中的员工设置不同的访问权限，以达到不同员工之间的权限隔离，通过IAM进行精细的权限管理。具体内容请参见[权限管理](#)。
- VPC和子网**  
 虚拟私有云（Virtual Private Cloud，VPC）为云数据库构建隔离的、用户自主配置和管理的虚拟网络环境，提升用户云上资源的安全性，简化用户的网络部署。您可以在VPC中定义安全组、VPN、IP地址段、带宽等网络特性，方便管理、配置内部网络，进行安全、快捷的网络变更。  
 子网提供与其他网络隔离的、可以独享的网络资源，以提高网络安全性。

具体内容请参见[创建虚拟私有云和子网](#)。

- **安全组**

安全组是一个逻辑上的分组，为同一个虚拟私有云内具有相同安全保护需求并相互信任的和GaussDB数据库实例提供访问策略。为了保障数据库的安全性和稳定性，在使用GaussDB数据库实例之前，您需要设置安全组，开通需访问数据库的IP地址和端口。

具体请参见[设置安全组规则](#)。

## 9.3 数据保护技术

GaussDB通过多种数据保护手段和特性，保障存储在GaussDB中的数据安全可靠。

表 9-1 多种数据保护手段

| 数据保护手段      | 简要说明                                                                           | 详细介绍                     |
|-------------|--------------------------------------------------------------------------------|--------------------------|
| 传输加密（HTTPS） | 支持HTTP和HTTPS两种传输协议，为保证数据传输的安全性，推荐您使用更加安全的HTTPS协议。                              | <a href="#">构造请求</a>     |
| 数据备份        | 支持设置数据库的备份和恢复，来保障数据的可靠性。                                                       | <a href="#">数据备份</a>     |
| 敏感操作保护      | 控制台支持敏感操作保护，开启后执行删实例等敏感操作时，系统会进行身份验证，进一步保证GaussDB配置和数据的安全性。                    | <a href="#">敏感操作保护介绍</a> |
| SSL数据加密     | 可以使用SSL来加密数据库GaussDB和客户端的连接。SSL通过互相认证、使用数字签名确保完整性、使用加密确保私密性，以实现客户端和服务器之间的安全通讯。 | <a href="#">SSL连接数据库</a> |

## 9.4 审计与日志

### 审计

云审计服务（Cloud Trace Service，CTS），是华为云安全解决方案中专业的日志审计服务，提供对各种云资源操作记录的收集、存储和查询功能，可用于支撑安全分析、合规审计、资源跟踪和问题定位等常见应用场景。

CTS的详细介绍和开通配置方法，请参见[CTS快速入门](#)。

通过云审计服务，您可以记录与GaussDB实例相关的操作事件，便于日后的查询、审计和回溯。具体内容请参见[支持审计的关键操作列表](#)。

### 日志

出于分析或审计等目的，用户可以开启实例的日志记录功能。当用户开启日志记录功能后，GaussDB可以通过管理控制台下载日志。

- **错误日志**  
GaussDB支持下载错误日志。错误日志记录了数据库运行时的日志，通过错误日志有助于分析系统中存在的问题。错误日志的详细介绍，请参见[错误日志](#)。
- **慢日志**  
GaussDB支持下载慢日志。慢日志可以帮助您定位SQL语句执行慢的问题，从而进行优化。慢日志的详细介绍，请参见[慢日志](#)。

## 9.5 监控安全风险

云监控服务为用户提供一个针对云数据库、云服务器等资源的立体化监控平台。使您全面了解云上的资源使用情况、业务的运行状况，并及时收到异常告警做出反应，保证业务顺畅运行。

### 监控指标

GaussDB提供基于云监控服务的资源和操作监控能力，例如CPU使用率、网络吞吐量的等。支持的监控指标以及如何创建告警规则，具体请参见[监控指标](#)。

监控指标周期目前支持1分钟、10秒，默认监控周期为1分钟。通过开启秒级监控可以提高监控指标的精确值。

### 事件监控

事件监控提供了事件类型数据上报、查询和告警的功能。方便您将业务中的各类重要事件或对GaussDB的操作事件收集到云监控服务，并在事件发生时进行告警。

## 9.6 故障恢复

GaussDB会在数据库实例的备份时段中创建数据库实例的自动备份。系统根据您的指定的备份保留期（1-732天）保存数据库实例的自动备份。具体请参见[自动备份](#)。

GaussDB提供了多种方式恢复实例的数据，用以满足不同的使用场景：

- [恢复数据到当前实例、新实例或者已有实例](#)
- [数据库实例恢复到指定时间点](#)

### 多可用区

可用区指在同一区域下，电力、网络隔离的物理区域，可用区之间内网互通，不同可用区之间物理隔离。GaussDB支持将实例的节点分别部署在多个可用区，以此来实现AZ级高可用。

### 故障转移

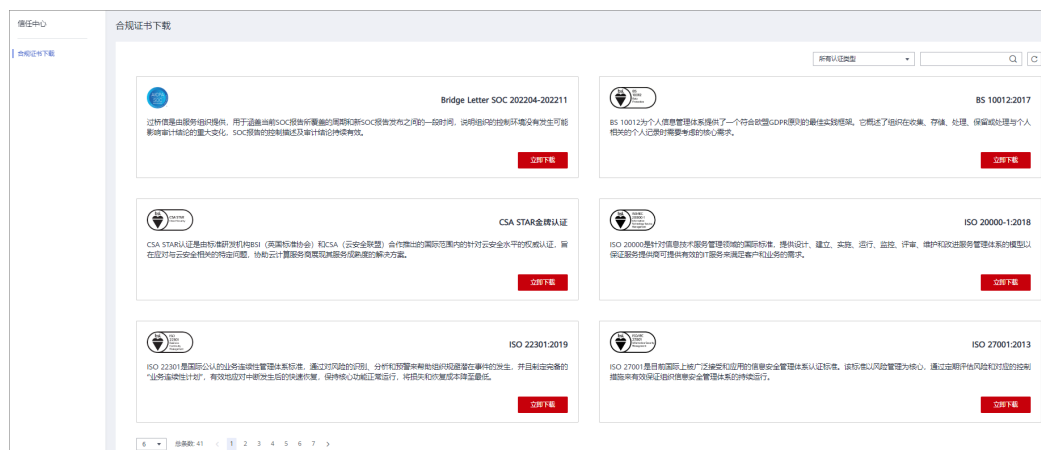
GaussDB是一个多节点的实例，主备版实例默认只有1个主节点，分布式版实例DN默认只有一个主节点，其余节点为备节点。当主节点发生故障时，备节点会自动升级为主节点，保证实例的可用性。

## 9.7 认证证书

### 合规证书

华为云服务及平台通过了多项国内外权威机构（ISO/SOC/PCI等）的安全合规认证，用户可自行[申请下载](#)合规资质证书。

图 9-2 合规证书下载



### 资源中心

华为云还提供以下资源来帮助用户满足合规性要求，具体请查看[资源中心](#)。

图 9-3 资源中心



### 销售许可证&软件著作权证书

另外，华为云还提供了以下销售许可证及软件著作权证书，供用户下载和参考。具体请查看[合规资质证书](#)。

图 9-4 销售许可证&软件著作权证书



# 10 权限管理

如果您需要对购买的GaussDB资源，为企业中的员工设置不同的访问权限，为达到不同员工之间的权限隔离，您可以使用统一身份认证服务（Identity and Access Management，简称IAM）进行精细的权限管理。该服务提供用户身份认证、权限分配、访问控制等功能，可以帮助您安全的控制华为云资源的访问。

如果华为云账号已经能满足您的要求，不需要创建独立的IAM用户进行权限管理，您可以跳过本章节，不影响您使用GaussDB服务的其它功能。

通过IAM，您可以在华为云账号中给员工创建IAM用户，并授权控制创建的IAM用户对华为云资源的访问范围。例如您的员工中有负责软件开发的人员，您希望开发人员拥有GaussDB的使用权限，但是不希望开发人员拥有删除GaussDB等高危操作的权限，那么您可以使用IAM为开发人员创建用户，通过授予仅能使用GaussDB，但是不允许删除GaussDB的权限，控制开发人员对GaussDB资源的使用范围。

IAM是华为云提供权限管理的基础服务，无需付费即可使用，您只需要为您账号中的资源进行付费。关于IAM的详细介绍，请参见[IAM产品介绍](#)。

## GaussDB 权限

默认情况下，管理员创建的IAM用户没有任何权限，需要将其加入用户组，并给用户组授予策略或角色，才能使得用户组中的用户获得对应的权限，这一过程称为授权。授权后，用户就可以基于被授予的权限对云服务进行操作。

GaussDB部署时通过物理区域划分，为项目级服务。授权时，“作用范围”需要选择“区域级项目”，然后在指定区域（如华北-北京1）对应的项目（cn-north-1）中设置相关权限，并且该权限仅对此项目生效；如果在“所有项目”中设置权限，则该权限在所有区域项目中都生效。访问GaussDB时，需要先切换至授权区域。

根据授权精细程度分为角色和策略。

- **角色**：IAM最初提供的一种根据用户的工作职能定义权限的粗粒度授权机制。该机制以服务为粒度，提供有限的服务相关角色用于授权。由于华为云各服务之间存在业务依赖关系，因此给用户授予角色时，可能需要一并授予依赖的其他角色，才能正确完成业务。角色并不能满足用户对精细化授权的要求，无法完全达到企业对权限最小化的安全管控要求。
- **策略**：IAM最新提供的一种细粒度授权的能力，可以精确到具体服务的操作、资源以及请求条件等。基于策略的授权是一种更加灵活的授权方式，能够满足企业对权限最小化的安全管控要求。例如：针对GaussDB服务，管理员能够控制IAM用户仅能对某一类数据库资源进行指定的管理操作。多数细粒度策略以API接口为粒度进行权限拆分，GaussDB支持的API授权项请参见[策略及授权项说明](#)。

如表10-1所示，包括了GaussDB的所有系统权限。

表 10-1 GaussDB 系统权限

| 策略名称                   | 描述                    | 类别   |
|------------------------|-----------------------|------|
| GaussDB FullAccess     | 云数据库GaussDB服务的所有执行权限。 | 系统策略 |
| GaussDB ReadOnlyAccess | 云数据库GaussDB服务的只读访问权限。 | 系统策略 |

表10-2列出了GaussDB常用操作与系统权限的授权关系，您可以参照该表选择合适的系统权限。

表 10-2 常用操作与系统权限的关系

| 操作            | GaussDB FullAccess | GaussDB ReadOnlyAccess |
|---------------|--------------------|------------------------|
| 创建GaussDB实例   | √                  | x                      |
| 删除GaussDB实例   | √                  | x                      |
| 查询GaussDB实例列表 | √                  | √                      |

表 10-3 常用操作与对应授权项

| 操作名称    | 授权项                                           | 备注                                                                                                                                                                                                                                                                                              |
|---------|-----------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 创建数据库实例 | gaussdb:instance:create<br>gaussdb:param:list | 界面选择VPC、子网、安全组需要配置：<br>vpc:vpcs:list<br>vpc:vpcs:get<br>vpc:subnets:get<br>vpc:securityGroups:get<br>创建包周期实例需要配置<br>bss:order:update<br>bss:order:view<br>bss:balance:view<br>创建加密实例需要在项目上配置：<br>kms:cmk:get<br>kms:cmk:list<br>操作失败上报事件监控需要配置：<br>"ces:alarmsOnOff:put"<br>"ces:alarms:create" |
| 规格变更    | gaussdb:instance:modifySpec                   | 操作失败上报事件监控需要配置：<br>"ces:alarmsOnOff:put"<br>"ces:alarms:create"                                                                                                                                                                                                                                 |
| 扩容节点    | gaussdb:instance:modifySpec                   | 操作失败上报事件监控需要配置：<br>"ces:alarmsOnOff:put"<br>"ces:alarms:create"                                                                                                                                                                                                                                 |
| 磁盘扩容    | gaussdb:instance:modifySpec                   | 操作失败上报事件监控需要配置：<br>"ces:alarmsOnOff:put"<br>"ces:alarms:create"                                                                                                                                                                                                                                 |
| 重启数据库实例 | gaussdb:instance:restart                      | 操作失败上报事件监控需要配置：<br>"ces:alarmsOnOff:put"<br>"ces:alarms:create"                                                                                                                                                                                                                                 |



| 操作名称      | 授权项                                        | 备注                                                                                                                          |
|-----------|--------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------|
| 删除数据库实例   | gaussdb:instance:delete                    | 退订包周期实例需要配置：<br>"bss:unsubscribe:update"<br>操作失败上报事件监控需要配置：<br>"ces:alarmsOnOff:put"<br>"ces:alarms:create"                 |
| 查询数据库实例列表 | gaussdb:instance:list                      | 无                                                                                                                           |
| 实例详情      | gaussdb:instance:list                      | 实例详情界面展示VPC、子网、安全组，需要对应配置vpc:*:get和vpc:*:list。显示磁盘已使用大小，需要配置ces:*:list。                                                     |
| 修改数据库实例密码 | gaussdb:instance:modify                    | 操作失败上报事件监控需要配置：<br>"ces:alarmsOnOff:put"<br>"ces:alarms:create"                                                             |
| 修改实例名称    | gaussdb:instance:modify                    | 无                                                                                                                           |
| 绑定/解绑公网IP | gaussdb:instance:modify                    | 界面列出公网IP需要配置：<br>vpc:publicips:get<br>vpc:publicips:list<br>操作失败上报事件监控需要配置：<br>"ces:alarmsOnOff:put"<br>"ces:alarms:create" |
| 创建参数模板    | gaussdb:param:create<br>gaussdb:param:list | 无                                                                                                                           |
| 修改参数模板    | gaussdb:param:modify                       | 无                                                                                                                           |
| 获取参数模板列表  | gaussdb:param:list                         | 无                                                                                                                           |
| 应用参数模板    | gaussdb:param:apply                        | 操作失败上报事件监控需要配置：<br>"ces:alarmsOnOff:put"<br>"ces:alarms:create"                                                             |
| 删除参数模板    | gaussdb:param:delete                       | 无                                                                                                                           |

| 操作名称       | 授权项                                 | 备注                                                                                                                                                                           |
|------------|-------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 创建手动备份     | gaussdb:backup:create               | 操作失败上报事件监控<br>需要配置：<br>"ces:alarmsOnOff:put"<br>"ces:alarms:create"                                                                                                          |
| 获取备份列表     | gaussdb:backup:list                 | 无                                                                                                                                                                            |
| 修改备份策略     | gaussdb:instance:modifyBackupPolicy | 无                                                                                                                                                                            |
| 删除手动备份     | gaussdb:backup:delete               | 操作失败上报事件监控<br>需要配置：<br>"ces:alarmsOnOff:put"<br>"ces:alarms:create"                                                                                                          |
| 恢复到新实例     | gaussdb:instance:create             | 界面选择VPC、子网、<br>安全组需要配置：<br>vpc:vpcs:list<br>vpc:vpcs:get<br>vpc:subnets:get<br>vpc:securityGroups:get<br>操作失败上报事件监控<br>需要配置：<br>"ces:alarmsOnOff:put"<br>"ces:alarms:create" |
| 查询项目标签     | gaussdb:tag:list                    | 无                                                                                                                                                                            |
| 批量添加删除项目标签 | gaussdb:instance:dealTag            | 无                                                                                                                                                                            |
| 修改配额       | gaussdb:quota:modify                | 无                                                                                                                                                                            |
| 查询预定义标签    | gaussdb:instance:list               | 查询预定义标签需要配置：<br>tms:resourceTags:list                                                                                                                                        |
| 查询配置日志组    | -                                   | 查询配置日志组需要配置：<br>lts:groups:get                                                                                                                                               |
| 查询配置日志流    | -                                   | 查询配置日志流需要配置：<br>lts:topics:get                                                                                                                                               |

| 操作名称     | 授权项                                                                                                                           | 备注                                                                                                                                                                                                                                                                                               |
|----------|-------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 存储空间自动扩容 | gaussdb:instance:autoEnlargePolicy                                                                                            | 如果选择自动扩容，IAM主账号（原华为账号）不需要添加授权项，IAM子账号需要添加如下IAM授权项： <ul style="list-style-type: none"><li>iam:agencies:listAgencies</li><li>iam:agencies:createAgency</li><li>iam:permissions:listRolesForAgencyOnProject</li><li>iam:permissions:grantRoleToGroupOnProject</li><li>iam:roles:listRoles</li></ul> |
| dbmind操作 | gaussdb:instance:sqlCollect<br>gaussdb:instance:sqlDiagnosis<br>gaussdb:instance:riskAnalysis<br>gaussdb:instance:indexAdvise | gaussdb:instance:sqlCollect SQL采集<br>gaussdb:instance:sqlDiagnosis 慢SQL诊断<br>gaussdb:instance:riskAnalysis 风险预测<br>gaussdb:instance:indexAdvise 索引推荐                                                                                                                                             |

# 11 约束与限制

云数据库 GaussDB在使用上有一些固定限制，用来提高实例的稳定性和安全性，具体详见[表11-1](#)。

GaussDB 单副本部署实例禁止在生产环境使用，如在生产环境使用，不承诺SLA。且在功能上有相关约束，具体约束限制请参考[表11-2](#)。

表 11-1 功能约束与限制

| 功能          | 使用限制                                                                                                                                                                                                                                                                                              |
|-------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 数据库访问       | <ul style="list-style-type: none"><li>● 如果GaussDB实例未开通公网访问，则该实例必须与云主机弹性云服务器处在同一个虚拟私有云子网内才能相互访问。</li><li>● 弹性云服务器必须处于目标GaussDB实例所属安全组允许访问的范围内。<br/>如果GaussDB实例与弹性云服务器处于不同的安全组，系统默认不能访问。需要在GaussDB的安全组添加一条“入”的访问规则。</li><li>● GaussDB实例的默认端口为8000。</li><li>● 数据库端口支持创建时设置，后期可修改。</li></ul>          |
| 部署          | 实例所部署的服务器，对用户都不可见，即只允许应用程序通过IP地址和端口访问数据库。                                                                                                                                                                                                                                                         |
| 数据库的root权限  | 创建实例页面只提供管理员root用户权限。<br><b>说明</b><br>在2022.08.30后，GaussDB为root用户开放了sysadmin权限。新创建实例的root用户都将拥有sysadmin权限，而存量实例执行版本升级后，root用户也将拥有sysadmin权限，如果需要进行版本升级，请联系技术支持处理。<br>如果存量实例未进行版本升级，则管理员root用户权限为：creatorole, createdb和monadmin。由于旧版本root权限低于完整的管理员用户权限，部分SQL语法/函数执行时会报权限不足，例如：create tablespace 等 |
| 重启GaussDB实例 | 无法通过命令行重启，必须通过GaussDB的管理控制台操作重启实例。                                                                                                                                                                                                                                                                |

| 功能          | 使用限制                                                                                                                                                                                                                                                                                                                                     |
|-------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| GaussDB备份查看 | GaussDB实例在对象存储服务上的备份文件，对用户不可见。                                                                                                                                                                                                                                                                                                           |
| 变更配置        | <ul style="list-style-type: none"> <li>默认不支持将规格参数变小，如需要将规格参数变小，您可以联系华为云技术支持，由华为云工程师给出分析评估后进行处理。</li> <li>规格变更前，须确保实例状态正常。在实例异常、节点异常、磁盘满的情况下不允许进行规格变更。</li> <li>高可用（1主2备）部署形态下，规格变更过程中会进行主备倒换，主备倒换过程中会有1min左右的业务中断。</li> <li>单副本的部署形态下，规格变更过程中会进行中断重启，中断重启过程中会有5~10min的业务中断。</li> <li>修改CPU/内存后，将会重启数据库实例，重启数据库实例将导致数据库业务短暂中断。</li> </ul> |
| 故障切换        | 对于主备版，主节点切换备节点，大概有10s左右服务不可用。                                                                                                                                                                                                                                                                                                            |
| 数据恢复        | 为避免数据丢失，建议数据恢复前备份重要数据。                                                                                                                                                                                                                                                                                                                   |
| 存储空间        | 若实例的磁盘空间已满，此时不可进行数据库写入操作，您需要扩容磁盘使实例恢复到正常状态。建议定期检查存储空间。                                                                                                                                                                                                                                                                                   |
| 性能优化        | 性能调优过程有时候需要重启集群，可能会中断当前业务。                                                                                                                                                                                                                                                                                                               |

表 11-2 主备版单副本实例的基本功能限制

| 功能     | 3.0以下版本是否支持 | 3.0以上版本是否支持 |
|--------|-------------|-------------|
| 创建实例   | Y           | Y           |
| 重启实例   | Y           | Y           |
| 参数修改   | Y           | Y           |
| 参数应用   | Y           | Y           |
| 重置密码   | Y           | Y           |
| 全量备份   | N           | Y           |
| 增量备份   | N           | Y           |
| 删除备份   | N           | Y           |
| 修改备份策略 | N           | Y           |

| 功能          | 3.0以下版本是否支持      | 3.0以上版本是否支持      |
|-------------|------------------|------------------|
| 恢复到当前实例     | N                | Y                |
| 恢复到新实例      | N                | Y                |
| 恢复到已有实例     | N                | Y                |
| 磁盘扩容        | Y                | Y                |
| 规格变更        | Y                | Y                |
| 热补丁升级       | N                | Y                |
| 就地升级        | Y(仅支持升级至3.0以上版本) | Y                |
| 灰度升级        | N                | Y(仅3.207及以上版本支持) |
| 查看监控指标      | Y                | Y                |
| 实例删除        | Y                | Y                |
| 回收站         | N                | Y                |
| 查询磁盘使用      | Y                | Y                |
| 创建数据库       | Y                | Y                |
| 查询数据库       | Y                | Y                |
| 创建schema及用户 | Y                | Y                |
| 删除schema及用户 | Y                | Y                |
| 执行数据库操作     | Y                | Y                |

# 12 计费说明

GaussDB目前支持按需计费和包周期计费方式。

## 计费项

GaussDB对您选择的数据库实例、数据库存储和备份存储（可选）收费。

表 12-1 GaussDB 计费项说明

| 计费项      | 计费说明                                                                            |
|----------|---------------------------------------------------------------------------------|
| 数据库实例    | 对所选的实例规格进行计费，提供按需计费和包周期计费方式。                                                    |
| 数据库存储    | 对数据库存储空间进行计费，提供按需计费和包周期计费方式。                                                    |
| 备份存储（可选） | GaussDB提供了部分免费存储空间，用于存放您的备份数据，其总容量约为您购买存储容量的100%。备份存储用量超过数据库存储空间的100%，开始按照阶梯计费。 |
| 公网流量     | GaussDB实例支持公网访问，公网访问会产生带宽流量费；GaussDB数据库实例在云内部网络产生的流量不计费。                        |

GaussDB管理费用详情，请参见[产品价格详情](#)。您可以通过GaussDB提供的价格计算器，选择您需要的实例规格，来快速计算出购买实例的参考价格。

## 计费模式

提供按小时、按月、按年的计费方式供您灵活选择，使用越久越便宜。

- 预付费（包年包月）：这种购买方式相对于按需付费提供更大的折扣，对于长期使用，推荐该方式。
- 按需付费（小时）：这种购买方式比较灵活，可以即开即停，实例从“开通”开启计费到“删除”结束计费，按实际购买时长计费。以自然小时为单位整点计费，不足一小时按实际使用时长计费。

## 变更配置

- 变更GaussDB实例规格：您可以根据业务需求变更GaussDB实例规格，变更后即刻按照变更后的实例规格的价格计费。
- 扩容存储空间：您可以根据业务需求增加您的存储空间，扩容后即刻按照新的存储空间计费。您需要注意的是存储空间只允许扩容，不能缩容。扩容磁盘的大小必须是（40\*分片数量）的整数倍。

## 续费

目前GaussDB提供“按需计费”和“包年/包月”计费方式的购买方式，您可以根据业务需要定制相应计算能力和存储空间的数据数据库实例。

- “按需计费”方式，即按实际使用时长计费，以自然小时为单位整点计费，不足一小时按实际使用时长计费，只要您账户上有足够余额，就可以一直使用服务。当账户余额不足时，就会导致欠费，因此在欠费前请及时充值。
- “包年/包月”计费方式，您在购买时一次性付费，使用过程中不会再另外扣费，只要您的账户上有足够余额，则不会影响您的使用。

如需续费，请进入“[续费管理](#)”页面进行续费操作。

## 到期与欠费

### ● 服务到期

“按需计费”实例，没有到期的概念。

“包年/包月”实例到期后无法在管理控制台进行该实例的操作，相关接口也无法调用，自动化监控或告警等运维也会停止。如果在保留期结束时您没有续费，实例将终止服务，系统中的数据也将被永久删除。

若您购买的实例已到期，请参见费用中心用户指南中[资源停止服务或逾期释放说明](#)章节进行处理。

### ● 欠费

“包年/包月”实例，没有欠费的概念。

“按需计费”实例是按每小时扣费，当余额不足，无法对上一个小时的费用进行扣费，就会导致实例欠费。您续费后解冻实例，可继续正常使用，请注意在保留期进行的续费，是以原到期时间作为生效时间，您应当支付从进入保留期开始到续费时的服务费用。

若您购买的实例已欠费，请参见费用中心用户指南中[欠费还款](#)章节进行处理。



# 13 GaussDB 与其他服务的关系

GaussDB与其他服务的关系如[表13-1](#)。

**表 13-1** 与其他服务的关系

| 相关服务              | 交互功能                                                                                      |
|-------------------|-------------------------------------------------------------------------------------------|
| 弹性云服务器 (ECS)      | GaussDB服务通过弹性云服务器 (Elastic Cloud Server, 简称ECS) 远程连接GaussDB可以有效的降低应用响应时间、节省公网流量费用。        |
| 虚拟私有云 (VPC)       | 对您的GaussDB实例进行网络隔离和访问控制。                                                                  |
| 对象存储服务 (OBS)      | 存储GaussDB实例的自动和手动备份数据。                                                                    |
| 云监控服务 (Cloud Eye) | 云监控服务是一个开放性的监控平台, 帮助用户实时监测 GaussDB资源的动态。云监控服务提供多种告警方式以保证及时预警, 为您的服务正常运行保驾护航。              |
| 云审计服务 (CTS)       | 云审计服务 (Cloud Trace Service, 简称CTS), 为用户提供云服务资源的操作记录, 供您查询、审计和回溯使用。                        |
| 企业管理服务 (EPS)      | 企业管理服务 (Enterprise Project Management Service, 简称EPS) 提供统一的云资源按企业项目管理, 以及企业项目内的资源管理、成员管理。 |
| 标签管理服务 (TMS)      | 标签管理服务 (Tag Management Service, 简称TMS) 是一种快速便捷将标签集中管理的可视化服务, 提供跨区域、跨服务的集中标签管理和资源分类功能。     |
| 数据管理服务 (DAS)      | 使用数据管理服务 (Data Admin Service, 简称DAS), 通过专业优质的可视化操作界面, 提高数据管理工作的效率和安全。                     |

# A 修订记录

| 发布日期       | 修改说明                                                                                                                                                                                                                                                                                                                                                                                 |
|------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 2024-05-14 | <ul style="list-style-type: none"><li>规格类型名称变更。变更后名称为：<ul style="list-style-type: none"><li>独享型（1:4）</li><li>独享型（1:8）</li><li>通用型（1:4）</li><li>鲲鹏独享型（1:4）</li><li>鲲鹏独享型（1:8）</li></ul></li><li>新增基础版规格。详见<a href="#">数据库实例规格</a>。</li></ul>                                                                                                                                          |
| 2023-12-30 | <p>包含以下内容：</p> <ul style="list-style-type: none"><li>新增<a href="#">GaussDB数据库引擎版本号说明</a>。</li><li>新增<a href="#">GaussDB数据库和内核引擎版本对应关系</a>。</li><li>新增<a href="#">查看云数据库GaussDB数据库和内核引擎版本</a>。</li><li>新增<a href="#">日志</a>。</li><li>优化<a href="#">图解云数据库GaussDB</a>、<a href="#">什么是云数据库GaussDB</a>、<a href="#">数据库实例规格</a>、<a href="#">数据库实例参数</a>和<a href="#">数据保护技术</a>。</li></ul> |