

阿里云 云数据库 POLARDB

POLARDB for MySQL用户指南

文档版本：20191120

法律声明

阿里云提醒您在阅读或使用本文档之前仔细阅读、充分理解本法律声明各条款的内容。如果您阅读或使用本文档，您的阅读或使用行为将被视为对本声明全部内容的认可。

1. 您应当通过阿里云网站或阿里云提供的其他授权通道下载、获取本文档，且仅能用于自身的合法合规的业务活动。本文档的内容视为阿里云的保密信息，您应当严格遵守保密义务；未经阿里云事先书面同意，您不得向任何第三方披露本手册内容或提供给任何第三方使用。
2. 未经阿里云事先书面许可，任何单位、公司或个人不得擅自摘抄、翻译、复制本文档内容的部分或全部，不得以任何方式或途径进行传播和宣传。
3. 由于产品版本升级、调整或其他原因，本文档内容有可能变更。阿里云保留在没有任何通知或者提示下对本文档的内容进行修改的权利，并在阿里云授权通道中不时发布更新后的用户文档。您应当实时关注用户文档的版本变更并通过阿里云授权渠道下载、获取最新版的用户文档。
4. 本文档仅作为用户使用阿里云产品及服务的参考性指引，阿里云以产品及服务的“现状”、“有缺陷”和“当前功能”的状态提供本文档。阿里云在现有技术的基础上尽最大努力提供相应的介绍及操作指引，但阿里云在此明确声明对本文档内容的准确性、完整性、适用性、可靠性等不作任何明示或暗示的保证。任何单位、公司或个人因为下载、使用或信赖本文档而发生任何差错或经济损失的，阿里云不承担任何法律责任。在任何情况下，阿里云均不对任何间接性、后果性、惩戒性、偶然性、特殊性或刑罚性的损害，包括用户使用或信赖本文档而遭受的利润损失，承担责任（即使阿里云已被告知该等损失的可能性）。
5. 阿里云文档中所有内容，包括但不限于图片、架构设计、页面布局、文字描述，均由阿里云和/或其关联公司依法拥有其知识产权，包括但不限于商标权、专利权、著作权、商业秘密等。非经阿里云和/或其关联公司书面同意，任何人不得擅自使用、修改、复制、公开传播、改变、散布、发行或公开发表阿里云网站、产品程序或内容。此外，未经阿里云事先书面同意，任何人不得为了任何营销、广告、促销或其他目的使用、公布或复制阿里云的名称（包括但不限于单独为或以组合形式包含“阿里云”、“Aliyun”、“万网”等阿里云和/或其关联公司品牌，上述品牌的附属标志及图案或任何类似公司名称、商号、商标、产品或服务名称、域名、图案标示、标志、标识或通过特定描述使第三方能够识别阿里云和/或其关联公司）。
6. 如若发现本文档存在任何错误，请与阿里云取得直接联系。

通用约定

格式	说明	样例
	该类警示信息将导致系统重大变更甚至故障，或者导致人身伤害等结果。	 禁止： 重置操作将丢失用户配置数据。
	该类警示信息可能会导致系统重大变更甚至故障，或者导致人身伤害等结果。	 警告： 重启操作将导致业务中断，恢复业务时间约十分钟。
	用于警示信息、补充说明等，是用户必须了解的内容。	 注意： 权重设置为0，该服务器不会再接受新请求。
	用于补充说明、最佳实践、窍门等，不是用户必须了解的内容。	 说明： 您也可以通过按Ctrl + A选中全部文件。
>	多级菜单递进。	单击设置 > 网络 > 设置网络类型。
粗体	表示按键、菜单、页面名称等UI元素。	在结果确认页面，单击确定。
Courier字体	命令。	执行 <code>cd /d C:/window</code> 命令，进入Windows系统文件夹。
##	表示参数、变量。	<code>bae log list --instanceid Instance_ID</code>
[]或者[a b]	表示可选项，至多选择一个。	<code>ipconfig [-all -t]</code>
{ }或者{a b}	表示必选项，至多选择一个。	<code>switch {active stand}</code>

目录

法律声明.....	I
通用约定.....	I
1 概述.....	1
2 POLARDB for MySQL快速入门.....	2
3 读写分离.....	3
4 使用存储包.....	8
5 待处理事件.....	13
6 设置集群白名单.....	16
7 计费管理.....	19
7.1 按小时付费转包年包月.....	19
7.2 手动续费集群.....	20
7.3 自动续费集群.....	22
8 连接数据库集群.....	29
8.1 查看连接地址.....	29
8.2 设置/释放自定义集群地址.....	31
8.3 连接数据库集群.....	36
9 集群管理.....	42
9.1 创建POLARDB for MySQL数据库集群.....	42
9.2 增加或删除节点.....	45
9.3 变更配置.....	49
9.4 临时升配.....	53
9.5 设置集群参数.....	56
9.6 设置可维护窗口.....	58
9.7 重启节点.....	59
9.8 主备切换.....	60
9.9 释放集群.....	61
9.10 克隆集群.....	62
10 多可用区部署.....	64
11 账号管理.....	69
11.1 账号概述.....	69
11.2 注册和登录阿里云账号.....	70
11.3 创建和使用子账号.....	72
11.4 创建数据库账号.....	75
11.5 管理数据库账号.....	78
12 数据库管理.....	81
13 备份与恢复.....	83

13.1 备份数据.....	83
13.2 恢复数据.....	84
14 诊断与优化.....	89
14.1 集群总览.....	89
14.2 一键诊断.....	91
14.3 问题分析.....	96
14.3.1 查询和终止实时会话.....	96
14.3.2 查看空间使用详情.....	96
14.3.3 查看事务.....	97
14.3.4 诊断热点表.....	98
14.3.5 SQL调优.....	98
14.3.6 事件回溯.....	102
14.3.7 查看节点性能诊断报告.....	103
14.4 慢SQL.....	104
14.5 性能洞察.....	106
15 性能监控与报警.....	108
16 SQL洞察.....	113
17 并行查询.....	119
17.1 并行查询.....	119
17.2 并行查询使用示例.....	124
17.3 并行查询使用限制.....	134
18 如何开启Binlog.....	136
19 POLARDB for MySQL常见问题.....	139
20 隐藏目录.....	142
20.1 查看日志.....	142

1 概述

POLARDB是阿里云自研的下一代关系型云数据库，有三个独立的引擎，分别可以100%兼容MySQL、100%兼容PostgreSQL、高度兼容Oracle语法。存储容量最高可达100TB，单库最多可扩展到16个节点，适用于企业多样化的数据库应用场景。

POLARDB采用存储和计算分离的架构，所有计算节点共享一份数据，提供分钟级的配置升降级、秒级的故障恢复、全局数据一致性和免费的数据备份容灾服务。POLARDB既融合了商业数据库稳定可靠、高性能、可扩展的特征，又具有开源云数据库简单开放、自我迭代的优势，例如POLARDB for MySQL性能最高可以提升至MySQL的6倍，而成本只有商用数据库的1/10。

基本概念

- 集群

一个集群包含一个主节点以及最多15个只读节点（最少一个，用于提供Active-Active高可用）。集群ID以pc开头（代表POLARDB cluster）。

- 节点

一个独立占用物理内存的数据库服务进程。节点ID以pi开头（代表POLARDB instance）。

- 数据库

在节点下创建的逻辑单元，一个节点可以创建多个数据库，数据库在节点内的命名唯一。

- 地域和可用区

地域是指物理的数据中心。可用区是指在同一地域内，拥有独立电力和网络的物理区域。更多信息请参考[阿里云全球基础设施](#)。

控制台

阿里云提供了简单易用的Web控制台，方便您操作阿里云的各种产品和服务，包括云数据库POLARDB。在控制台上，您可以创建、连接和配置POLARDB数据库。

关于控制台的界面介绍，请参考[阿里云管理控制台](#)。

POLARDB控制台地址：<https://polardb.console.aliyun.com>

2 POLARDB for MySQL快速入门

快速入门旨在介绍如何创建POLARDB for MySQL集群、进行基本设置以及连接数据库集群，使您能够了解从购买POLARDB到开始使用的流程。

使用流程

通常，从购买POLARDB（创建新集群）到可以开始使用，您需要完成如下操作。

1. [创建POLARDB for MySQL数据库集群](#)
2. [设置集群白名单](#)
3. [#unique_7](#)
4. [#unique_8](#)
5. [连接数据库集群](#)

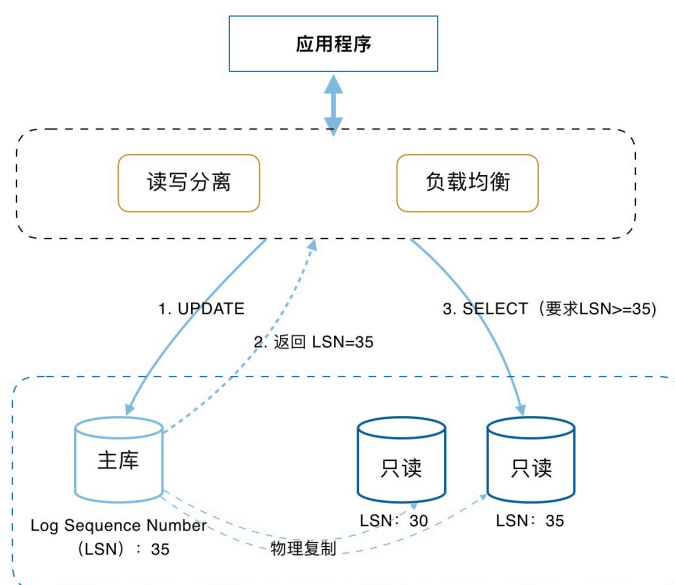
3 读写分离

POLARDB for MySQL集群自带读写分离功能。应用程序只需连接一个集群地址，写请求会自动发往主节点，读请求会自动根据各节点的负载（当前未完成的请求数）发往主节点或只读节点。

功能优势

- 读一致性

在POLARDB的链路中间层进行读写分离时，中间层会追踪（track）各个节点已经应用（apply）的重做日志（redolog）位点，即日志序号（LSN）；同时，每次数据更新时将此次更新的位点记录为Session LSN。当POLARDB收到新的请求时，将Session LSN与各个节点的LSN进行比较，仅将请求发往LSN早于Session LSN的节点，从而保证会话的一致性。表面上看该方案可能导致主节点压力大，但是因为POLARDB是物理复制，速度极快，在上述场景中，当更新完成后，返回客户端结果时复制就同步在进行，而当下一个读请求到来时，主节点和只读节点之间的数据复制极有可能已经完成，然后大多数应用场景都是读多写少，所以经验证在该机制下即保证了会话一致性，也保证了读写分离负载均衡的效果。



- 原生支持读写分离，提升性能

如果您在云上通过自己搭建代理层实现读写分离，在数据到达数据库之前需要经历多个组件的语句解析和转发，对响应延迟有较大的影响。而POLARDB读写分离在已有的高安全链路中直接内置，没有任何额外的组件来消耗时间，能够有效降低延迟，提升处理速度。

- 维护方便

在传统模式下，您需要在应用程序中配置主节点和每个只读节点的连接地址，并且对业务逻辑进行拆分，才能实现将写请求发往主节点而将读请求发往各个节点。

POLARDB提供集群地址，应用程序连接该地址后即可对主节点和只读节点进行读写操作，读写请求会被自动转发，转发逻辑完全对使用者透明，可降低维护成本。

同时，您只需添加只读节点的个数，即可不断扩展系统的处理能力，应用程序无需做任何修改。

- 节点健康检查，提升数据库系统的可用性

读写分离模块自动对集群内的所有节点进行健康检查，当发现某个节点宕机或者延迟超过阈值后，将不再分配读请求给该节点，读写请求在剩余的健康节点间进行分配。以此确保单个只读节点发生故障时，不会影响应用的正常访问。当节点被修复后，POLARDB会自动将该节点纳回请求分配体系内。

- 免费使用，降低资源及维护成本

免费提供读写分离功能，无需支付任何额外费用。

转发逻辑

- 可读可写模式转发逻辑：

- 只发往主节点

- 所有DML操作（INSERT、UPDATE、DELETE、SELECT FOR UPDATE）。
 - 所有DDL操作（建表/库、删表/库、变更表结构、权限等）。
 - 所有事务中的请求。
 - 用户自定义函数。
 - 存储过程。
 - EXECUTE语句。
 - *Multi Statements*。
 - 使用到临时表的请求。
 - SELECT last_insert_id()。
 - 所有对用户变量的查询和更改。
 - SHOW PROCESSLIST。
 - KILL（SQL语句中的KILL，非命令KILL）。

- 发往只读节点或主节点

- 非事务中的读请求。
 - COM_STMT_EXECUTE命令。

- 总是发往所有节点

- 所有系统变量的更改。
 - USE命令。
 - COM_STMT_PREPARE命令。
 - COM_CHANGE_USER/COM_QUIT/COM_SET_OPTION等命令。

- 只读模式转发逻辑：

- 不允许执行DDL、DML操作。
 - 所有请求按照负载均衡的方式转发到各只读节点。
 - 所有请求不会转发到可读可写节点。

使用限制

- 若执行了*Multi Statements*或存储过程，当前连接的后续请求会全部路由到主节点，需断开当前连接并重新连接才能恢复读写分离。

- 集群地址在只读模式下，不支持设置环境变量，例如 `set @endtime=now(); select * from tab where dt < @endtime`，可能导致查询无法获得正确的环境变量。
- 使用视图时无法保证会话一致性，例如 `CREATE VIEW tab2 AS SELECT * FROM tab1; INSERT INTO tab1(key, value) (1, 1); SELECT * FROM tab2 where key=1;` 可能会因为延迟而查不到结果。

申请或修改集群地址

1. 登录[POLARDB控制台](#)。
2. 在控制台左上角，选择地域。
3. 单击目标集群ID。
4. 在基本信息页面的访问信息区域，找到集群地址（推荐）。
5. 单击申请，在弹出的对话框中单击确认，刷新后即可看到集群地址。



说明：

存量集群若未申请集群读写分离连接地址的需要手动申请集群地址，新购集群自动开通集群地址。若已有集群地址，请直接查看第6步。

6. 单击修改，设置新的集群地址，并单击提交。

访问信息 ①

白名单 ②

白名单列表  [配置](#)

主地址 ②

私网  [修改](#)

公网 [申请](#)

集群地址【推荐】 ② [新增自定义地址](#)

默认地址 (pe ) [修改配置](#)

读写模式 可读可写 (自动读写分离)

私网  [修改](#)

公网 [申请](#)

[节点配置](#)

[高级配置](#)

FAQ

1. 为什么刚插入的语句，立即查的时候查不到？

答：读写分离的架构下，主节点和只读节点之间复制会有延迟，但POLARDB支持会话一致性，即同一个会话内保证能读到之前的更新。

2. 为什么只读节点没有压力？

答：默认情况下事务中的请求都会路由到主节点，若是用sysbench做压测，0.5版本的sysbench可以加上--oltp-skip-trx=on，1.0版本的sysbench可以加上--skip-trx=on去掉事务，若业务上因为事务较多导致只读库负载过低，可以提交工单开启读写分离下的分布式事务。

3. 为什么某个节点请求数比别的节点多？

答：当前是根据负载来分发请求的，负载小的节点接收的请求数会更多。

4. 是否支持0毫秒延迟的读取？

答：POLARDB集群的主节点和只读节点在正常负载情况下，具有毫秒级的延迟，读写分离连接地址暂时不支持在数据写入后0毫秒的读取。如果要求0毫秒延迟的读取，可使用主地址（动态指向POLARDB主节点）将读写请求发给主节点。

5. 新增的只读节点会自动加入到读写分离吗？

新增只读节点之后新建的读写分离连接会转发请求到该只读节点。新增只读节点之前建立的读写分离连接不会转发请求到新增的只读节点，需要断开该连接并重新建立连接，例如，重启应用。

相关API

API	描述
#unique_11	创建POLARDB集群的公网地址。
#unique_12	创建POLARDB自定义集群地址。
#unique_13	查询POLARDB集群的地址信息。
#unique_14	修改POLARDB集群地址属性。
#unique_15	修改POLARDB集群默认访问地址前缀。
#unique_16	释放POLARDB集群地址（除了自定义集群地址的私网地址）。
#unique_17	释放POLARDB自定义集群地址。

4 使用存储包

为了更好地帮助您降低存储成本，POLARDB推出了预付费形式的存储包。

背景信息

POLARDB的存储空间无需手动配置，根据数据量自动伸缩，您只需为实际使用的数据量按量付费。当您的数据量较大时，推荐您使用POLARDB存储包，相比按量付费，预付费购买存储包有折扣，购买的容量越大，折扣力度就越大。

存储包和按量付费的价格对比

下表为按月计费存储包和按量付费的价格对比。存储容量越大，使用存储包越经济。

容量 (GB)	中国大陆		中国香港及海外	
	按量付费 (元/月)	存储包 (元/月)	按量付费 (元/月)	存储包 (元/月)
100	350	350	390	390
500	1,750	1,750	1,950	1,950
1,000	3,500	3,150	3,900	3,510
3,000	10,500	9,450	11,700	10,530
5,000	17,500	15,750	19,500	17,550
10,000	35,000	31,500	39,000	35,100
30,000	105,000	84,000	117,000	93,600
50,000	175,000	140,000	195,000	156,000
100,000	350,000	280,000	390,000	312,000

注意事项

- 每种类型资源包只允许购买一个，存储包容量不够时，可以[升级存储包](#)。
- 存储包暂不支持降级。
- 存储包由资源包类型规定的地域内的所有集群共享使用。
- 超出存储包容量的部分以按量付费的方式收取费用。例如购买了1000G的存储包，有三个POLARDB集群，存储容量分别是300G、400G和500G，那么 $300+400+500-1000=200$ G，存储包抵扣后，超出的200G需要以按量付费的方式收取费用，您可以在费用中心[查看存储包抵扣量和按量付费的存储量](#)。

**说明:**更多关于存储包的说明, 请参见[常见问题](#)。

购买存储包

1. 登录[购买POLARDB存储包](#)页面。
2. 设置如下参数。

参数	说明
资源包类型	- 中国大陆通用: 购买后可用于中国大陆地域的POLARDB集群。 - 中国香港及海外通用: 购买后可用于中国香港及海外地域的POLARDB集群。
存储包规格	存储包的容量大小。
购买时长	购买存储包的时长。

3. 单击立即购买。
4. 勾选服务协议, 单击去支付完成支付。

查看存储包抵扣量和按量付费的存储量

1. 登录[费用中心](#)。

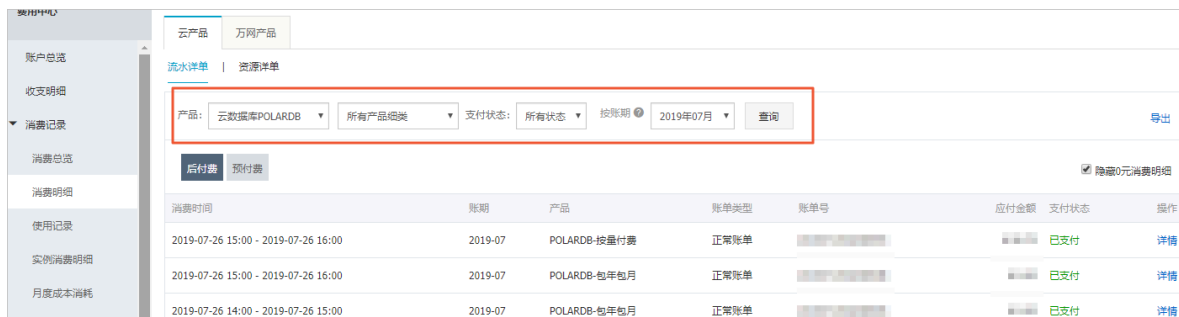
**说明:**

如果跳转到新版费用中心, 请在左侧导航栏单击返回旧版。



2. 在左侧导航栏中，选择消费记录 > 消费明细。

3. 设置搜索条件查询POLARDB相关账单，单击查询。



4. 单击目标账单右侧详情。

5. 展开费用详单查看抵扣详情。



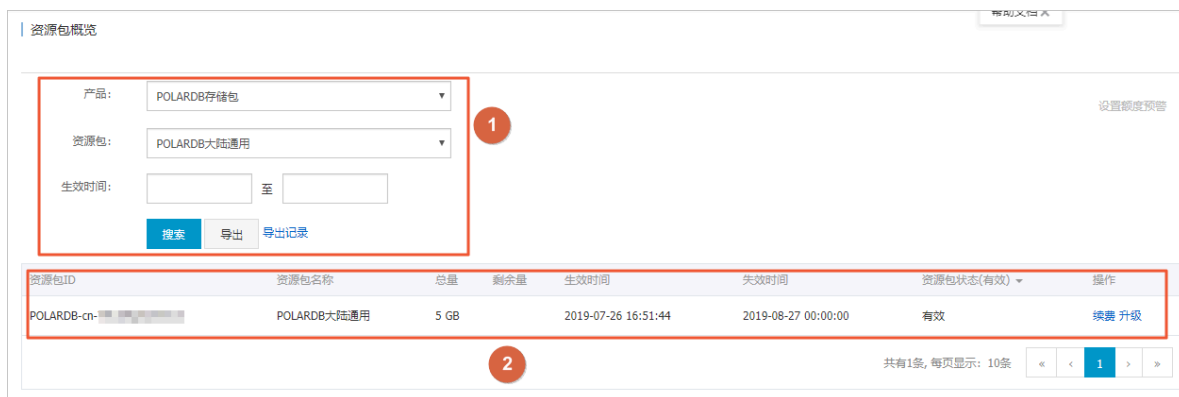
续费/升级存储包

1. 登录费用中心。



2. 在左侧导航栏中, 选择资源包管理 > 资源概览。

3. 在产品栏选择POLARDB存储包, 设置搜索条件, 单击搜索。



4. 您可以按如下方法进行续费或升级操作：

- 续费

- a. 单击操作列的续费。

- b. 选择续费时长，勾选服务协议，单击去支付完成支付。

- 升级

- a. 单击操作列的升级。

- b. 选择存储包规格，勾选服务协议，单击去支付完成支付。

常见问题

- 存储包是否跟集群绑定售卖？

答：不绑定。您需要单独购买存储包，购买后会自动抵扣相应地域内的集群存储空间。

- 存储包是否可以被多个集群共享？

答：可以。存储包由资源包类型（中国大陆或中国香港及海外）规定的地域内的所有集群共享使用。

- 存储包是否可以被不同引擎的集群共享？

答：可以。存储包可以同时用于POLARDB for MySQL/PostgreSQL/Oracle的集群。

- 存储包容量不够了怎么办？可以再买一个吗？

答：每种类型资源包只允许购买一个。存储包容量不够时可以进行[升级](#)。

- 当前数据量超出存储包容量的部分如何计费？

答：超出存储包容量的部分以按量付费的方式收取费用，详情请参见[#unique_20/unique_20_Connect_42_section_u9d_n9d_3jt](#)。

- 存储包支持3TB和5TB，业务只需要4TB，如何选购？

答：您可以先购买3TB，存储量接近5TB时再升级为5TB。

5 待处理事件

当POLARDB出现待处理事件时，会在控制台提醒您及时处理。

前提条件

您有未处理的运维事件。



说明:

当您有未处理的运维事件时，可以在控制台左侧导航栏的待处理事件看到提醒。

云数据库POLAR...

待处理事件

集群列表

待处理事件

历史事件

数据库软件升级

硬件维护与升级

尊敬的用户，为了给您提供更出色的性能和稳定性，POLARDB会定期进行数据库软件升级，修复缺陷、优化性能或发布新的功能。

每次升级不超过1个小时，升级中每个连接地址都会有不超过30秒的连接闪断，请确保应用具备重连机制。升级过程中，控制台上的部分变更类功能（比如升降配置、增减节点、修改参数、重启等）均暂无法使用，但查询类功能（如性能监控等）不受影响。

修改切换时间

☐	任务ID	集群名称	兼容数据库	开始时间	切换时间	最晚开始时间
☐	58771	pc-xxxxxx	MySQL 5.6	2019-05-24 02:10:00	2019-05-24 02:10:00	2019-05-31 23:59:59

每页显示: 10 < 上一页 1 下一页 >

背景信息

POLARDB运维事件（数据库软件升级、硬件维护与升级）除了在短信、语音、邮件或站内信通知之外，还会在控制台进行通知。您可以查看具体的事件类型、任务ID、集群名称、切换时间等，也可以手动修改切换时间。

修改切换时间

1. 登录POLARDB控制台。

2. 在左侧导航栏中，单击待处理事件。



说明：

强制要求预约时间的运维事件会弹窗提醒，请尽快完成预约。

3. 在待处理事件页面，选择相应的事件类型。



说明：

不同的事件类型页面会显示不同的通知信息。

- 在下方事件列表查看事件的详细信息，如需修改切换时间，请在左侧勾选对应的实例，然后单击修改切换时间，在弹出的对话框中设置时间并单击确认。



说明:

切换时间不能晚于最晚操作时间。

历史事件

您可以在POLARDB控制台里查看已完成的事件。

- 登录POLARDB控制台。
- 在左侧导航栏中，单击历史事件。



- 在历史事件页面，选择相应的事件类型。不同的事件类型页面会显示不同的信息。

6 设置集群白名单

创建POLARDB for MySQL数据库集群后，请把需要访问数据库集群的服务器IP地址添加到白名单，并创建集群的初始账号，才能连接和使用该集群。

背景信息

只有已添加到白名单中的IP地址才可以访问集群。

注意事项

- 默认情况下，白名单只包含IP地址127.0.0.1，表示任何IP地址均无法访问该数据库集群。
- 若将白名单设置为%或者0.0.0.0/0，表示允许任何IP地址访问数据库集群。该设置将极大降低数据库的安全性，如非必要请勿使用。
- POLARDB暂不支持自动获取VPC中的ECS内网IP以供您选择，请手动填写需要访问POLARDB的ECS内网IP。

设置白名单

1. 进入[POLARDB控制台](#)。
2. 单击目标集群ID，进入基本信息页面。
3. 在访问信息区域，可以配置已有白名单或新增白名单分组。



- 单击白名单分组右侧的配置，配置当前分组的白名单。
 - 单击新增白名单分组，可以新增白名单分组。
4. 填写白名单，单击提交。
- 如果您的ECS服务器需要访问POLARDB，可在ECS实例详情页面配置信息区域，查看ECS服务器的IP地址，然后填写到白名单中。



说明:

如果ECS与POLARDB位于同一地域（例如，华东1），填写ECS的私网IP地址；如果ECS与POLARDB位于不同的地域，填写ECS的公网IP地址，或者将ECS迁移到POLARDB所在地域后填写ECS私网IP地址。

- 如果您本地的服务器、电脑或其它云服务器需要访问POLARDB，请填写其IP地址到白名单中。

下一步

[#unique_7](#)

[#unique_22](#)

FAQ

1. 如何指定服务器只能访问集群中的某个节点？

答：使用[自定义集群地址](#)访问集群，就只能访问集群中的特定节点。

2. 白名单可以支持填写更多的IP地址吗？

答：暂不支持，您可以填写IP网段实现该需求。

3. 已添加ECS的IP地址到白名单中，但是还是无法访问。

答：

- 确认白名单是否正确。如果是通过内网地址访问，需添加ECS的私网IP地址。如果是通过公网地址进行访问，需添加ECS的公网IP地址。
- 确认网络类型是否一致。如果ECS实例的网络类型是经典网络，可参考[经典网络迁移到专有网络方案](#)将ECS实例迁移至POLARDB所在的专有网络。



说明：

如果该ECS 还要访问其他经典网络内网资源，请勿操作，因为迁移后会无法访问经典网络。

或者通过[Classlink](#)打通经典网络到专有网络的网络。

- 确认是否位于同一个VPC。如果不是，需要重新购买一个POLARDB，或者通过[云企业网](#)来打通两个VPC网络实现访问。

4. 什么原因导致公网连接失败？

答：

- 如果是ECS通过公网地址进行访问，请确认添加的是ECS的公网IP地址，而不是私网IP地址。
- 白名单设置为0.0.0.0/0，然后尝试访问，如果能成功访问，表示白名单中之前填写的公网地址错误。请参考[#unique_24](#)确定真正的公网地址。

5. 如何实现内网连接？

答：ECS和POLARDB内网访问需要满足以下条件：

- 相同地域。
- 相同网络类型，如果是VPC，需要在相同VPC下。
- ECS内网IP在POLARDB集群白名单中。

6. 如何限制某个用户只能从特定的IP地址访问POLARDB？

答：可以创建高权限账号，然后使用高权限账号对普通账号限定访问IP。

```
1  
2  
3 CREATE USER 'alitest'@'192.168.1.101' ;  
4  
5  
6 select * from mysql.user where user='alitest';|
```

相关API

API	描述
#unique_25	查看允许访问数据库集群的IP名单。
#unique_26	修改允许访问数据库集群的IP名单。

7 计费管理

7.1 按小时付费转包年包月

您可以根据需求将后付费（按小时付费）的集群转变为预付费（包年包月）的计费方式。

前提条件

- 集群状态为运行中。
- 集群没有未完成的按小时付费转包年包月的订单。如果有，需要先在[订单管理](#)页面支付或作废该订单。

背景信息

按小时付费转包年包月操作对集群的运行不会有任何影响。



说明:

历史规格不支持直接转包年包月，请先[变更配置](#)，然后再转包年包月。

注意事项

包年包月的集群无法转成按小时付费的集群。在您将集群转为包年包月前请务必考虑清楚，以免造成资源浪费。

操作步骤

1. 登录[POLARDB控制台](#)。
2. 在控制台左上角，选择集群所在地域。
3. 找到目标集群，在操作列中选择… > 转包年包月。

云数据库POLARDB... 集群列表							
集群列表							
集群名称	状态	兼容数据库	节点数	主节点配置	已使用数据	付费类型	操作
pc-xxxxxx	运行中	MySQL 5.6	3	4核 32GB	2.76 GB	按小时付费 2018年12月10日 16:56:41 创建	升配 降节点 ...
pc-xxxxxx	运行中	MySQL 5.6	2	4核 16GB	2.77 GB	按小时付费 2018年12月7日 15:11:12 创建	降配
pc-xxxxxx	运行中	MySQL 5.6	3	2核 4GB	2.78 GB	包年包月 2018年12月21日 00:00:00 到期	续费到期集群
pc-xxxxxx	运行中	MySQL 5.6	3	4核 32GB	2.80 GB	按小时付费 2018年11月1日 11:49:52 创建	转包年包月

4. 选择购买时长，勾选云数据库POLARDB服务协议，单击去开通，根据提示完成支付。



说明:

- 本操作会产生一个新购订单，只有完成了订单的支付，计费方式的变更才能生效。

- 若未支付或未成功支付，您的[订单管理](#)页面将会出现未完成订单，导致您无法新购集群或再次执行转包年包月。此时需支付或作废该订单。

7.2 手动续费集群

本文介绍如何通过POLARDB控制台或续费管理控制台进行手动续费。

背景信息

您可以通过POLARDB控制台或续费管理控制台进行手动续费。通过续费管理控制台还可以为多个实例进行批量续费。



说明：

按小时付费集群没有到期时间，不涉及续费操作。

方法一：POLARDB控制台续费

1. 登录[POLARDB控制台](#)。
2. 在控制台左上角，选择集群所在地域。
3. 找到目标集群，在右侧选择… > 续费。

集群名称	状态	兼容数据库	节点数	主节点配置	已使用数据	付费类型	操作
pc-xxxxxx	运行中	MySQL 5.6	3	2核 4GB	2.74 GB	包年包月 2019年1月20日 00:00:00 到期	升级 节点 ...
pc-xxxxxx	创建中	MySQL 5.6	2	2核 4GB	-	按小时付费 2019年12月10日 16:56:14 创建	续费
pc-xxxxxx	变更配置中	MySQL 5.6	2	4核 16GB	2.75 GB	按小时付费 2019年12月5日 18:59:16 创建	续费

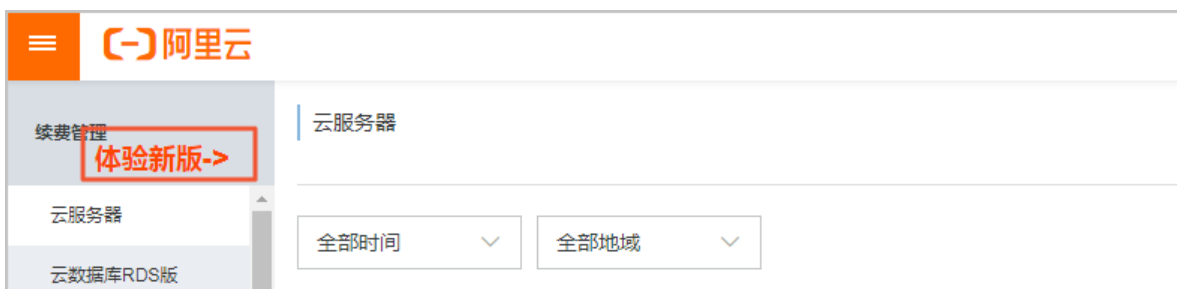
4. 选择续费时长并勾选服务协议，单击去支付完成支付即可。

方法二：续费管理控制台续费

1. 登录[POLARDB控制台](#)。
2. 在控制台右上方，选择费用 > 续费管理。

集群名称	状态	兼容数据库	节点数	主节点配置	已使用数据	操作
xxxxxx	运行中	Oracle	2	2核 8GB	8.11 GB	续费管理
xxxxxx	运行中	PostgreSQL 11	2	2核 8GB	5.90 GB	续费管理

3. 在控制台左上角单击体验新版，切换到新版控制台。



4. 通过搜索过滤功能在手动续费页签中找到目标集群，您可以单个续费或批量续费：

• 单个续费

a. 在目标集群右侧，单击续费。



说明：

- 示例为新版续费管理控制台操作步骤，如果您使用旧版控制台，需要在左侧导航栏中找到云数据库POLARDB，然后进行续费操作。

- 如果目标集群在自动续费或到期不续费页签中，您可以单击恢复手动续费，在弹出的对话框中单击确定即可恢复为手动续费。

b. 选择续费时长并勾选服务协议，单击去支付完成支付即可。

· 批量续费

a. 勾选目标集群，单击下方批量续费。

b. 选择每个集群的续费时长，单击去支付完成支付即可。

自动续费

开通自动续费可以免去您定期手动续费的烦恼，且不会因为忘记续费而导致业务中断。详情请参见[#unique_31](#)。

7.3 自动续费集群

本文为您介绍如何为集群开通自动续费，免去您手动续费的烦恼。

背景信息

包年包月集群有到期时间，如果到期未续费，会导致业务中断甚至数据丢失。开通自动续费可以免去您定期手动续费的烦恼，且不会因为忘记续费而导致业务中断。



说明：

按小时付费集群没有到期时间，不涉及续费操作。

注意事项

- 自动续费将于集群到期前9天开始扣款，支持现金及代金券扣款，请保持账户余额充足。
- 若您在自动扣款日期前进行了手动续费，则系统将在下一次到期前进行自动续费。
- 自动续费功能于次日生效。若您的集群将于次日到期，为避免业务中断，请手动进行续费，详细步骤请参见[#unique_33](#)。

购买集群时开通续费



说明:

开通自动续费后，系统将根据您的购买时长为周期进行自动续费。例如，如果您购买了3个月的集群并勾选了自动续费，则每次自动续费时会缴纳3个月的费用。

在创建新集群时，可以勾选自动续费。

购买集群后开通自动续费



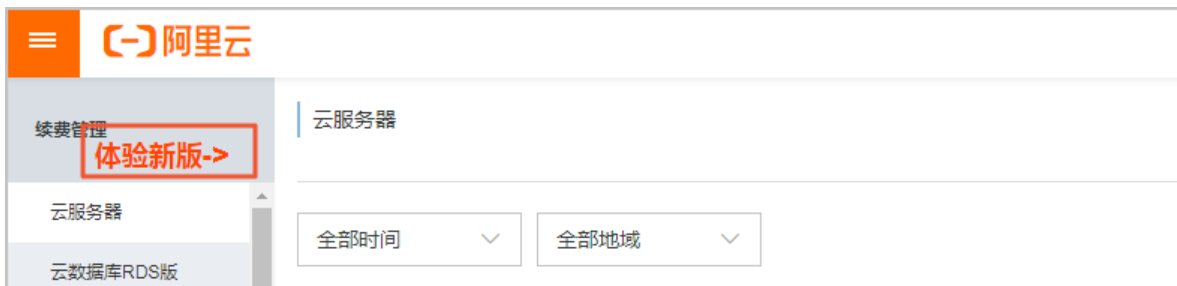
说明:

开通自动续费后，系统将根据您选择的续费周期进行自动续费。例如，如果您选择了3个月的续费周期，则每次自动续费时会缴纳3个月的费用。

1. 登录[POLARDB管理控制台](#)。
2. 在控制台右上方，选择费用 > 续费管理。



3. 在控制台左上角单击体验新版，切换到新版控制台。



4. 通过搜索过滤功能在手动续费或到期不续费页签中找到目标集群，您可以单个开通或批量开通：

• 单个开通

a. 单击右侧开通自动续费。



说明:

示例为新版续费管理控制台操作步骤，如果您使用旧版控制台，需要在左侧导航栏中找到云数据库POLARDB，然后开通自动续费。

b. 在弹出的对话框中，选择自动续费周期，单击开通自动续费。



· 批量开通

a. 勾选目标集群，单击下方开通自动续费。



b. 在弹出的对话框中，选择自动续费周期，单击开通自动续费。



修改自动续费周期

1. 登录POLARDB管理控制台。
2. 在控制台右上方，选择费用 > 续费管理。



3. 在控制台左上角单击体验新版，切换到新版控制台。

4. 通过搜索过滤功能在自动续费页签中找到目标集群，单击右侧修改自动续费。



说明：

示例为新版续费管理控制台操作步骤，如果您使用旧版控制台，需要在左侧导航栏中找到云数据库POLARDB，然后修改自动续费。

5. 在弹出的对话框中，修改自动续费周期后，单击确定。

关闭自动续费

1. 登录POLARDB管理控制台。

2. 在控制台右上方，选择费用 > 续费管理。



3. 在控制台左上角单击体验新版，切换到新版控制台。

4. 通过搜索过滤功能在自动续费页签中找到目标集群，单击右侧恢复手动续费。



说明：

示例为新版续费管理控制台操作步骤，如果您使用旧版控制台，需要在左侧导航栏中找到云数据库POLARDB，然后关闭自动续费。

5. 在弹出的对话框中，单击确定。

相关API

API	描述
#unique_34	创建数据库集群。  说明： 创建集群时开通自动续费。
#unique_35	设置包年包月集群自动续费状态。  说明： 创建集群后开通自动续费。
#unique_36	查询包年包月集群自动续费状态。

8 连接数据库集群

8.1 查看连接地址

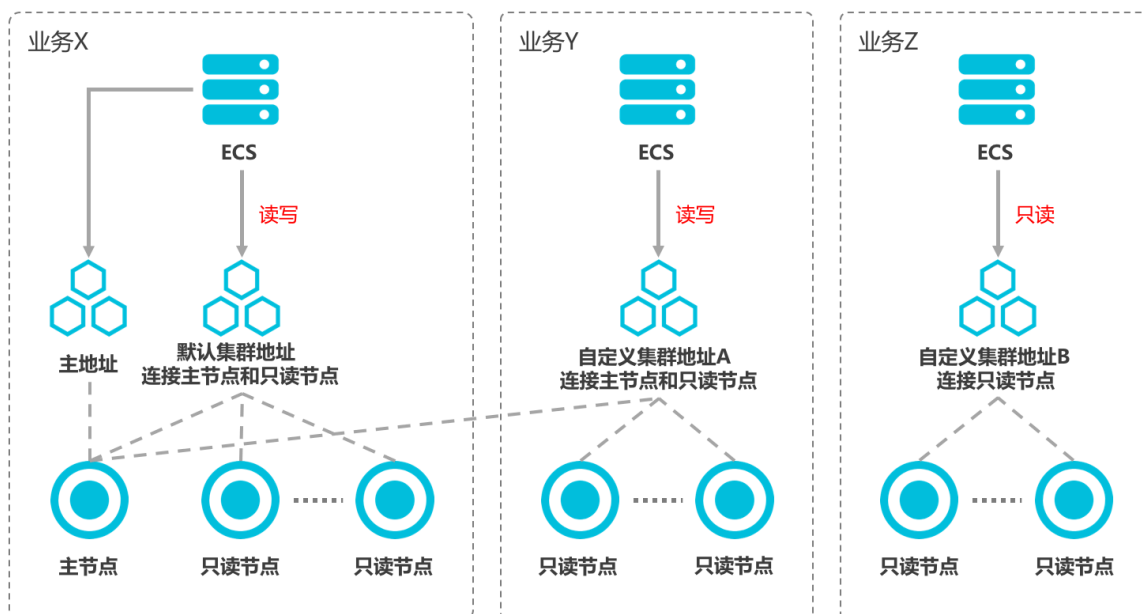
POLARDB for MySQL数据库连接地址包括集群访问地址和主访问地址。

查看方法

1. 登录[POLARDB控制台](#)。
2. 在控制台左上角，选择集群所在地域。
3. 单击目标集群的ID。
4. 在基本信息页面的访问信息区域，查看连接地址。

集群地址和主地址

地址类型	地址说明	支持的网 络类型	查看方式
集群地址（推荐）	应用程序只需连接一个集群地址，即可连接到多个节点。带有读写分离功能，写请求会自动发往主节点，读请求会自动根据各节点的负载发往主节点或只读节点。  说明： POLARDB包含一个默认的集群地址，您还可以根据业务需求创建多个自定义的集群地址，自定义集群地址可以连接到指定的节点，以及设置读写模式等。详情请参见#unique_39。	私网和公网	1. 找到目标集群，单击集群的ID。 2. 在访问信息区域查看。
主地址	总是连接到主节点，支持读和写操作。当主节点发生故障时，主访问地址会自动切换到新的主节点。	私网和公网	



私网地址和公网地址

地址类型	说明	使用场景
私网地址	- 通过私网地址访问可以发挥POLARDB的最佳性能。 - 私网地址无法被释放。	例如： - ECS与数据库集群位于同一VPC，那么ECS可以通过私网地址访问数据库集群。 - 使用DMS通过VPC访问数据库集群
公网地址	- 需手动申请公网的连接地址，也可以释放公网的连接地址。 - 公网即因特网。通过公网访问将无法实现POLARDB最佳性能。	例如：通过公网访问数据库集群进行维护操作。

下一步

连接数据库集群

相关API

API	描述
#unique_13	查询集群的地址信息。
#unique_11	创建集群的公网地址。
#unique_15	修改集群默认访问地址。

API	描述
#unique_16	释放集群地址。

8.2 设置/释放自定义集群地址

集群地址除了自动生成的默认地址，还可以手动新增自定义地址。

前提条件

2019年4月29日起创建的集群，可以直接新增自定义集群地址。2019年4月29日之前创建的集群需要[提交工单](#)申请，之后即可新增自定义集群地址。

背景信息

您可以在POLARDB for MySQL集群上使用自定义集群地址，并且可以设置它们的读写模式、一致性级别，以及选择关联的只读节点等等，方便应用于不同的业务场景，增强了业务的灵活性。



说明:

- 一个集群最多可以有4个集群地址（1个默认地址+3个自定义地址）。
- 默认的集群地址无法释放。自定义的集群地址可以被释放。
- 默认的集群地址也可以和自定义地址一样进行自定义设置，具体注意事项请参见[#unique_41/unique_41_Connect_42_section_6qz_cik_z5a](#)。

新增自定义集群地址

1. 登录 [POLARDB控制台](#)。
2. 在控制台左上角，选择集群所在地域。
3. 单击目标集群ID。

4. 在访问信息中的集群地址【推荐】后，单击新增自定义地址。

访问信息 ②

白名单 ②

白名单列表
配置

主地址 ②

私网

公网

申请

修改

集群地址【推荐】 ②

新增自定义地址

默认地址 () 修改配置

读写模式

可读可写 (自动读写分离)

私网

公网

申请

修改

> 节点配置

> 高级配置

5. 设置以下参数：

参数	说明
读写模式	本地地址的读写模式，分为只读和可读可写（自动读写分离）。  说明： 创建自定义地址后还可以修改读写模式。修改读写模式后，只对新建连接生效，已有的连接保持原来的模式。
读负载节点	在左侧选择想要加入本地地址用于处理读请求的节点。可选节点包括主节点和所有只读节点，本地地址只会将读请求发往被选中的节点。  说明： - 至少选择2个节点。 - 不论是否选中主节点，写请求都只会发往主节点。
新节点自动加入	新增的节点是否要自动添加到该地址中。
负载均衡策略	读写分离时，处理读请求的多个节点之间的调度策略，无需选择。

参数	说明
一致性级别	- 最终一致性：能够提供最好的性能。 - 会话一致性：提供Session级的读一致性保证，会略微增加主节点的负载。 具体请参见#unique_42。  说明： 如果读写模式为只读，只能选择最终一致性。

创建自定义地址

×

读写模式 ☐ 只读 ☒ 可读可写（自动读写分离）

节点配置

读负载均衡节点 ?

可选节点

☐ pi-

1 项

已选节点

☐ pi-
☐ pi-

2 项

>

<

节点的选择不影响读写模式。不论是否选中主节点，写操作都会发向主节点。

新节点自动加入 ? ☐ 开启 ☒ 关闭

高级配置

负载均衡策略 ? 基于负载的自动调度

一致性级别 ?

会话一致性 ▼

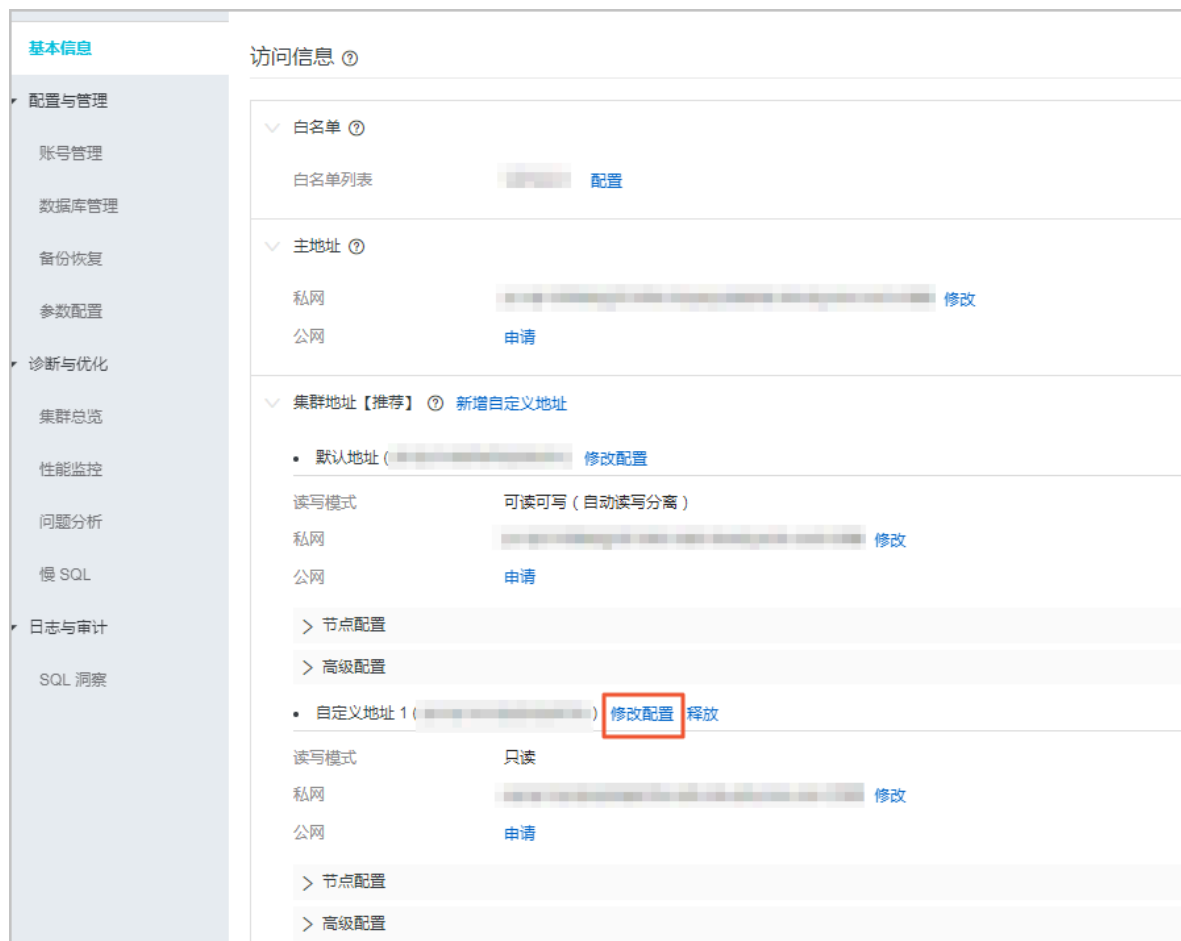
确认

取消

6. 单击确认。

修改自定义集群地址

1. 登录 [POLARDB控制台](#)。
2. 在控制台左上角，选择集群所在地域。
3. 单击目标集群ID。
4. 在访问信息中的自定义地址后，单击修改配置。



5. 设置以下参数：

参数	说明
读写模式	本地地址的读写模式，分为只读和可读可写（自动读写分离）。

参数	说明
读负载节点	在左侧选择想要加入本地地址用于处理读请求的节点。可选节点包括主节点和所有只读节点，本地地址只会将读请求发往被选中的节点。  说明: - 至少选择2个节点。 - 不论是否选中主节点，写请求都只会发往主节点。 - 增加节点时对该地址的使用无任何影响，剔除节点时，该节点上的长连接会中断。
新节点自动加入	新增的节点是否要自动添加到该地址中。
负载均衡策略	读写分离时，处理读请求的多个节点之间的调度策略，无需选择。
一致性级别	- 最终一致性：能够提供最好的性能。 - 会话一致性：提供Session级的读一致性保证，会略微增加主节点的负载。 具体请参见#unique_42。  说明: - 如果读写模式为只读，只能选择最终一致性。 - 一致性级别修改后对所有连接立即生效。

6. 单击确认。

释放自定义集群地址

1. 登录 [POLARDB控制台](#)。
2. 在控制台左上角，选择集群所在地域。
3. 单击目标集群ID。

4. 在访问信息中的集群地址【推荐】中找到目标自定义地址，单击释放。



5. 在弹出的对话框中，单击确认。

相关API

API	描述
#unique_12	创建自定义集群地址。
#unique_13	查询集群地址。
#unique_14	修改集群地址。
#unique_17	释放自定义集群地址。

8.3 连接数据库集群

本文将分别介绍如何使用DMS（Data Management Service）和通用MySQL客户端连接POLARDB for MySQL集群。

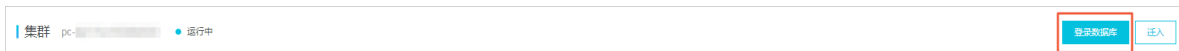
前提条件

已创建数据库集群的高权限账号或普通账号。具体操作请参见[创建数据库账号](#)。

使用DMS连接集群

DMS是阿里云提供的图形化的数据管理工具，它是一种集数据管理、结构管理、访问安全、BI图表、数据趋势、数据轨迹、性能与优化和服务管理于一体的数据管理服务。支持对关系型数据库（MySQL、SQL Server、PostgreSQL等）和NoSQL数据库（MongoDB、Redis等）的管理，同时还支持Linux服务器管理。

1. 找到目标集群，单击集群ID，进入基本信息页面。
2. 单击页面右上角的登录数据库。



3. 在数据库登录页面，输入主地址和端口号（以英文冒号隔开），以及高权限账号或普通账号的用户名和密码，然后单击登录。



说明：

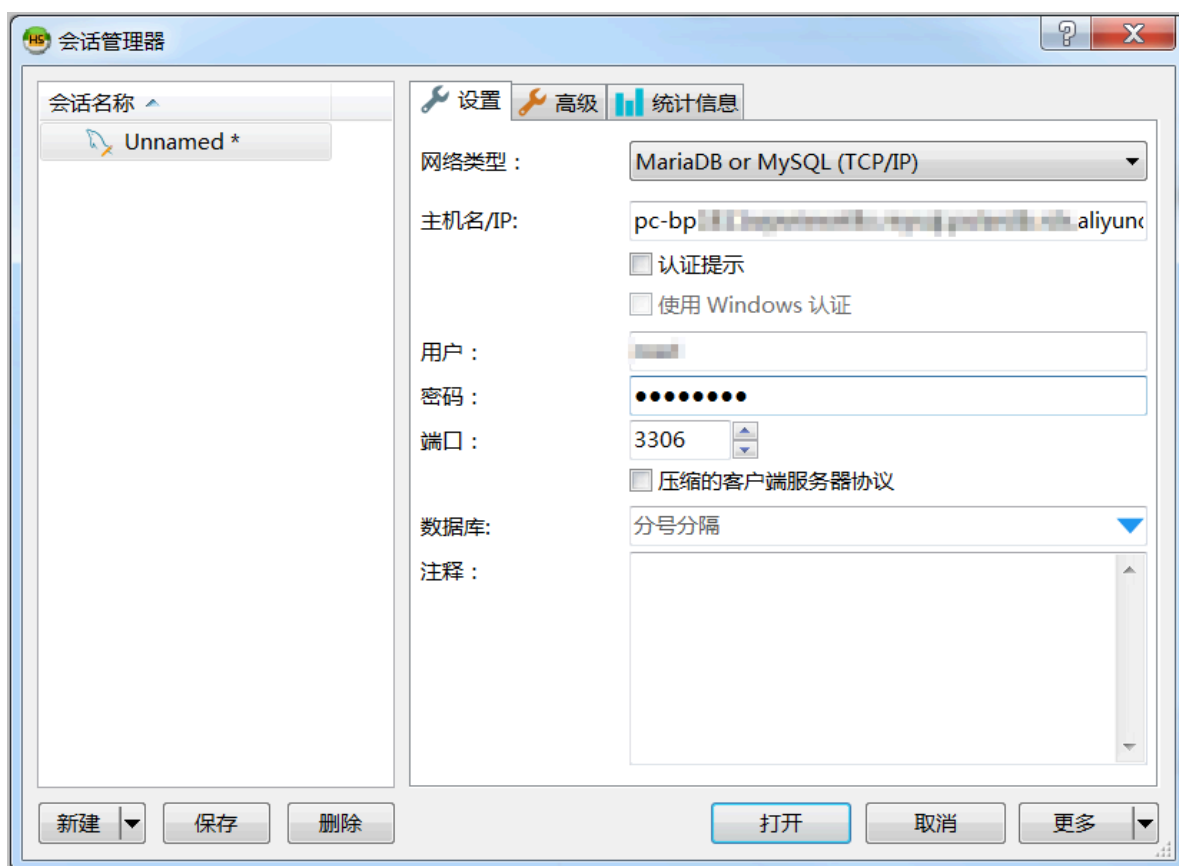
DMS登录仅支持主地址，不支持集群地址。关于如何查看连接地址，请参见[查看连接地址](#)。

使用客户端连接集群

您可以使用任何通用的客户端连接POLARDB集群。本文以HeidiSQL为例。

1. 启动HeidiSQL客户端。

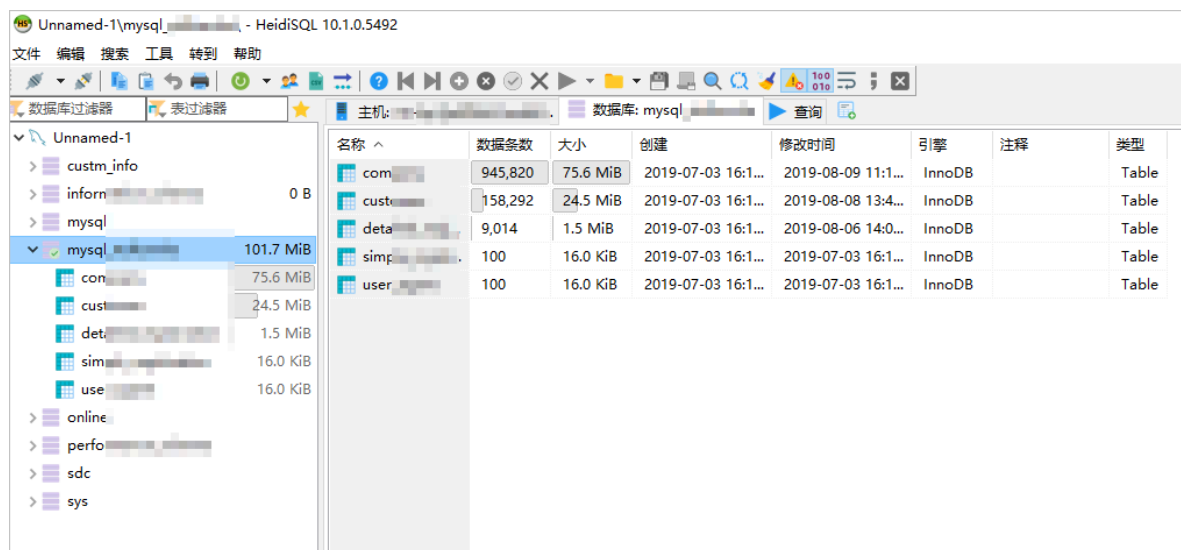
2. 在左下角单击新建。



3. 输入要连接的POLARDB集群信息，参数说明如下：

参数	说明
网络类型	连接数据库的形式。选择MariaDB or MySQL（TCP/IP）。
主机名/IP地址	输入POLARDB集群的私网地址或公网地址。 - 若您的客户端部署在ECS实例上，且ECS实例与要访问的POLARDB集群的地域、网络类型相同，请使用私网地址。例如ECS实例和POLARDB集群都是华东1的专有网络实例，使用私网地址连接能提供安全高效的访问。 - 其它情况只能使用公网地址。 查看POLARDB集群的私网、公网地址及端口信息的步骤如下： - 登录POLARDB控制台。 - 在页面左上角，选择集群所在地域。 - 找到目标集群，单击集群ID。 - 在基本信息页中，即可查看私网、公网地址及其端口信息。
用户	要访问POLARDB集群的账号名称。
密码	以上账号的密码。
端口	输入POLARDB集群的私网地址或公网地址对应的端口号。

4. 单击打开。若连接信息无误，即会成功连接集群。



使用命令行连接集群

如果您的服务器安装了MySQL，可以通过命令行连接POLARDB for MySQL数据库集群，连接方式如下：

```
mysql -h<连接地址> -P<端口> -u<用户名> -p<密码> -D<数据库>
```

参数	说明	示例
-h	POLARDB集群的私网网地址或公网地址。连接地址请参见 查看连接地址 。	pc-bpxxxxxxxxxxxxxxxxx.mysql.polardb.rds.aliyuncs.com
-P	POLARDB集群的端口号。 - 若使用私网连接，需输入POLARDB集群的私网端口。 - 若使用公网连接，需输入POLARDB集群的公网端口。  说明： - 默认端口为3306。 - 如果端口号为默认端口，该参数可以不填。	3306
-u	要访问POLARDB集群的账号名称。	root

参数	说明	示例
-p	以上账号的密码。  说明: 该参数非必填参数。 - 如果不填写该参数，后续操作中会重新要求输入密码。 - 如果填写该参数，-p与数据库密码之间不能有空格。	password233
-D	需要登录的数据库名称。  说明: 该参数非必填参数。 可以不输入-D仅输入数据库名称。	mysql

解决连接失败问题

- IP白名单设置有误。
 - 由于白名单设置中只有默认地址127.0.0.1。127.0.0.1表示禁止任何IP地址访问POLARDB集群。因此需在白名单中添加对端的IP地址，具体操作请参见[设置集群白名单](#)。
 - 白名单设置成了0.0.0.0，正确格式为0.0.0.0/0。



说明:

0.0.0.0/0表示允许任何设备访问POLARDB集群，请谨慎使用。

- 您在白名单中添加的设备公网IP地址可能并非设备真正的出口IP地址。例如，公网IP地址不固定，可能会变动。或者IP地址查询工具、网站查询的公网IP地址不准确。
- 私网、公网地址使用错误。

使用私网地址从公网进行连接，或者使用公网地址从私网进行连接，都会导致连接失败。

请确认您使用的地址类型。如果您需要从私网连接POLARDB集群，请使用POLARDB集群的私网地址；如果您需要从公网连接POLARDB集群，请使用POLARDB集群的公网地址。

- 网络类型不同，ECS实例采用经典网络，POLARDB集群采用专有网络（VPC）。
- 解决办法一（推荐）：将ECS实例从经典网络迁移到VPC，具体操作请参见[单ECS迁移示例](#)。



说明：

切换后，两者必须处于同一个VPC，才能内网互通。

- 解决办法二：使用[ClassicLink](#)功能，使经典网络的ECS实例可以和VPC中的云资源通过内网互通。
- 解决办法三：ECS实例使用POLARDB集群的公网连接地址连接POLARDB集群，即通过公网连接POLARDB集群。这种方式的性能、安全性、稳定性较差。

9 集群管理

9.1 创建POLARDB for MySQL数据库集群

本文介绍如何通过POLARDB管理控制台创建POLARDB for MySQL数据库集群。

前提条件

已注册阿里云账号或已创建子账号。

- 如要注册阿里云账号，请[点此注册](#)。
- 如要创建子账号，请参见[创建和使用子账号](#)进行子账号的创建和授权。

背景信息

一个集群包含一个主节点以及最多15个只读节点，只读节点最少一个，用于提供双主（Active-Active）高可用。节点是虚拟化的数据库服务器，节点中可以创建和管理多个数据库。



说明：

- POLARDB仅支持专有网络VPC（Virtual Private Cloud）。VPC是阿里云上一种隔离的网络环境，安全性比传统的经典网络更高。
- POLARDB与其他阿里云产品通过内网互通时才能发挥POLARDB的最佳性能，因此，建议将POLARDB与云服务器ECS配合使用，且与ECS创建于同一个VPC，否则POLARDB无法发挥最佳性能。如果您ECS的网络类型为经典网络，需将ECS从经典网络迁移到VPC，具体请参见[ECS实例迁移](#)。

优惠活动

首购折扣价：首次购买POLARDB享受折扣价。详情请参见[优惠活动](#)。

操作步骤

1. 登录阿里云。

- 如果使用阿里云账号，请[点此登录](#)。
- 如果使用子账号，请[点此登录](#)。更多信息请参见[创建和使用子账号](#)。

2. 进入[POLARDB购买页面](#)。

3. 选择包年包月或按量付费。



说明：

- 包年包月：在创建集群时支付计算节点（一个主节点和一个只读节点）的费用，而存储空间会根据实际数据量按小时计费，并从账户中按小时扣除。如果您要长期使用该集群，包年包月方式更为经济，而且购买时长越长，折扣越多。
- 按量付费（按小时付费）：无需预先支付费用，计算节点和存储空间（根据实际数据量）均按小时计费，并从账户中按小时扣除。如果您只需短期使用该集群，可以选择按量付费，用完即可释放，节省费用。

4. 设置如下参数。

控制台区域	参数	说明
基本配置	地域	集群所在的地理位置。购买后无法更换地域。  说明： 请确保POLARDB与需要连接的ECS创建于同一个地域，否则它们无法通过内网互通，只能通过外网互通，无法发挥最佳性能。
	创建方式	创建POLARDB集群的方式。 · 默认创建：创建一个全新的POLARDB。 · 从RDS克隆：基于所选的RDS实例，快速克隆一个数据完全一样的POLARDB集群。详情请参见#unique_47。 - 源RDS引擎：源RDS实例的引擎类型，不可变更。 - 源RDS版本：源RDS实例的版本，不可变更。 - 源RDS实例：可选的源RDS实例，不包括只读实例。 · 从RDS迁移：先从RDS克隆数据，然后保持增量同步，主要用于迁移。在正式迁移切换前POLARDB为只读，且默认开启Binlog。详情请参见#unique_48。 - 源RDS引擎：源RDS实例的引擎类型，不可变更。 - 源RDS版本：源RDS实例的版本，不可变更。 - 源RDS实例：可选的源RDS实例，不包括只读实例。
	可用区	集群的主可用区。 · 可用区是地域中的一个独立物理区域，不同可用区之间没有实质性区别。 · 您可以选择将POLARDB与ECS创建在同一可用区或不同的可用区。 · 您只需要选择主可用区，系统会自动选择备可用区。
	网络类型	· 无需选择。 · 仅支持专有网络VPC（Virtual Private Cloud）。VPC是一种隔离的网络环境，安全性和性能均高于传统的经典网络。

控制台区域	参数	说明
	VPC网络 VPC交换机	请确保POLARDB与需要连接的ECS创建于同一个VPC，否则它们无法通过内网互通，无法发挥最佳性能。 - 如果您已创建符合您网络规划的VPC，直接选择该VPC。 例如，如果您已创建ECS，且该ECS所在的VPC符合您的规划，那么选择该VPC。 - 如果您未创建符合您网络规划的VPC，您可以使用默认VPC和交换机： - 默认VPC： ■ 在您选择的地域中是唯一的。 ■ 网段掩码是16位，如172.31.0.0/16，最多可提供65536个私网IP地址。 ■ 不占用阿里云为您分配的VPC配额。 - 默认交换机： ■ 在您选择的可用区中是唯一的。 ■ 网段掩码是20位，如172.16.0.0/20，最多可提供4096个私网IP地址。 ■ 不占用VPC中可创建交换机的配额。 - 如果以上默认VPC和交换机无法满足您的要求，您可以自行创建VPC和交换机。
配置	数据库引擎	- 兼容MySQL 8.0（完全兼容）。原生支持并行查询，特定场景下（TPC-H测试）性能提升十倍，详情请参见并行查询。 - 兼容MySQL 5.6（完全兼容）。 - 兼容PostgreSQL 11（完全兼容）。 - 兼容Oracle（高度兼容）。
	节点规格	按需选择。所有POLARDB节点均为独享型，性能稳定可靠。关于各规格的具体信息，请参见 #unique_20 。
	节点个数	- 无需选择。系统将自动创建一个与主节点规格相同的只读节点。 - 如果主节点故障，系统会自动将只读节点切换为新的主节点，并重新生成一个只读节点。 - 关于只读节点的更多信息，请参见#unique_51。
	存储费用	无需选择。系统会根据实际数据使用量按小时计费，详情请参见产品价格。  说明： 创建集群时无需选择存储容量，存储容量随数据量的增减而自动弹性伸缩。

5. 设置购买时长（仅针对预付费集群）和集群数量，然后单击右侧的立即购买。



说明:

最多可以一次性创建50个集群，适用于游戏批量开服等业务场景。

6. 在确认订单页面，确认订单信息，阅读和勾选服务协议，单击去支付。

完成支付后，集群将在十分钟左右创建成功，您可在集群列表中看到创建的集群。



说明:

- 当集群中的节点状态为运行中时，整个集群可能仍未创建完成，此时集群不可用。只有当集群状态为运行中时，集群才可以正常使用。
- 请确认已选中正确的地域，否则无法看到您创建的集群。
- 当您的数据量较大时，推荐您购买POLARDB**存储包**，相比按小时付费，预付费购买存储包有折扣，购买的容量越大，折扣力度就越大。

下一步

设置集群白名单

相关API

API	描述
#unique_34	创建数据库集群。
#unique_52	查看集群列表。
#unique_53	查看指定POLARDB集群的详细属性。
#unique_36	查询POLARDB包年包月集群自动续费状态。
#unique_35	设置POLARDB包年包月集群自动续费状态。

9.2 增加或删除节点

创建POLARDB集群后，您可以手动增加或删除只读节点。

背景信息

集群最多包含15个只读节点，最少一个只读节点（用于保障集群的高可用）。同一集群中，所有节点的规格总是保持一致。

节点费用

增加节点时的计费方式如下：

- 如果集群为包年包月（预付费），则增加的节点也是包年包月。
- 如果集群为按小时付费（后付费），则增加的节点也是按小时付费。

**说明:**

- 包年包月和按小时付费的只读节点都可以随时释放，释放后会[退款或停止计费](#)。
- 增加节点仅收取节点规格的费用（详情请参见[#unique_20](#)），存储费用仍然按实际使用量收费，与节点数量无关。

节点数量对性能的影响

详情请参见[POLARDB for MySQL性能白皮书](#)。

注意事项

- 仅当集群没有正在进行的配置变更时，才可以增加或删除只读节点。
- 为避免操作失误，每次操作只能增加或删除一个只读节点，增加或删除多个只读节点请多次操作。
- 增加或删除节点需要5分钟左右生效。

增加只读节点

**说明:**

新增只读节点之后新建的读写分离连接会转发请求到该只读节点。新增只读节点之前建立的读写分离连接不会转发请求到新增的只读节点，需要断开该连接并重新建立连接，例如，重启应用。

1. 登录[POLARDB控制台](#)。
2. 在控制台左上角，选择集群所在地域。

3. 进入增删节点向导页面。您可以按照如下两种方式操作：

- 找到目标集群，在操作列单击增删节点。

集群列表							
POLARDB简介							
创建新集群	集群ID	请输入	Q	刷新		下载	
集群名称	状态	兼容数据库	节点数	主节点配置	已使用数据	付费类型	操作
pc-xxxxxx	运行中	MySQL 5.6	2	2核 4GB	2.83 GB	包年包月 2019年8月15日 00:00:00 到期	升降配 增删节点

- 找到目标集群，单击集群ID，在基本信息页面下方单击增删节点。

基本信息

配置与管理

账号管理

数据库管理

备份恢复

参数配置

诊断与优化

集群总览

性能监控

问题分析

慢 SQL

日志与审计

SQL 洞察

集群地址【推荐】 ② 新增自定义地址

默认地址 (pe-) 修改配置

读写模式 可读可写 (自动读写分离)

私网 pc- 修改

公网 申请

> 节点配置

> 高级配置

节点信息

增删节点

升降配

节点名称	可用区	状态	当前角色	规格	最大IOPS	操作
pi-	杭州 可用区G	运行中	主节点	2核 4GB	8000	重启
pi-	杭州 可用区G	运行中	只读节点	2核 4GB	8000	重启

4. 勾选增加节点并单击确定。

增删节点向导

您当前的付费方式为 **包年包月**, 支持以下配置变更方案：

☒ **增加节点**

支持您在当前生命周期内立即增加POLARDB的数据库计算节点，增加一个节点大概需要5分钟，整个过程对数据库无任何影响。使用默认集群地址可自动识别新节点，自动分流到新节点，达到负载均衡，不需要修改应用配置。参考文档：[增加节点](#), [包年包月集群增加节点的计费规则](#)

☐ **删除节点**

支持您在当前生命周期内立即删除POLARDB的数据库计算节点，该节点上的连接会发生闪断，其他节点不受影响。使用集群地址可自动屏蔽失效节点，不需要修改应用配置。参考文档：[删除节点](#), [包年包月集群删除节点的退费规则](#)

确定

取消

5. 单击 增加一个只读节点，勾选服务协议，单击去支付并完成支付。

删除只读节点

1. 登录POLARDB控制台。
2. 在控制台左上角，选择集群所在地域。
3. 进入增删节点向导页面。您可以按照如下两种方式操作：
 - 找到目标集群，在操作列单击增删节点。



集群列表								POLARDB简介	
创建新集群		集群ID	请输入	Q	刷新		下载		
集群名称	状态	兼容数据库	节点数	主节点配置	已使用数据	付费类型	操作		
pc-xxxxxx	运行中	MySQL 5.6	2	2核 4GB	2.83 GB	包年包月 2019年8月15日 00:00:00 到期	升降配 增删节点		

- 找到目标集群，单击集群ID，在基本信息页面下方单击增删节点。



基本信息	集群地址【推荐】 新增自定义地址
	默认地址 (pe-xxxxxx) 修改配置
	读写模式 可读可写 (自动读写分离)
	私网 pc-xxxxxx 修改
	公网 申请
配置与管理	节点配置
	高级配置
诊断与优化	节点信息
	增删节点 升降配
	节点名称 可用区 状态 当前角色 规格 最大IOPS 操作
日志与审计	ph-xxxxxx 杭州 可用区G 运行中 主节点 2核 4GB 8000 重启
	ph-xxxxxx 杭州 可用区G 运行中 只读节点 2核 4GB 8000 重启

4. 勾选删除节点并单击确定。

×

增删节点向导

您当前的付费方式为 **包年包月**，支持以下配置变更方案：

☐ 增加节点

支持您在当前生命周期内立即增加POLARDB的数据库计算节点，增加一个节点大概需要5分钟，整个过程对数据库无任何影响。使用默认集群地址可自动识别新节点，自动分流到新节点，达到负载均衡，不需要修改应用配置。参考文档：[增加节点](#)，[包年包月集群增加节点的计费规则](#)

☒ 删除节点

支持您在当前生命周期内立即删除POLARDB的数据库计算节点，该节点上的连接会发生闪断，其他节点不受影响。使用集群地址可自动屏蔽失效节点，不需要修改应用配置。参考文档：[删除节点](#)，[包年包月集群删除节点的退费规则](#)

确定

取消

5. 单击想要删除的节点后面的，并在弹出对话框中单击确定。



说明：

集群中必须保留至少一个只读节点，以保障集群的高可用。

6. 勾选服务协议，单击确认。

相关API

API	描述
#unique_57	增加POLARDB集群节点
#unique_58	变更POLARDB集群节点规格
#unique_59	重启POLARDB集群节点
#unique_60	删除POLARDB集群节点

9.3 变更配置

本文介绍如何升级或降级集群的规格，新规格会立即开始生效（每个节点需要5到10分钟）。您可以根据业务需求变更集群的配置。

前提条件

集群没有正在进行的配置变更时，才可以变更集群规格。

背景信息

您可以根据业务需求变更集群的配置。POLARDB支持三维扩展能力：

- 计算能力纵向扩展：集群规格升降配。本文介绍详细信息。
- 计算能力横向扩展：增加或减少只读节点。具体操作说明，请参见[增加或删除节点](#)。
- 存储空间横向扩展：POLARDB采用Serverless架构，无需手动设置容量或扩缩容，容量随用户数据量的变化而自动在线调整。

变更配置的费用说明

详情请参见[#unique_55](#)。

注意事项

- 您只能对整个集群进行规格升降级，无法对集群中的单个节点进行规格升降级。
- 集群规格的升降级不会对集群中已有数据造成任何影响。
- 在集群规格变更期间，POLARDB服务会出现几秒钟的闪断且部分操作不能执行的状况，建议您在业务低谷期执行变更。闪断后需在应用端重新连接。

操作步骤

1. 登录[POLARDB控制台](#)。
2. 在控制台左上角，选择集群所在地域。

3. 进入升降配向导页面。您可以按照如下两种方式操作：

- 单击目标集群操作栏中的升降配。

集群列表

创建新集群

集群ID

请输入

Q

刷新

下载

集群名称	状态	兼容数据库	节点数	主节点配置	已使用数据	付费类型	操作
pc-xxxxxx	运行中	MySQL 5.6	2	2核 4GB	2.83 GB	包年包月 2019年8月15日 00:00:00 到期	升降配 增删节点

- 单击目标集群ID，在基本信息页面下方，单击升降配。

基本信息

配置与管理

账号管理

数据库管理

备份恢复

参数配置

诊断与优化

集群总览

性能监控

问题分析

慢 SQL

日志与审计

SQL 洞察

集群地址【推荐】 ① 新增自定义地址

默认地址 (pe-) 修改配置

读写模式 可读可写 (自动读写分离)

私网 pc- 修改

公网 申请

> 节点配置

> 高级配置

节点信息

增删节点

升降配

节点名称	可用区	状态	当前角色	规格	最大IOPS	操作
ph-	杭州 可用区G	运行中	主节点	2核 4GB	8000	重启
ph-	杭州 可用区G	运行中	只读节点	2核 4GB	8000	重启

4. 勾选升级配置或降级配置，单击确定。

升降配向导（包年包月）

您当前的付费方式为 **包年包月**，支持以下配置变更方案：

☒ 升级配置

支持您在当前生命周期内立即升级POLARDB的规格配置，预计10分钟生效，过程中每个连接地址都会有不超过30秒的连接闪断，请确保应用具备重连机制。参考文档：[变更配置](#)

☐ 临时升级配置

支持您临时升级POLARDB的规格配置，应对短时间（一般小于7天）的业务高峰期。临时升级只需要支付升级期间的费用，升级前支付（预付费）。临时升级期间暂不支持添加节点，请先扩容节点，再操作临时升级。临时升级期间也不支持普通升降级或增删节点，因此建议您尽量一次性升级到最高的配置，避免重复升级。临时升级和到期还原时，会引起连接闪断，请确保应用具备重连机制。详细功能和计费介绍，请参考文档：[临时升配](#)

☐ 降级配置

支持您在当前生命周期内立即降低POLARDB的规格配置，预计10分钟生效，过程中每个连接地址都会有不超过30秒的连接闪断，请确保应用具备重连机制。参考文档：[变更配置](#)、[降配退费规则](#)

确定
取消



说明：

仅包年包月集群支持临时升级配置，详情请参见[#unique_61](#)。

5. 选择所需的规格。



说明：

同一集群中，所有节点的规格总是保持一致。

6. 勾选服务协议，单击去支付并完成支付。



说明：

规格变更预计需要10分钟生效。

相关API

API	描述
#unique_58	变更POLARDB集群节点规格。

9.4 临时升配

POLARDB的包年包月集群支持临时升配，可以帮助您轻松应对短时间的业务高峰期。

前提条件

- 集群为包年包月集群。
- 集群没有尚未生效的续费变配订单。
- 集群没有尚未生效的临时升配订单。

功能说明

临时升配是指临时升级规格，提升整体性能。到达指定的还原时间后，集群的规格会自动还原到临时升配前的状态。



说明:

不支持临时降级，如需降级请参见[变更配置](#)。

影响

还原过程可能会出现闪断，请确保应用程序具备重连机制。

注意事项

- 还原时间不能晚于集群到期时间的前1天。例如集群1月10日到期，则临时升配的还原时间最多为1月9日。
- 临时升配的最短时间为1天，由于设置还原时间后无法修改，建议升配时间为14天以内。
- 临时升配期间不支持普通的[变更配置](#)。
- 临时升配后如果性能不够，在还原时间到达之前最多可以再进行1次升配，此次设置的还原时间不能早于第1次。

计费

临时升配的价格是新老配置差价的1.5倍。计算公式如下：

临时升配N天，费用 = (新规格包月价格 - 老规格包月价格) / 30 x 1.5 x N。

临时升配

1. 登录[POLARDB控制台](#)。
2. 在控制台左上角，选择集群所在地域。

3. 进入升降配向导页面。您可以按照如下两种方式操作：

- 单击目标集群操作栏中的升降配。

集群列表							
POLARDB简介							
创建新集群	集群ID	请输入	Q	刷新		下载	
集群名称	状态	兼容数据库	节点数	主节点配置	已使用数据	付费类型	操作
pc-xxxxxx	运行中	MySQL 5.6	2	2核 4GB	2.83 GB	包年包月 2019年8月15日 00:00:00 到期	升降配 增删节点

- 单击目标集群ID，在基本信息页面下方，单击升降配。

基本信息

配置与管理

账号管理

数据库管理

备份恢复

参数配置

诊断与优化

集群总览

性能监控

问题分析

慢 SQL

日志与审计

SQL 洞察

集群地址【推荐】 新增自定义地址

默认地址 (pe-) 修改配置

读写模式 可读可写 (自动读写分离)

私网 pc- 修改

公网 申请

> 节点配置

> 高级配置

节点信息

增删节点

升降配

节点名称	可用区	状态	当前角色	规格	最大IOPS	操作
ph-	杭州 可用区G	运行中	主节点	2核 4GB	8000	重启
ph-	杭州 可用区G	运行中	只读节点	2核 4GB	8000	重启

4. 勾选临时升级配置，单击确定。



说明：

仅包年包月集群支持临时升级配置。

升降配向导（包年包月）

×

您当前的付费方式为 **包年包月**，支持以下配置变更方案：

☐

升级配置

支持您在当前生命周期内立即升级POLARDB的规格配置，预计10分钟生效，过程中每个连接地址都会有不超过30秒的连接闪断，请确保应用具备重连机制。参考文档：[变更配置](#)

☒

临时升级配置

支持您临时升级POLARDB的规格配置，应对短时间（一般小于7天）的业务高峰期。临时升级只需要支付升级期间的费用，升级前支付（预付费）。临时升级期间暂不支持添加节点，请先扩容节点，再操作临时升级。临时升级期间也不支持普通升降级或增删节点，因此建议您尽量一次性升级到最高的配置，避免重复升级。临时升级和到期还原时，会引起连接闪断，请确保应用具备重连机制。详细功能和计费介绍，请参考文档：[临时升配](#)

☐

降级配置

支持您在当前生命周期内立即降低POLARDB的规格配置，预计10分钟生效，过程中每个连接地址都会有不超过30秒的连接闪断，请确保应用具备重连机制。参考文档：[变更配置](#)、[降配退费规则](#)

确定

取消

5. 选择节点规格。

6. 选择还原时间。

配置变更

可选择在一定时间内提升实例的配置，并在到期后自动恢复

规格

pl-

8 核 32 GB（独享）

所有POLARDB节点规格均为独享型配置，性能稳定可靠，详见 [规格与定价](#)。

pl-

8 核 32 GB（独享）

所有POLARDB节点规格均为独享型配置，性能稳定可靠，详见 [规格与定价](#)。

[撤销所有规格变更](#)

临时变更

还原时间

2019-07-03

13

时

注意：到达还原时间后，带宽将降为升级前的值。还原过程不中断业务，但带宽从高变低有可能会出现闪断，建议后端应用具备重连机制。临时升级支持最短升级间隔为2小时，按小时单价计费，支付完成后带宽即刻升级成功，升级过程不中断业务

说明：

文档版本：20191120

55

- 临时升配后如果性能不够，在还原时间到达之前最多可以再进行1次升配，此次设置的还原时间不能早于第1次。
- 临时升配的最短时间为1天，由于设置还原时间后无法修改，建议升配时间为14天以内。
- 还原时间不能晚于集群到期时间的前一天。

7. 勾选服务协议，单击去支付完成支付。

9.5 设置集群参数

本文将介绍如何通过控制台修改集群参数。

背景信息

您可以通过POLARDB控制台修改集群参数，关于参数的详细说明，请参考[参数说明](#)。

注意事项

- 请按照控制台上规定的修改范围修改参数值。

<

集群 pc- 运行中

基本信息

配置与管理

账号管理

数据库管理

备份恢复

参数配置

诊断与优化

日志与审计

提交修改 全部撤销 请输入

名称	当前值	重启	默认值	修改范围
autocommit	ON	否	ON	[ON OFF]
automatic_sp_privileges	ON	否	ON	[ON OFF]
back_log	3000	是	3000	[0-65535]
binlog_checksum	CRC32	否	CRC32	[NONE CRC32]
binlog_rows_query_log_events	OFF	否	OFF	[ON OFF]
binlog_row_image	FULL	否	FULL	[full minimal nobiob]
binlog_stmt_cache_size	32768	否	32768	[4096-4294967295]
character_set_filesystem	binary	否	binary	[utf8 latin1 gbk utf8mb4 binary]
character_set_server	utf8	是	utf8	[utf8 latin1 gbk utf8mb4]
connect_timeout	10	否	10	[2-31536000]

- 部分参数修改后需要重启全部节点，重启前请做好业务安排，谨慎操作。详情请参见参数设置页面中的重启列。

<

集群

pc-1

运行中

基本信息

配置与管理

账号管理

数据库管理

备份恢复

参数配置

诊断与优化

日志与审计

提交修改

全部撤销

请输入

Q

名称	当前值	重启	默认值	修改范围
autocommit [?]	ON	否	ON	[ON OFF]
automatic_sp_privileges [?]	ON	否	ON	[ON OFF]
back_log [?]	3000	是	3000	[0-65535]
binlog_checksum [?]	CRC32	否	CRC32	[NONE CRC32]
binlog_rows_query_log_events [?]	OFF	否	OFF	[ON OFF]
binlog_row_image [?]	FULL	否	FULL	[full minimal noblob]
binlog_stmt_cache_size [?]	32768	否	32768	[4096-4294967295]
character_set_filesystem [?]	binary	否	binary	[utf8 latin1 gbk utf8mb4 binary]
character_set_server [?]	utf8	是	utf8	[utf8 latin1 gbk utf8mb4]
connect_timeout [?]	10	否	10	[2-31536000]

操作步骤

1. 登录POLARDB控制台。
2. 在控制台左上角，选择集群所在地域。
3. 单击目标集群ID。
4. 在左侧导航栏中，选择配置与管理 > 参数配置。
5. 修改一个或多个参数的当前值，单击提交修改。

提交修改

全部撤销

请输入

Q

名称	当前值	重启	默认值	修改范围
autocommit ?	ON	否	ON	[ON OFF]
automatic_sp_privileges ?	ON	否	ON	[ON OFF]
back_log ?	4000	是	3000	[0-65535]
binlog_checksum ?	CRC32	否	CRC32	[NONE CRC32]
binlog_rows_query_log_events ?	OFF	否	OFF	[ON OFF]

6. 在弹出的保存改动对话框中，单击确定。



相关API

API	描述
#unique_64	查看集群的参数
#unique_65	修改集群的参数

9.6 设置可维护窗口

本文为您介绍如何设置可维护窗口，以便您在维护过程中不会影响业务。

背景信息

在阿里云平台上，为保障云数据库POLARDB的稳定性，后端系统会不定期对集群进行维护操作，确保集群平稳运行。您可以根据业务规律，将可维护窗口设置在业务低峰期，以免维护过程中对业务造成影响。

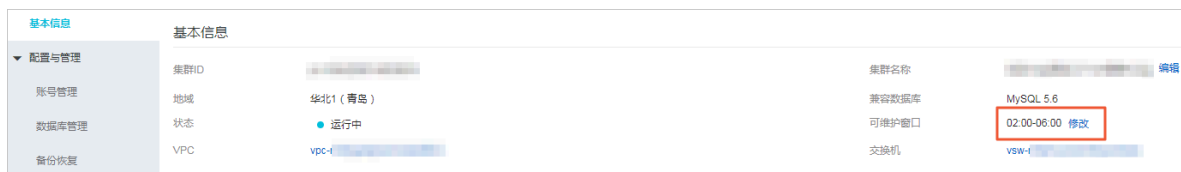
注意事项

- 在进行正式维护前，POLARDB 会给阿里云账号中设置的联系人发送短信和邮件，请注意查收。
- 集群维护当天，为保障整个维护过程的稳定性，集群会在所设置的可维护窗口之前的一段时间，进入集群维护中的状态，当集群处于该状态时，数据库本身正常的数据库访问不会受到任何影响，但该集群的控制台上，除了账号管理、数据库管理和添加 IP 白名单外，其他涉及变更类的功能均无法使用（如常用的升降级、重启等操作均无法重启），查询类如性能监控等可以正常查阅。
- 在进入集群所设置的可维护窗口后，集群会在该段时间内发生1到2次的连接闪断，请确保您的应用程序具有重连机制。闪断后，集群即可恢复到正常状态。

操作步骤

1. 登录[POLARDB控制台](#)。

2. 在控制台左上角，选择集群所在地域。
3. 单击目标集群ID。
4. 在基本信息中的可维护窗口后，单击修改。



5. 在编辑可维护窗口中，选择集群的可维护窗口，单击提交。

相关API

API	描述
CreateDBCluster	创建数据库集群。
ModifyDBClusterMaintainTime	修改集群可运维时间。

9.7 重启节点

本文介绍当数据库节点出现连接数满或性能问题时，如何手动重启节点。

背景信息

当数据库节点出现连接数满或性能问题时，您可以手动重启节点。重启节点会造成连接中断，重启前请做好业务安排，谨慎操作。

操作步骤

1. 登录[POLARDB控制台](#)。
2. 在控制台左上角，选择集群所在地域。
3. 单击目标集群ID。
4. 在基本信息页面下方节点信息里找到需要重启的节点。



5. 单击操作栏中的重启。
6. 在弹出的对话框中，单击确认。

相关API

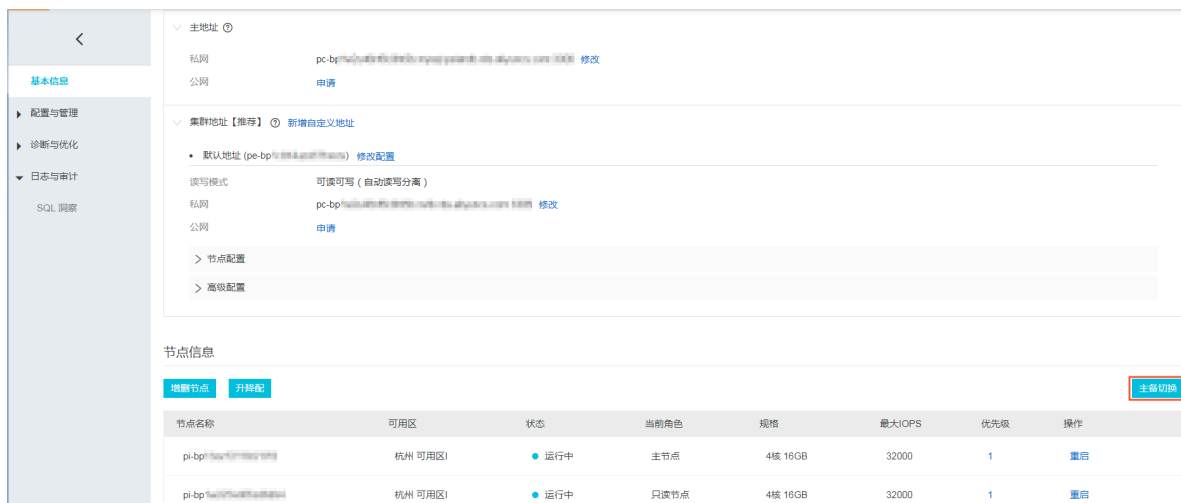
API	描述
#unique_59	重启数据库节点。

9.8 主备切换

POLARDB是一个多节点的集群，其中一个节点是主节点（Master），其他节点为只读节点。除了因系统故障自动进行主备切换外，您也可以手动进行主备切换，指定一个只读节点为新的主节点，适用于高可用演练，或者需要指定某个节点为主节点的场景。

手动切换

1. 登录[POLARDB控制台](#)。
2. 在控制台左上角，选择集群所在地域。
3. 单击目标集群ID。
4. 在基本信息页面的节点信息区域，单击主备切换。



5. 在弹出的对话框中，选择新主节点，单击确认。

主备切换

提升一个只读节点为新的主节点。

新主节点:

请选择

切换时可能会出现30秒左右的闪断，请确保应用具备重连机制。

确认

取消



说明:

如果您没有选择新主节点，系统将自动选举优先级最高的只读节点作为新主节点进行切换。切换时可能会出现30秒左右的闪断，请确保应用具备重连机制。

自动切换

POLARDB采用双活（Active-Active）的高可用集群架构，可读写的主节点和只读节点之间自动进行故障切换（Failover），系统自动选举新的主节点。

POLARDB每个节点都有一个故障切换（Failover）优先级，决定了故障切换时被选举为主节点的概率高低。当多个节点的优先级相同时，则有相同的概率被选举为主节点。

自动选举主节点按以下步骤进行：

1. 系统找出当前可以被选举的所有只读节点。
2. 选择优先级最高的一个或多个只读节点。
3. 如果切换第一个节点失败（例如，网络原因、复制状态异常等），则会尝试切换下一个，直至成功。

相关API

API	描述
#unique_70	手动对POLARDB集群进行主备切换，可以指定一个只读节点为新的主节点。

9.9 释放集群

本文介绍如何手动释放整个集群。

背景信息

您可以根据业务需求手动释放后付费（按小时付费）的集群。

注意事项

- 预付费（包年包月）集群不支持手动释放，集群到期后会自动被释放。
- 只有在运行状态下的集群才能被手动释放。
- 集群被释放后，数据将无法找回，请谨慎操作。
- 本功能用于释放整个集群，包括集群中的所有节点。如要释放单个只读节点，请参考[增加或删除节点](#)。
- 按小时付费的集群可以直接转为包年包月，具体请参见[按小时付费转包年包月](#)。

操作步骤

1. 登录[POLARDB控制台](#)。
2. 在控制台左上角，选择集群所在地域。
3. 找到目标集群，单击操作列的… > 释放。



4. 在弹出的提示框中，单击确认。

相关API

API	描述
#unique_52	查看集群列表。
#unique_72	删除数据库集群。

9.10 克隆集群

您可以根据已有的POLARDB集群的数据（包括账号信息，但不包括集群参数配置信息），克隆出相同的POLARDB集群。

注意事项

被克隆的是执行克隆动作时的数据。克隆开始后，新写入的数据不会被克隆。

操作步骤

1. 登录[POLARDB控制台](#)。

2. 在控制台左上角，选择集群所在地域。
3. 找到要克隆的集群，单击操作列的… > 恢复到新集群。
4. 在配置页面设置以下参数：

参数	说明
克隆源类型	这里选择本集群。
地域	指集群所在的地理位置。克隆集群的地域和原集群相同，不支持修改。
主可用区	· 可用区是地域中的一个独立物理区域，不同可用区之间没有实质性区别。 · 您可以选择将POLARDB与ECS创建在同一可用区或不同的可用区。
网络类型	· 无需选择。 · 仅支持专有网络VPC（Virtual Private Cloud）。VPC是一种隔离的网络环境，安全性和性能均高于传统的经典网络。
VPC网络 VPC交换机	从下拉菜单中选择VPC和交换机，或者 创建新的VPC和交换机 。  说明： 请确保POLARDB与需要连接的ECS创建于同一个VPC，否则它们无法通过内网互通，无法发挥最佳性能。
数据库引擎	无需选择。
节点规格	按需选择。不同规格有不同的最大存储容量和性能，具体请参见 #unique_20 。
节点个数	无需选择。系统将自动创建一个与主节点规格相同的只读节点。
集群名称	· 可选。 · 如果留空，系统将为您自动生成一个集群名称。创建集群后还可以修改集群名称。
购买时长	预付费集群需要填写此参数。
购买数量	默认为1，无法修改。

5. 阅读并勾选《云数据库 POLARDB服务协议》，单击去开通。

10 多可用区部署

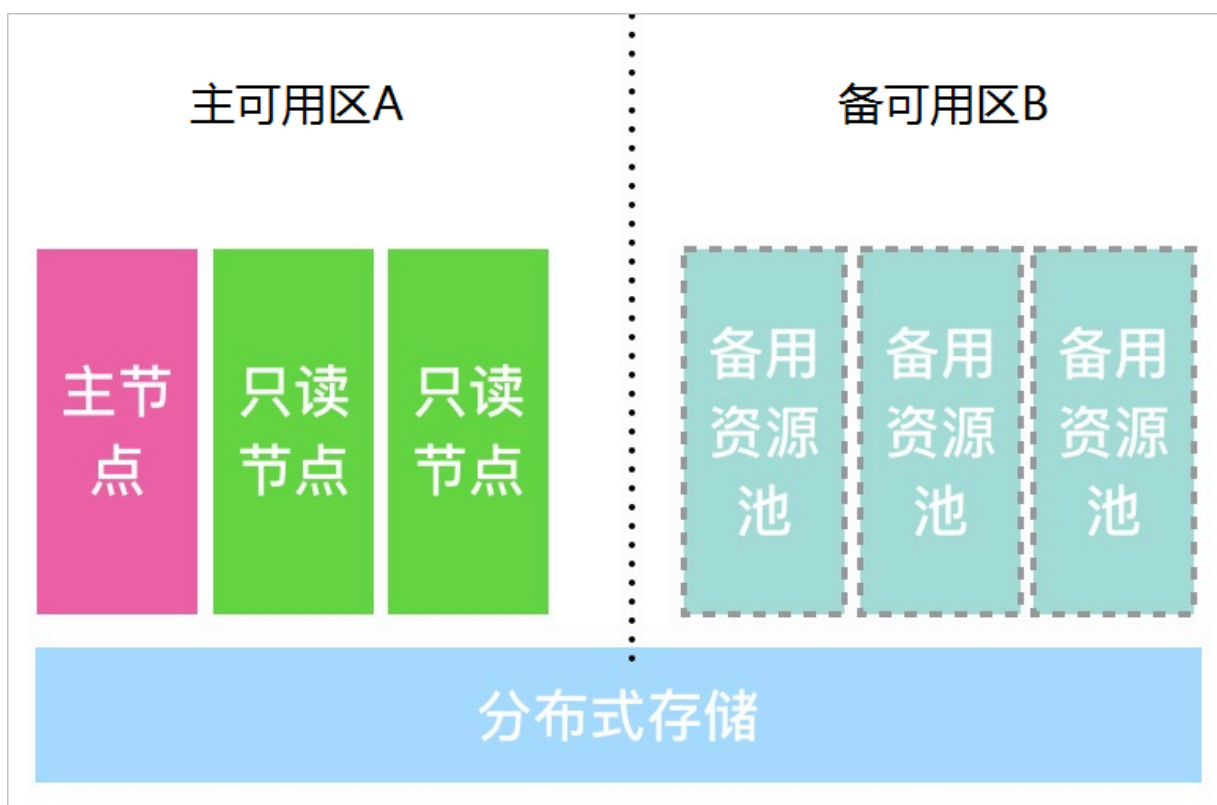
POLARDB for MySQL支持创建多可用区的集群。相比单可用区集群，多可用区集群具备更高的容灾能力，可以抵御机房级别的故障。本文将为您介绍如何实施多可用区部署以及如何更换主可用区。

前提条件

- 地域为：华东1（杭州）、华北3（张家口）。
- 地域当前拥有足够计算资源的可用区数量大于等于2。

多可用区架构

使用多可用区集群时，数据分布在多个可用区内。计算节点暂时要求位于主可用区，POLARDB会在备可用区预留足够的资源用于主可用区故障时进行故障切换。多可用区架构如下。



费用

多可用区功能不需要支付额外费用。



说明:

单可用区集群也会免费升级至多可用区集群。

如何实现多可用区架构

当满足前提条件时，**新建集群**会默认为多可用区集群。



包年包月 | 按小时付费

包年包月集群中也存在按小时付费的计费项，共两个：存储空间、SQL洞察（可选）。

地域：
华东1（杭州） 华东2（上海） 华北1（青岛） 华北2（北京） 华北5（呼和浩特） 华南1（深圳）
香港 华北3（张家口）
所在的地理位置，购买后无法更换。
请确保POLARDB与需要连接的ECS位于同一个地域，否则它们内网无法直接互通。

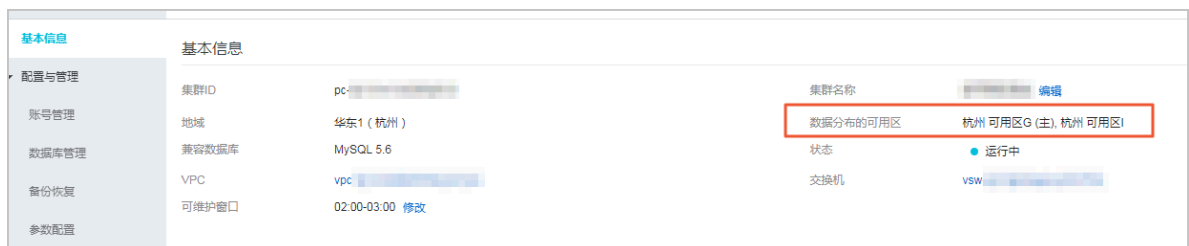
创建方式：
默认创建 从RDS克隆 从RDS迁移
创建一个全新的POLARDB

主可用区：
华东1可用区G 华东1可用区I
您可以选择将POLARDB与ECS创建在同一可用区或不同的可用区。
在有二个及以上可用区的地域，POLARDB会自动复制数据到备可用区，用于灾难恢复。

网络类型：
专有网络
请确保与需要连接的ECS在同一个VPC网络！如需使用其他VPC网络或者VPC交换机，请到[控制台](#)创建。

查看集群所属可用区

1. 登录**POLARDB控制台**。
2. 在控制台左上角，选择集群所在地域。
3. 单击目标集群ID。
4. 在基本信息页面，查看数据分布的可用区。



基本信息 | 配置与管理 | 账号管理 | 数据库管理 | 备份恢复 | 参数配置

基本信息

集群ID	pc-xxxxxx	集群名称	xxxxxx 编辑
地域	华东1（杭州）	数据分布的可用区	杭州可用区G（主）、杭州可用区I
兼容数据库	MySQL 5.6	状态	运行中
VPC	vpc-xxxxxx	交换机	vsw-xxxxxx
可维护窗口	02:00-03:00 修改		

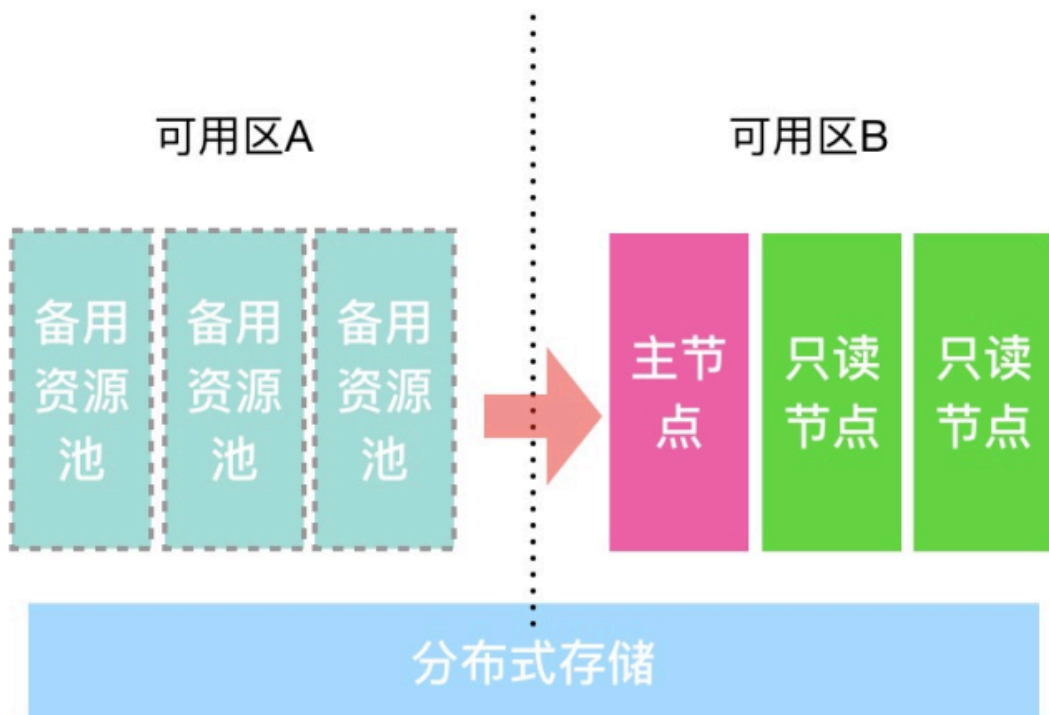
更换主可用区

POLARDB支持更换主可用区，您可以通过该功能将数据库集群计算节点迁移到其他可用区，适用于灾难恢复或者让ECS就近访问的场景。

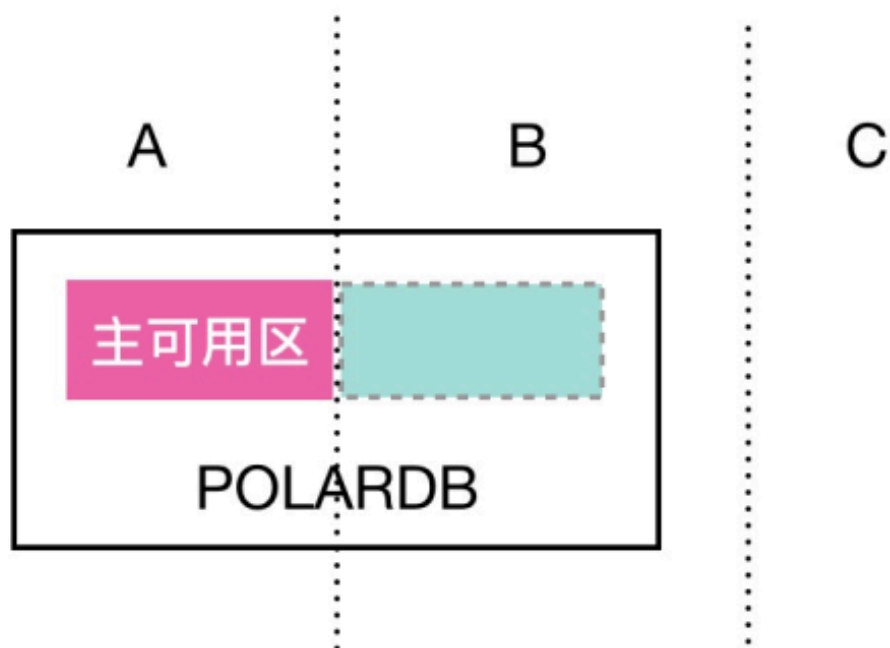


说明：

目前，更换主可用区将强制绑定所有节点同时迁移到目标可用区，数据库连接地址（集群访问地址和主访问地址）不变，但使用的虚拟交换机vSwitch和IP地址可能会发生变化。



- 如果目标可用区是备可用区（如下图所示从A到B），则不需要迁移数据。系统只需要切换数据库计算节点，因此可以达到比较快的跨机房切换效果（平均耗时5分钟/节点），该操作常用于容灾演练。
- 如果目标可用区不在备可用区（如下图所示从A到C），则需要迁移数据。系统执行迁移时间长短跟数据容量有关，可能需要几个小时，请谨慎操作。该操作一般用于调整应用和数据库的可用区分布，达到就近访问数据库的目的。



1. 登录POLARDB控制台。
2. 在控制台左上角，选择集群所在地域。
3. 单击目标集群ID。
4. 在基本信息页面的节点信息区域，单击迁移可用区。

节点信息							
删除节点		升降配		主备切换		迁移可用区	
节点名称	可用区	状态	当前角色	规格	最大IOPS	Failover 优先级	操作
pi-12345678901234567890	杭州 可用区I	运行中	主节点	4核 16GB	32000	1	重启
pi-12345678901234567890	杭州 可用区I	运行中	只读节点	4核 16GB	32000	1	重启

5. 在更换主可用区对话框中，选择目标可用区和目标交换机，单击确认。

更换主可用区

* 目标可用区

请选择

* 目标交换机

请选择

数据库节点

☒ 全部

生效时间

☒ 立即生效

更换主可用区将迁移全部数据库节点到新的可用区，连接地址不变，但可能会使用新可用区的IP。该操作可能会影响数据库的可用性，详见[文档](#)

确认

取消



说明:

更换主可用区将迁移全部数据库节点到新的可用区，连接地址不变，但可能会使用新可用区的IP地址。该操作可能会影响数据库的可用性，请谨慎操作。

11 账号管理

11.1 账号概述

本文介绍阿里云控制台账号和POLARDB数据库集群账号的相关内容。

控制台账号

您可以使用以下账号登录控制台：

- **阿里云账号**：该账号是阿里云资源的归属和计费主体。购买阿里云产品之前，您需要先注册阿里云账号。
- **子账号（可选）**：如果其他人需要使用您账号下的资源，您可以使用RAM控制台创建和管理子账号。子账号本身不拥有资源，且以主账号作为计费主体。

数据库集群账号

您可以使用以下账号登录数据库集群。更多信息请参见[创建数据库账号](#)。

账号类型	说明
高权限账号	· 只能通过控制台创建和管理。 · 一个集群只能有一个高权限账号，可以管理所有普通账号和数据库。 · 开放了更多权限，可满足个性化和精细化的权限管理需求，比如可按用户分配不同表的查询权限。 · 拥有集群中所有数据库的所有权限。 · 可以断开任意账号的连接。
普通账号	· 可以通过控制台或者SQL语句创建和管理。 · 一个集群可以创建多个普通账号，具体的数量与数据库内核有关。 · 需要手动给普通账号授予特定数据库的权限。 · 普通账号不能创建和管理其他账号，也不能断开其他账号的连接。

相关API

API	描述
#unique_78	创建账号。
#unique_79	查看账号列表。
#unique_80	修改账号备注。
#unique_81	修改账号密码。

API	描述
#unique_82	账号授权。
#unique_83	撤销账号权限。
#unique_84	重置账号权限。
#unique_85	删除账号。

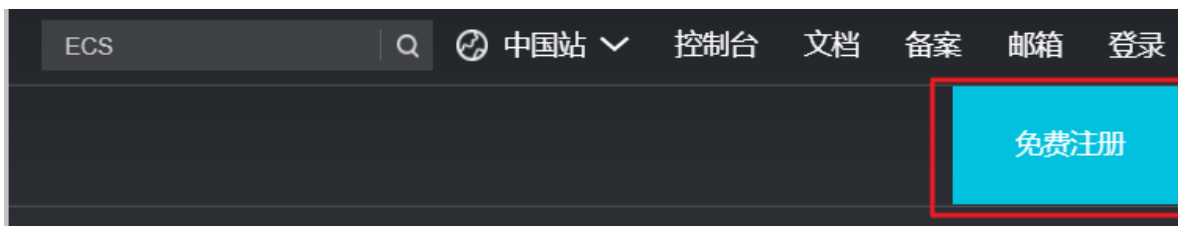
11.2 注册和登录阿里云账号

本文介绍如何注册和登录阿里云账号。

注册阿里云账号

您可以通过两种方式注册阿里云账号：

- 进入阿里云官网（<https://www.aliyun.com>），单击右上角的免费注册。



- 直接访问注册页面（<https://account.aliyun.com/register/register.htm>）。

登录阿里云账号

阿里云账号的登录入口与子账号不同。

- 阿里云账号的登录入口：<https://account.aliyun.com/login/login.htm>。

密码登录

扫码登录

扫码登录

淘宝及1688会员可直接使用会员名登录

邮箱/会员名/8位ID

登录密码

登录

忘记密码 忘记会员名 免费注册

其他方式登录：



- 子账号的登录入口：<https://signin.aliyun.com/login.htm>

[使用主帐号登录](#)

子用户登录

登录用户名称

RAM用户新版登录格式：<子用户名称>@<企业别名> 例如：
username@company-alias

下一步

11.3 创建和使用子账号

如果只有您本人使用POLARDB，那么使用阿里云账号即可。如果需要让其他人使用您账号下的资源，请创建子账号。

创建RAM子账号

1. 您可以使用阿里云账号或有RAM权限的子账号来创建子账号，首先需要登录RAM控制台。具体操作如下：

- 如果使用阿里云账号，请[点此登录](#)。
- 如果使用子账号，请[点此登录](#)。



说明：

子账号登录的格式为子账号名@公司别名。

2. 在左侧导航栏的人员管理菜单下，单击用户。

3. 单击新建用户。



说明:

单击添加用户，可一次性创建多个RAM用户。

4. 输入登录名称和显示名称。

5. 在访问方式区域下，选择控制台密码登录。

6. 控制台密码选择自动生成默认密码或自定义登录密码。

7. 要求重置密码选择用户在下次登录时必须重置密码或无需重置。

8. 多因素认证选择不要求。

9. 单击确认。

在授权页面下为RAM用户授权

1. 在左侧导航栏的权限管理菜单下，单击授权。

2. 单击新增授权。

3. 在被授权主体区域下，输入RAM用户名称后，单击需要授权的RAM用户。

4. 在左侧权限策略名称列表下，单击需要授予RAM用户的权限策略。



说明:

在右侧区域框，选择某条策略并单击×，可撤销该策略。

5. 单击确定。

6. 单击完成。

在用户页面下为RAM用户授权

1. 在左侧导航栏的人员管理菜单下，单击用户。

2. 在用户登录名称/显示名称列表下，找到目标RAM用户。

3. 单击添加权限，被授权主体会自动填入。

4. 在左侧权限策略名称列表下，单击需要授予RAM用户的权限策略。



说明:

在右侧区域框，选择某条策略并单击×，可撤销该策略。

5. 单击确定。

6. 单击完成。

登录子账号

前提条件：您已完成上述账号授权步骤。

您可以通过两种地址登录子账号：

- 通用登录地址：<https://signin.aliyun.com/login.htm>

如果通过此地址登录，您需手动输入子账号名以及公司别名。格式为子账号名@公司别名。

- 专用登录地址：如果您可以登录RAM控制台，可以在RAM控制台查看到您公司的子账号登录地址。



如果通过此地址登录，系统将自动为您填写公司别名，您只需输入子账号名。

更多操作

您还可以对子账号进行更多的操作，如把子账号添加到用户组、为子账号分配角色、为用户组或角色授权等。详情请参见[RAM用户指南](#)。

11.4 创建数据库账号

本文为您介绍如何创建数据库账号，以及高权限账号与普通账号的区别。

背景信息

POLARDB for MySQL支持两种数据库账号：高权限账号和普通账号。您可以在控制台管理所有账号。



说明：

出于安全原因，POLARDB不提供root账号。

账号类型	说明
高权限账号	· 只能通过控制台创建和管理。 · 一个集群只能有一个高权限账号，可以管理所有普通账号和数据库。 · 开放了更多权限，可满足个性化和精细化的权限管理需求，例如可按用户分配不同表的查询权限。 · 拥有集群中所有数据库的所有权限。 · 可以断开任意账号的连接。
普通账号	· 可以通过控制台或者SQL语句创建和管理。 · 一个集群可以创建多个普通账号，具体的数量与数据库内核有关。 · 需要手动给普通账号授予特定数据库的权限。 · 普通账号不能创建和管理其他账号，也不能断开其他账号的连接。

创建高权限账号

1. 登录[POLARDB控制台](#)。
2. 在控制台左上角，选择集群所在地域。
3. 单击目标集群ID。
4. 在左侧导航栏中，选择账号管理。
5. 单击创建账号。

6. 设置以下参数：

参数	说明
账号名	填写账号名称。要求如下： · 以小写字母开头，以字母或数字结尾； · 由小写字母、数字或下划线组成； · 长度为2~16个字符。 · 不能使用某些预留的用户名，如root、admin。
账号类型	此处选择高权限账号。  说明： 如果已经创建过高权限账号，则无法再选择高权限账号，因为每个集群只能有一个高权限账号。
密码	设置账号的密码。要求如下： · 由大写字母、小写字母、数字或特殊字符组成，至少包含其中三类； · 长度为8~32个字符； · 特殊字符为!@#\$\$%^&*()_+~=
确认密码	再次输入密码。
备注	备注该账号的相关信息，便于后续账号管理。要求如下： · 不能以http://或https://开头； · 必须以大小写字母或中文开头； · 可以包含大小写字母、中文、数字、下划线"_"或连字符"-"; · 长度为2~256个字符。

7. 单击确定。

创建普通账号

1. 登录[POLARDB控制台](#)。
2. 在控制台左上角，选择集群所在地域。
3. 单击目标集群ID。
4. 在左侧导航栏中，选择账号管理。
5. 单击创建账号。

6. 设置以下参数：

参数	说明
账号名	填写账号名称。要求如下： · 以小写字母开头，以字母或数字结尾； · 由小写字母、数字或下划线组成； · 长度为2~16个字符； · 不能使用某些预留的用户名，如root、admin。
账号类型	此处选择普通账号。
授权数据库	为该账号授予一个或多个数据库的权限。本参数也可以留空，在创建账号后再给账号授权。 - 从左侧选中一个或多个数据库，单击向右的箭头添加到右侧。 - 在右侧框中，为数据库选择读写、只读或仅DML权限。
密码	设置账号密码。要求如下： · 由大写字母、小写字母、数字或特殊字符组成，至少包含其中三类。 · 长度为8~32个字符。 · 特殊字符为!@#\$%^&*()_+-=。
确认密码	再次输入密码。
备注	备注该账号的相关信息，便于后续账号管理。要求如下： · 不能以http://或https://开头。 · 必须以大小写字母或中文开头。 · 可以包含大小写字母、中文、数字、下划线"_"或连字符"-"。 · 长度为2~256个字符。

7. 单击确定。

重置高权限账号的权限

如果高权限账号自身出现问题，例如权限被意外回收（REVOKE），您可以通过输入高权限账号的密码来重置高权限账号的权限，使其恢复正常。

1. 登录[POLARDB控制台](#)。
2. 在控制台左上角，选择集群所在地域。
3. 单击目标集群ID。
4. 在左侧导航栏中，选择账号管理。
5. 单击高权限账号右侧的重置权限。
6. 在弹出的对话框中，输入高权限账号的密码，即可重置该账号的权限。

下一步

[查看连接地址](#)

相关API

API	描述
#unique_78	创建账号。
#unique_79	查看账号列表。
#unique_80	修改账号备注。
#unique_81	修改账号密码。
#unique_82	账号授权。
#unique_83	撤销账号权限。
#unique_84	重置账号权限。

11.5 管理数据库账号

本文为您介绍如何管理数据库账号，包括重置高权限账号权限、修改普通账号权限、修改账号密码以及删除账号等。

背景信息

POLARDB for MySQL支持两种数据库账号：高权限账号和普通账号。您可以在控制台管理所有账号和数据库。

注意事项

出于安全原因，POLARDB不提供root账号。

创建数据库账号

具体操作请参见[创建数据库账号](#)。

修改账号密码

1. 登录[POLARDB控制台](#)。
2. 单击目标集群ID。
3. 在左侧导航栏中，单击配置与管理 > 账号管理。

4. 找到目标账号，单击操作栏中的修改密码。



5. 在弹出的对话框中，输入新密码和确认新密码，单击确定。

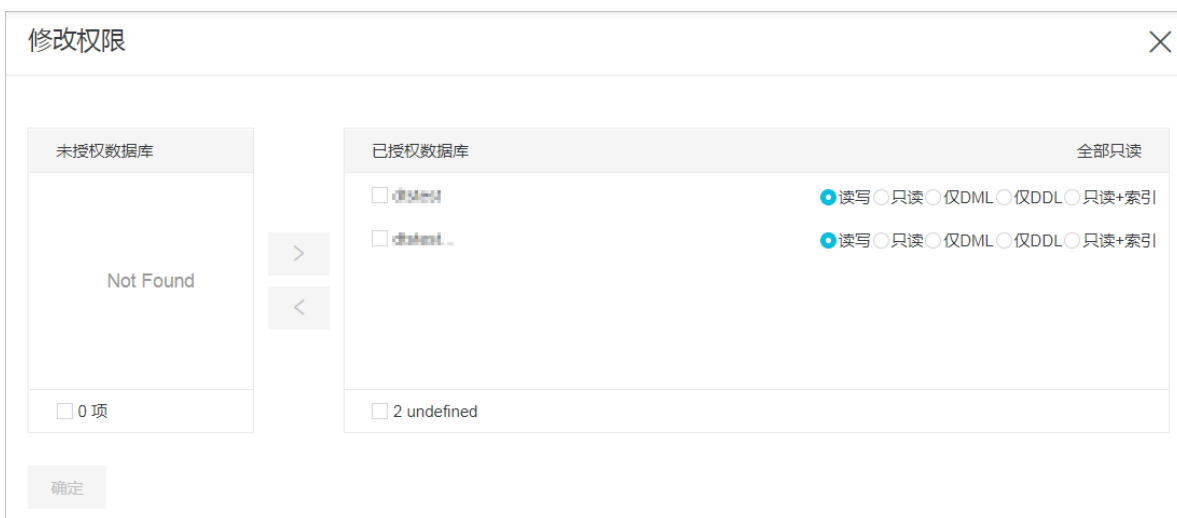
重置高权限账号权限

如果高权限账号自身出现问题，例如权限被意外回收（REVOKE），您可以通过输入高权限账号的密码来重置高权限账号的权限，使其恢复正常。

1. 登录POLARDB控制台。
2. 单击目标集群ID。
3. 在左侧导航栏中，单击配置与管理 > 账号管理。
4. 找到目标账号，单击操作栏中的重置权限。
5. 在弹出的对话框中，输入高权限账号的密码，单击确定，即可重置该账号的权限。。

修改普通账号权限

1. 登录POLARDB控制台。
2. 单击目标集群ID。
3. 在左侧导航栏中，单击配置与管理 > 账号管理。
4. 找到目标账号，单击操作栏中的修改权限。



5. 在弹出的对话框中，修改授权数据库以及修改已授权数据库的权限，单击确定。

删除账号

1. 登录[POLARDB控制台](#)。
2. 单击目标集群ID。
3. 在左侧导航栏中，单击配置与管理 > 账号管理。
4. 找到目标账号，单击操作栏中的删除。
5. 在弹出的对话框中，单击确定。

相关API

API	描述
#unique_78	创建账号
#unique_79	查看账号列表
#unique_80	修改账号备注
#unique_81	修改账号密码
#unique_82	账号授权
#unique_83	撤销账号权限
#unique_84	重置账号权限
#unique_85	删除账号

12 数据库管理

您可以在控制台创建和管理所有的数据库。

创建数据库

1. 登录[POLARDB控制台](#)。
2. 在控制台左上角，选择集群所在地域。
3. 单击目标集群的ID。
4. 在左侧导航栏中，选择配置与管理 > 数据库管理。
5. 单击创建数据库。
6. 设置以下参数。

The screenshot shows the '创建数据库' (Create Database) dialog box in the POLARDB console. The dialog is open over a list of existing databases. The fields in the dialog are:

- 数据库(DB)名称:** test1 (5/64 characters, valid).
- 支持字符集:** utf8mb4 (selected), utf8, gbk, latin1.
- 授权账号:** [selected account] (with a '创建新账号' link).
- 账号权限:** 读写 (selected), 只读, 仅DML.
- 备注说明:** [empty text area, 0/256 characters].

A '确定' (Confirm) button is at the bottom.

参数	说明
数据库 (DB) 名称	· 以字母开头，以字母或数字结尾； · 由小写字母、数字、下划线或中划线组成； · 长度为2~64个字符。 · 数据库名称在实例内必须是唯一的。
支持字符集	选择utf8mb4、utf8、gbk或latin1。 如果需要其他字符集，请在右侧下拉菜单中选择需要的字符集。

参数	说明
授权账号	选中需要授权访问本数据库的账号。本参数可以留空，在创建数据库后再绑定账号。  说明： 此处只会显示普通账号，因为高权限账号拥有所有数据库的所有权限，不需要授权。
账号权限	选择要授予账号的权限：读写、只读或仅DML。
备注说明	用于备注该数据库的相关信息，便于后续数据库管理。要求如下： · 不能以http://或https://开头； · 必须以大小写字母或中文开头； · 可以包含大小写字母、中文、数字、下划线"_"或连字符"-"; · 长度为2~256个字符。

7. 单击确定。

删除数据库

1. 登录[POLARDB控制台](#)。
2. 在控制台左上角，选择集群所在地域。
3. 单击目标集群的ID。
4. 在左侧导航栏中，选择配置与管理 > 数据库管理。
5. 找到目标数据库，单击操作列中的删除。
6. 在弹出的对话框中，单击确定。

相关API

API	描述
#unique_89	创建数据库。
#unique_90	查看数据库列表信息。
#unique_91	修改数据库描述。
#unique_92	删除数据库。

13 备份与恢复

13.1 备份数据

POLARDB采用物理备份（快照备份），每天自动备份一次，您也可以手动发起备份。自动备份和手动备份都不会影响集群的运行。备份文件保留的时间为7天。

备份类型

备份类型	说明
自动备份	- 默认为每天一次，您可以设置自动执行备份的时间段和周期。具体请参见设置自动备份。 - 备份文件不可删除。
手动备份	- 可随时发起。每个集群最多可以有3个手动创建的备份。具体请参见手动创建备份。 - 备份文件可删除。

费用

POLARDB备份文件占用的存储空间暂不收费。

设置自动备份

1. 登录[POLARDB控制台](#)。
2. 在控制台左上角，选择集群所在地域。
3. 单击目标集群的ID。
4. 在左侧导航栏中，选择配置与管理 > 备份恢复。
5. 单击备份设置。



6. 在弹出的对话框中，设置自动执行备份的时间段和周期，单击确定。



说明：

出于安全考虑，自动备份的频率为每周至少两次。

手动创建备份

1. 登录[POLARDB控制台](#)。
2. 在控制台左上角，选择集群所在地域。
3. 单击目标集群的ID。
4. 在左侧导航栏中，选择配置与管理 > 备份恢复。
5. 单击创建备份。



6. 在弹出的对话框中，单击确定。



说明：

每个集群最多可以有3个手动创建的备份。

恢复数据

请参见[恢复数据](#)。

相关API

API	描述
#unique_98	创建POLARDB集群全量快照备份。
#unique_99	查询POLARDB集群备份信息。
#unique_100	删除POLARDB集群备份。
#unique_101	查询POLARDB集群自动备份策略。
#unique_102	修改POLARDB集群自动备份策略。

13.2 恢复数据

本文为您介绍恢复POLARDB for MySQL数据的流程以及不同恢复方式的操作步骤。

恢复数据的流程

恢复POLARDB for MySQL数据的流程如下：

1. 将历史数据恢复到新集群中。您可以选择以下其中一种恢复方式：

- [按时间点恢复（恢复到任意时间点）](#)
- [按备份集（快照）恢复](#)

2. [登录到集群](#)，验证数据的准确性。

3. 将需要的数据迁移到原集群。



说明：

恢复后的集群包含原集群的数据和账号信息，不包含原集群的参数设置。

按时间点恢复数据（恢复到任意时间点）

您可以将过去7天内任意一个时间点的数据恢复到新集群。

1. 登录[POLARDB控制台](#)。
2. 在控制台左上角，选择集群所在地域。
3. 单击目标集群ID。
4. 在左侧导航栏中，选择配置与管理 > 备份恢复。
5. 单击按时间点恢复，在弹出的对话框中单击确认。

< 集群 pc-bp10jei4cx6j298sw ● 运行中 登录数据库 迁入		创建备份 按时间点恢复 备份设置 2018年10月20日 - 2018年12月20日 🔍				
基本信息 配置与管理 账号管理 数据库管理 备份恢复 参数配置 诊断与优化	备份开始/结束时间	备份方法	备份类型	备份策略	状态	操作
	2018-12-20 13:56:24	快照备份	全量备份	手动备份	0%	恢复备份 删除备份
	2018-12-20 04:47:03 - 2018-12-20 04:47:23	快照备份	全量备份	系统备份	备份完成	恢复备份 删除备份
	2018-12-19 04:47:03 - 2018-12-19 04:47:28	快照备份	全量备份	系统备份	备份完成	恢复备份 删除备份
	2018-12-18 04:47:06 - 2018-12-18 04:47:26	快照备份	全量备份	系统备份	备份完成	恢复备份 删除备份

6. 在克隆实例页面中，选择新集群的计费方式：

- **预付费：**在创建集群时需要支付计算集群（一个主节点和一个只读节点）的费用，而存储空间会根据实际数据量按小时计费，并从账户中按小时扣除。如果您要长期使用该集群，预付费方式更加划算，而且购买时长越长，折扣越多。
- **按小时付费：**无需预先支付费用，计算集群和存储空间（实际数据量）均按小时计费，并从账户中按小时扣除。如果您只需短期使用该集群，可以选择按量付费，用完即可释放，节省费用。

7. 设置以下参数：

参数	说明
克隆源类型	选择备份时间点。

参数	说明
备份时间点	可以设置为过去7天内的任意时间点。
地域	无需修改，与原集群相同。
可用区	无需修改。
网络类型	无需修改。
VPC网络和VPC交换机	建议保持不变，即原集群所在的VPC网络和交换机。
数据库引擎	无需修改。
节点规格	不同规格有不同的最大存储容量和性能，具体请参见 #unique_20 。
节点个数	无需修改。系统将自动创建一个与主节点规格相同的只读节点。
集群名称	如果留空，系统将为您自动生成一个集群名称。创建集群后还可以修改集群名称。
购买时长	预付费集群需要填写此参数。
集群数量	默认为1，无法修改。

8. 阅读并勾选《云数据库 POLARDB服务协议》，单击去开通，然后完成支付。

按备份集（快照）恢复

1. 登录[POLARDB控制台](#)。
2. 在控制台左上角，选择集群所在地域。
3. 单击目标集群ID。
4. 在左侧导航栏中，选择配置与管理 > 备份恢复。
5. 找到目标备份集（快照），单击恢复备份，在弹出的对话框中单击确定。
6. 在克隆实例页面中，选择新集群的计费方式：
 - 预付费：在创建集群时需要支付计算集群（一个主节点和一个只读节点）的费用，而存储空间会根据实际数据量按小时计费，并从账户中按小时扣除。如果您要长期使用该集群，预付费方式更加划算，而且购买时长越长，折扣越多。
 - 按小时付费：无需预先支付费用，计算集群和存储空间（实际数据量）均按小时计费，并从账户中按小时扣除。如果您只需短期使用该集群，可以选择按量付费，用完即可释放，节省费用。

7. 设置以下参数：

参数	说明
克隆源类型	选择备份集。
克隆源备份集	请确认是否为您要恢复的备份集。
地域	无需修改，与原集群相同。
可用区	无需修改。
网络类型	无需修改。
VPC网络和VPC交换机	建议保持不变，即原集群所在的VPC网络和交换机。
数据库引擎	无需修改。
节点规格	不同规格有不同的最大存储容量和性能，具体请参见 #unique_20 。
节点个数	无需修改。系统将自动创建一个与主节点规格相同的只读节点。
集群名称	如果留空，系统将为您自动生成一个集群名称。创建集群后还可以修改集群名称。
购买时长	预付费集群需要填写此参数。
集群数量	默认为1，无法修改。

8. 阅读并勾选《云数据库 POLARDB服务协议》，单击去开通，然后完成支付。

常见问题

1. 按时间点恢复的功能是否依赖Binlog？Binlog保存几天，就可以恢复这几天内任意时间点的数据库吗？

答：按时间点恢复的功能不依赖Binlog。集群可以恢复到过去7天内的任意时间点，是基于Redo日志的，与Binlog无关。

2. 数据恢复是否基于“全量备份+Binlog”？

答：数据恢复是基于“全量快照备份+Redo日志”。

Redo日志量大小与数据库的写负载有关，写入/更新频繁的数据库会产生较多的Redo日志。系统会定期上传Redo日志，随后清理本地Redo日志，清理前本地Redo日志会临时占用集群的存储空间，收取一定的费用，上传后就不再收取这部分费用了。

相关主题

[备份数据](#)

相关API

API	描述
#unique_34	创建POLARDB集群。  说明: 克隆集群时，参数CreationOption取值需要为CloneFromPolarDB。

14 诊断与优化

14.1 集群总览

本文介绍如何在POLARDB控制台查看集群总览。

功能介绍

POLARDB控制台提供集群总览功能，详情如下：

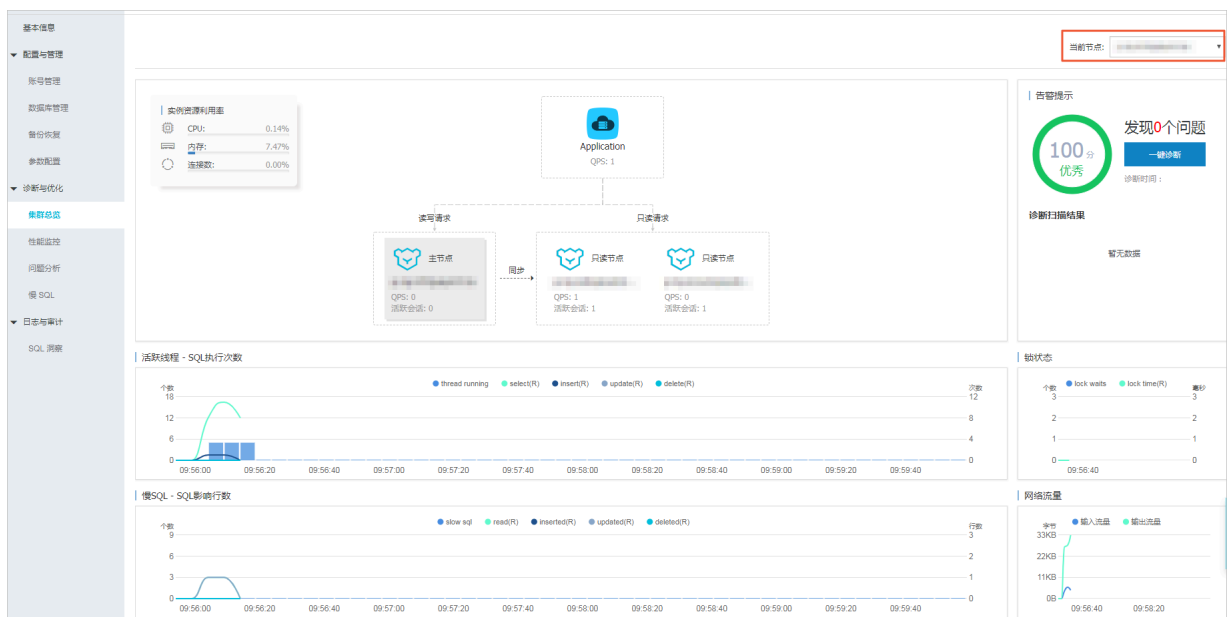
- 集群的链路拓扑。
- 节点的CPU、内存和连接数的利用率。
- 活跃线程 - SQL执行次数：包括thread running、select(R)、insert(R)、update(R)、delete(R)，可以自定义是否显示。
- 慢SQL - SQL影响行数：包括slow sql、read(R)、inserted(R)、updated(R)、deleted(R)，可以自定义是否显示。
- 锁状态：包括lock waits、lock time(R)，可以自定义是否显示。
- 网络流量：包括输入流量、输出流量，可以自定义是否显示。



说明：

上述的(R)表示该字段数值在右侧Y轴查看。

除了查看以上内容，控制台还提供一键诊断功能，检查节点的系统资源、系统状态、会话、慢SQL、事务和锁。



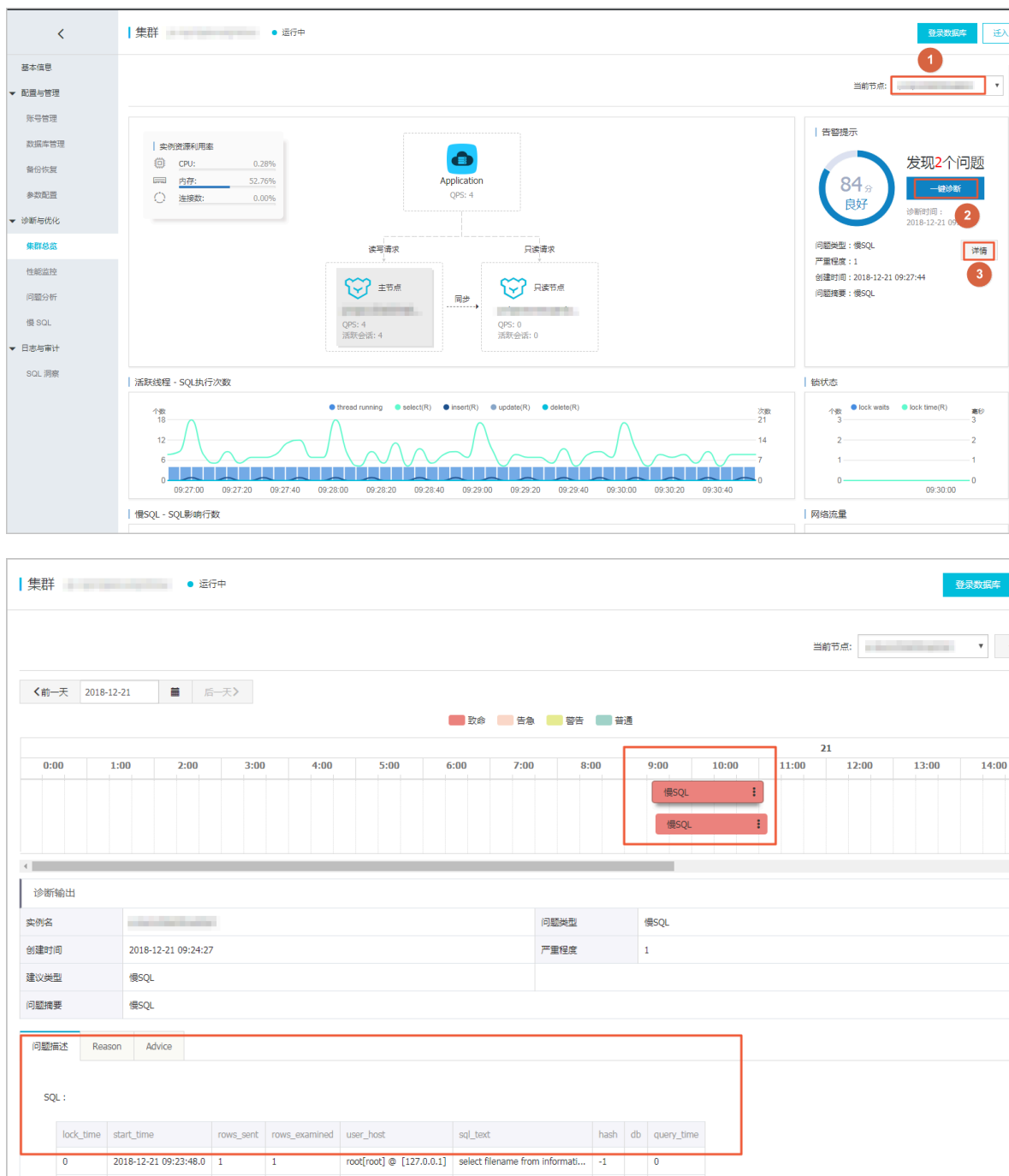
操作步骤

1. 登录[POLARDB控制台](#)。
2. 在控制台左上角，选择集群所在地域。
3. 单击目标集群ID。
4. 在左侧导航栏中，选择诊断与优化 > 集群总览。
5. 右侧选择想要总览和诊断的节点，单击下方一键诊断，等待诊断结果生成。



说明:

若节点诊断有问题，单击详情即可查看具体内容。



14.2 一键诊断

POLARDB for MySQL提供了一键诊断功能，该功能包括会话管理和实时性能展示，提供可视化的诊断结果供您查看。

前提条件

- 集群版本需要为POLARDB for MySQL5.6和MySQL8.0。
- 首次使用一键诊断功能需要对集群进行账号授权，参见[账号授权](#)。

账号授权

1. 登录POLARDB控制台。
2. 在控制台左上角，选择集群所在地域。
3. 单击目标集群ID。
4. 在左侧导航栏中，选择诊断与优化 > 一键诊断。

账号授权

* 数据库账号：

* 密码：

授权

5. 输入数据库账号和密码，单击授权。
6. 授权成功后，单击完成，进入会话管理页面。

会话管理

会话管理页面为您提供了查看会话、统计会话、分析SQL和优化SQL等功能。

1. 登录POLARDB控制台。
2. 在控制台左上角，选择集群所在地域。
3. 单击目标集群ID。
4. 在左侧导航栏中，选择诊断与优化 > 一键诊断。

会话管理									
实例会话（数据更新时间：2019年11月13日 14:08:28）距下次刷新还有5s 先选择会话，再单击优化；结束会话时，使用Shift键可选择多个会话									
当前节点 pc-bp1w2u46nf6c8ht5b (逻辑节点)									
优化	10秒SQL分析	刷新	导出全部会话	自动刷新	显示全部	搜索会话		结束选中会话	结束全部会话
ID	用户	主机	数据库名	命令	执行时间	状态	SQL	事务持续时间	
会话ID	root	192.168.1.100	testdb	SELECT	100ms	运行中	SELECT * FROM testdb		
会话ID	root	192.168.1.100	testdb	SELECT	100ms	运行中	SELECT * FROM testdb		
会话ID	root	192.168.1.100	testdb	SELECT	100ms	运行中	SELECT * FROM testdb		
会话ID	root	192.168.1.100	testdb	SELECT	100ms	运行中	SELECT * FROM testdb		
会话ID	root	192.168.1.100	testdb	SELECT	100ms	运行中	SELECT * FROM testdb		
会话ID	root	192.168.1.100	testdb	SELECT	100ms	运行中	SELECT * FROM testdb		

5. 您可以在会话管理页面执行如下操作：

- SQL优化

选中SQL列有SQL的会话，单击优化，可以查看到SQL诊断优化内容。



说明：

您也可以单击专家服务，付费购买数据库专家服务，为您提供云数据库产品本身能力范围之外的专业的数据库专家服务，主要包括紧急救援、健康诊断、性能调优、护航保障、Oracle迁移等。

实例会话 (数据更新时间: 2019-08-06 14:42:18) 距下次刷新还有26s 先选择会话，再单击优化；结束会话时，使用Shift键可连选多个会话

ID	用户	主机	数据库名	命令	执行时间	状态	SQL	事务持续时间
2150932			mysql_...	Sleep	19			
2150929			mysql_...	Query	3	Sending...	<code>select ...</code>	3

SQL诊断优化 (专家服务)

SQL

复制

执行计划

ID	select_type	table	extra	rows	possible_k...	key	key_len	ref
1	SIMPLE	company		945820				

诊断结果

诊断时间: 2019-08-05 16:25:11

确认 取消

• 10秒SQL分析

单击10秒SQL分析，系统会在10秒内每隔1秒执行一次`show processlist`，然后将所有的结果集进行统计分析，可以非常清晰地看到在这10秒内，哪些SQL执行的次数最多，是否存在慢SQL等等。



- 刷新会话

单击刷新可以立刻查询到当前的会话。



说明:

- 您可以打开自动刷新开关，每隔30秒会自动刷新一次。
- 默认显示的为活跃会话，您可以打开显示全部开关，显示所有会话。

- 导出全部会话

单击导出全部会话可以将会话信息保存到本地进行查看。

- 结束会话

- 选中会话并单击结束选中会话，在弹出的对话框中单击确定，可以将会话结束。



说明:

您可以按住Shift键选中多个会话。

- 单击结束全部会话可将所有会话结束。
- 查看会话统计

在下方的会话统计栏分类展示区域，您可以查看会话统计信息。

会话统计			
概要			
用户	统计项	统计值	操作
会话总数		7	
运行中会话总数		0	
运行中会话最长时间		0	
按用户统计(1)			
用户	活跃数	总数	操作
	0	7	
按访问来源统计(1)			
来源	活跃数	总数	操作
100.100.100.100 (9)	0	7	
按数据库统计(3)			
DB	活跃数	总数	操作
custom_info	0	3	
mysql_	0	3	
	0	1	

实时性能

您可以实时查看集群的QPS、TPS、连接数、网络流量等信息。

1. 登录[POLARDB控制台](#)。
2. 在控制台左上角，选择集群所在地域。
3. 单击目标集群ID。
4. 在左侧导航栏中，选择**诊断与优化** > **一键诊断**。
5. 单击**实时性能**页签。
6. 在实时性能页签，您可以查看集群实时性能。



说明：

详细的性能参数说明请单击左上方参数说明查看。



- 单击实时图表页签，可以查看实时图表。

实时图表会每5秒刷新一次，输出为折线图，供您查看系统性能的变化趋势。您可以在右上角查看到剩余刷新次数。

- 单击实时表格页签，可以查看实时表格。

表格显示更详细的性能参数，每5秒新增一行实时的性能值，最多显示999行。

14.3 问题分析

14.3.1 查询和终止实时会话

您可以查看并终止所选择的节点的实时会话信息，还可以诊断会话和查询会话诊断历史。本文将介绍如何诊断和终止当前实例的实时会话。

操作步骤

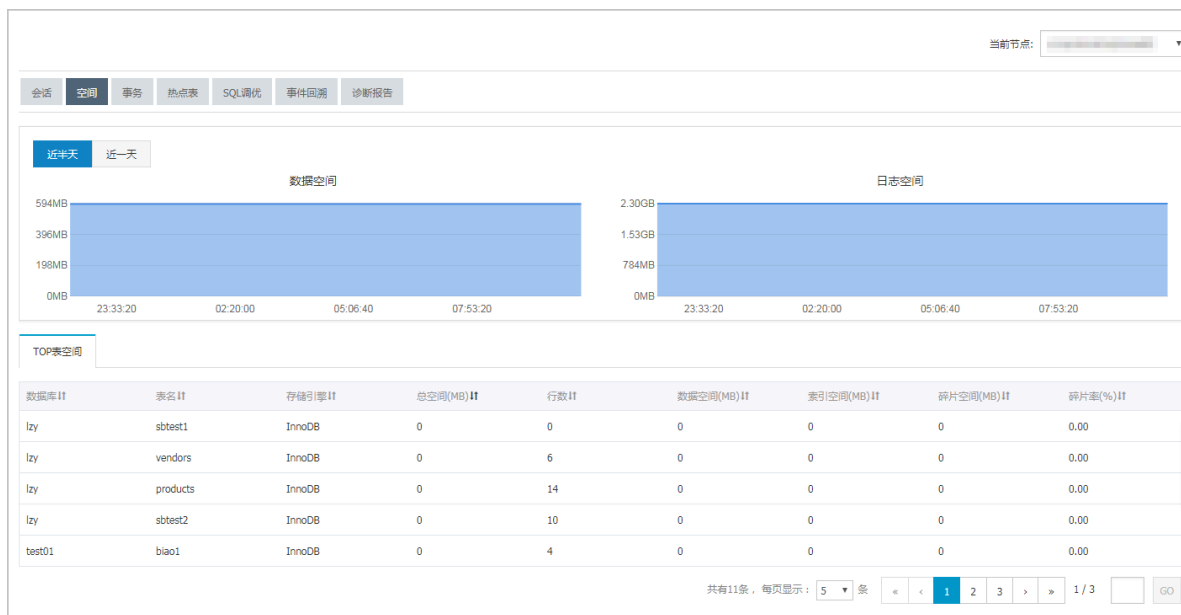
1. 进入[POLARDB控制台](#)。
2. 选择地域。
3. 找到目标集群，单击集群名称列的集群ID。
4. 在左侧导航栏中选择[诊断与优化 > 问题分析](#)。
5. 选择会话页签，在下方列表可以看到会话，勾选后单击终止会话。

14.3.2 查看空间使用详情

您可以查看节点数据空间和日志空间的大小，还可以查看表空间的TOP排名。本文将介绍如何查看空间的使用详情。

操作步骤

1. 进入[POLARDB控制台](#)。
2. 选择地域。
3. 找到目标集群，单击集群名称列的集群ID。
4. 在左侧导航栏中选择诊断与优化 > 问题分析。
5. 选择空间页签，可以看到数据空间和日志空间的大小，以及TOP表空间详情。



14.3.3 查看事务

本文将介绍如何查看事务。您可以设置时间段创建分析SQL日志的任务，并查看SQL记录数和事务分析。

操作步骤

1. 登录[POLARDB控制台](#)。
2. 在控制台左上角，选择集群所在地域。
3. 单击目标集群ID。
4. 在左侧导航栏中，选择诊断与优化 > 问题分析。
5. 选择事务页签，设置开始时间和时长，单击分析SQL日志。

预期结果

等分析结果生成后，即可在事务分析列查看具体内容。

14.3.4 诊断热点表

您可以查看并诊断节点中所有的热点表，根据诊断结果会返回相应的优化建议。本文将介绍如何诊断热点表。

操作步骤

1. 进入[POLARDB控制台](#)。
2. 选择地域。
3. 找到目标集群，单击集群名称列的集群ID。
4. 在左侧导航栏中选择诊断与优化 > 问题分析。
5. 选择热点表页签，单击创建诊断。

14.3.5 SQL调优

本文将介绍如何使用SQL调优功能优化和执行SQL语句。

背景信息

SQL调优功能可以根据您输入的SQL语句，提出优化建议。您也可以直接在SQL调优功能中登录数据库，并使用SQL命令进行插入和管理数据的操作。

操作步骤

1. 登录[POLARDB控制台](#)。
2. 在控制台左上角，选择集群所在地域。
3. 单击目标集群ID。
4. 在左侧导航栏中，选择诊断与优化 > 问题分析。
5. 选择SQL调优页签，单击右侧登录数据库。

6. 填写登录信息后单击登录。

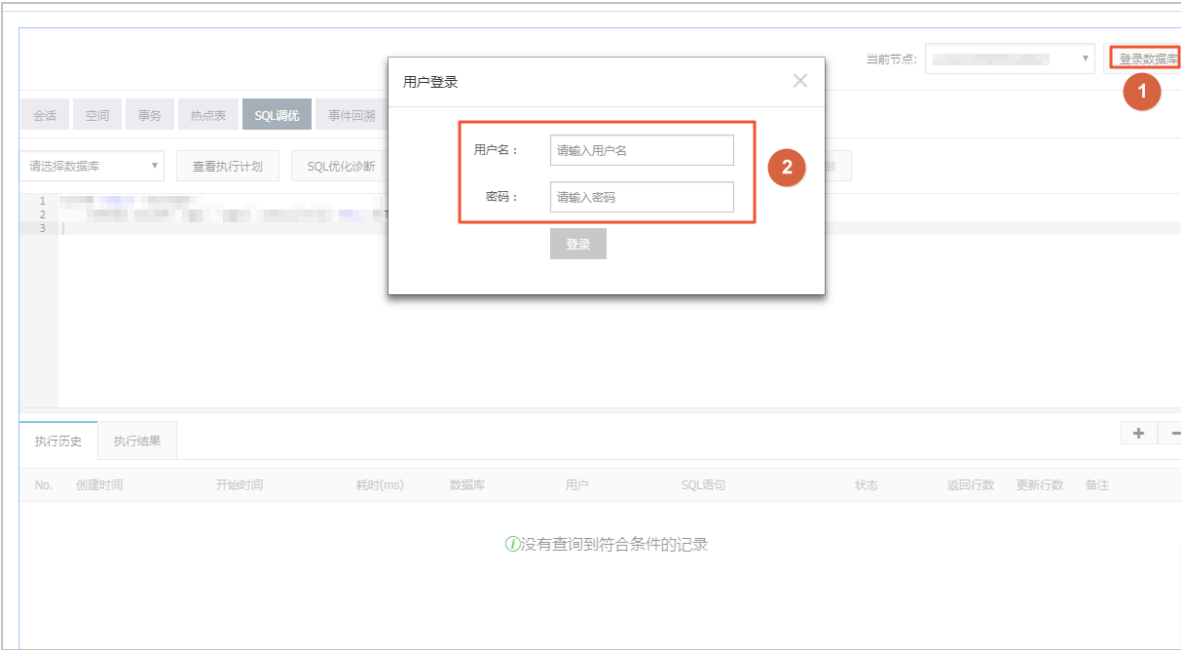


表 14-1: 参数说明

参数名称	说明
用户名	有管理相应数据库权限的账号名称。
密码	登录数据库的账号对应的密码。

7. 选择要查询或管理的数据库。



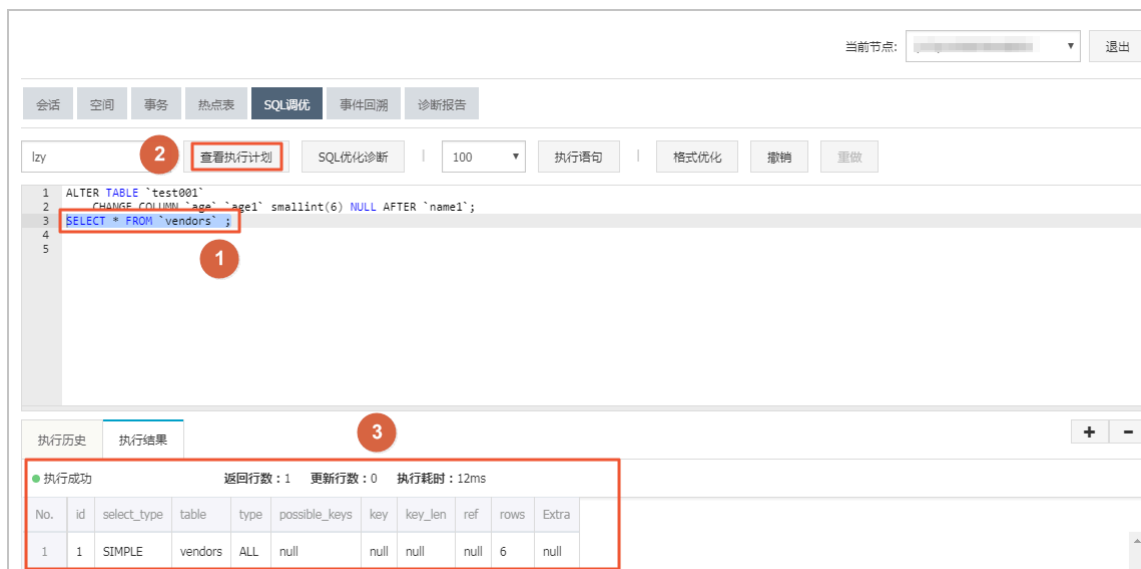
8. 在输入框中填写SQL语句，然后选择如下操作。



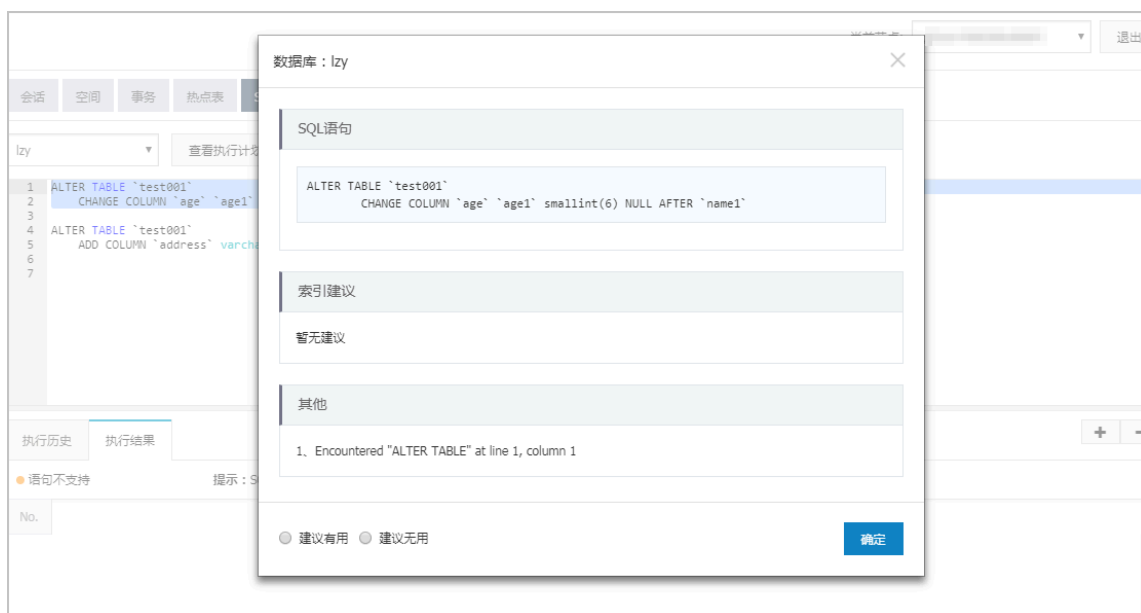
说明:

SQL操作中提供的所有功能都不支持批量操作，若您同时输入了多条SQL语句，只能选中一条目标语句，进行后续操作。

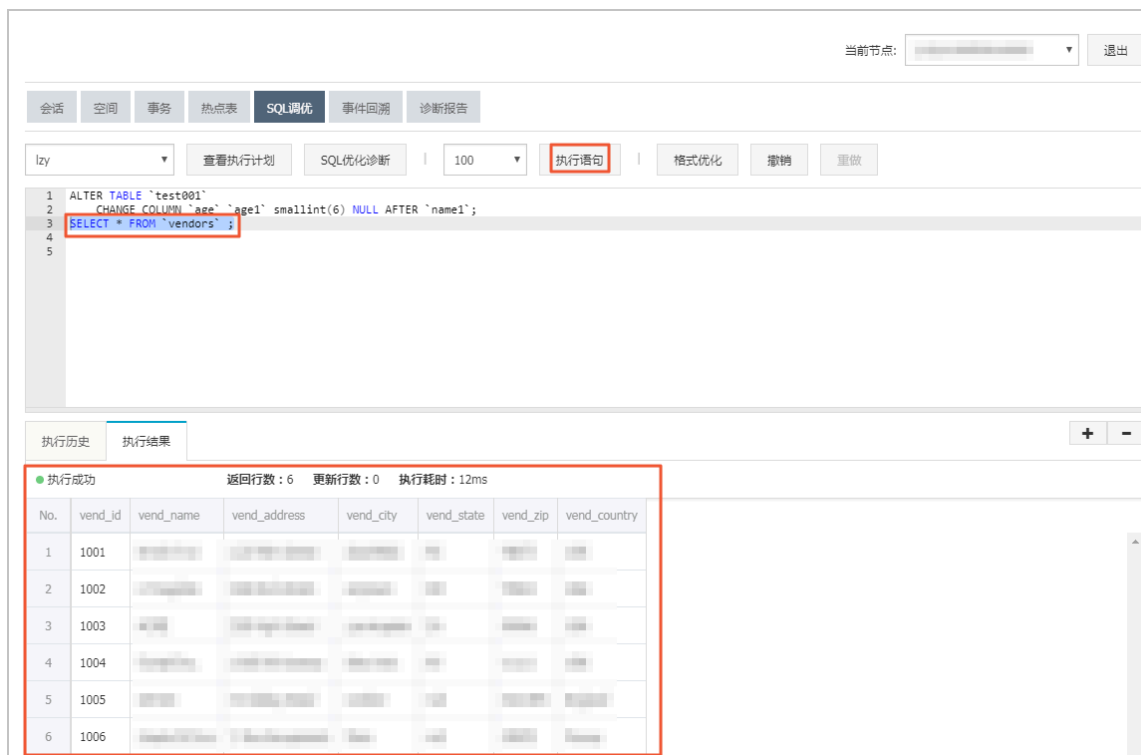
- 单击查看执行计划，即可在执行结果中查看SQL语句具体的执行计划。



- 单击SQL优化诊断，系统会对所输入的SQL语句进行诊断并给出优化建议，如索引优化。



- 单击执行语句并在弹出的对话框中单击确认，即可在已选数据库中执行SQL命令，可在执行结果中查看SQL执行结果。



- 单击格式优化，系统会自动优化所输入SQL语句的格式。
- 单击撤销，可以撤销上一步对SQL语句进行的修改。若您误撤销了上一步的操作，可以立刻单击重做，即可恢复被撤销的修改。

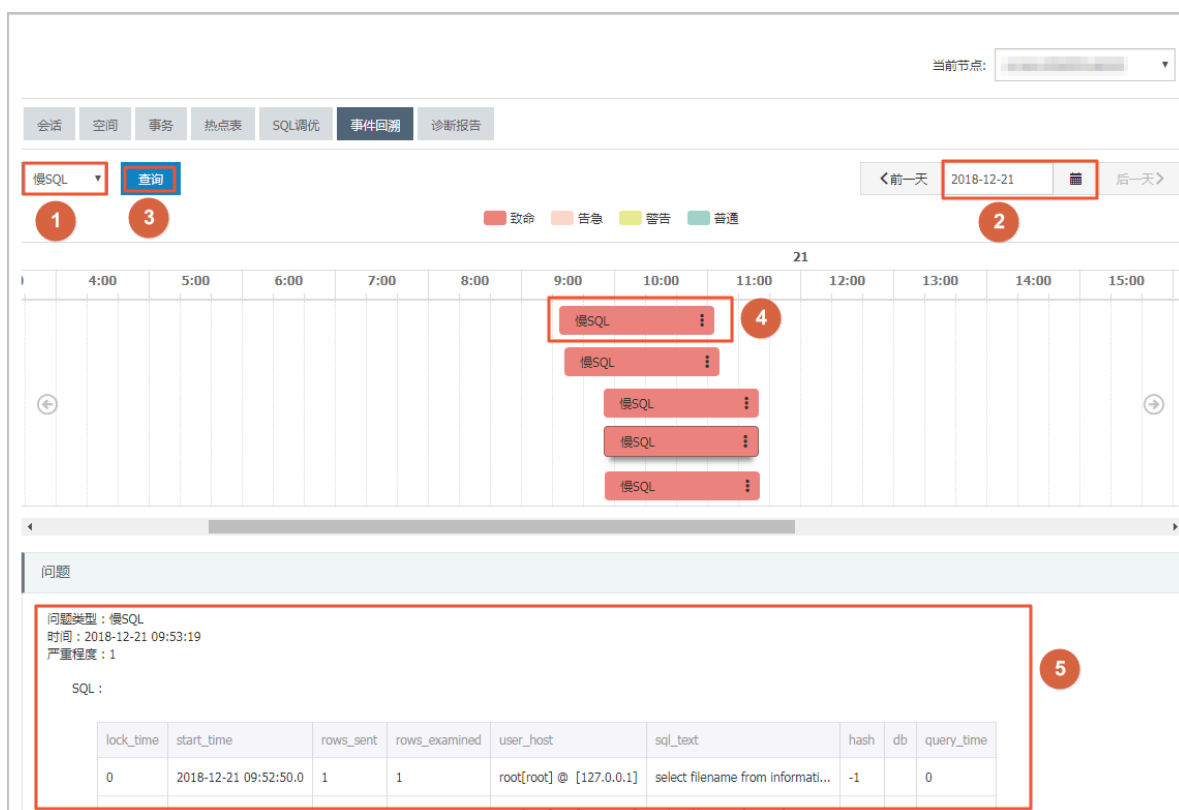
14.3.6 事件回溯

事件回溯功能可以统计所有类型的问题信息并分类展示。本文将介绍如何使用事件回溯功能查看各类型问题。

操作步骤

1. 进入[POLARDB控制台](#)。
2. 选择地域。
3. 找到目标集群，单击集群名称列的集群ID。
4. 在左侧导航栏中选择诊断与优化 > 问题分析。
5. 选择事件回溯页签，选择想要查看的问题类型以及日期，单击查询。

6. 单击问题标签，即可在下方看到具体的问题详情。



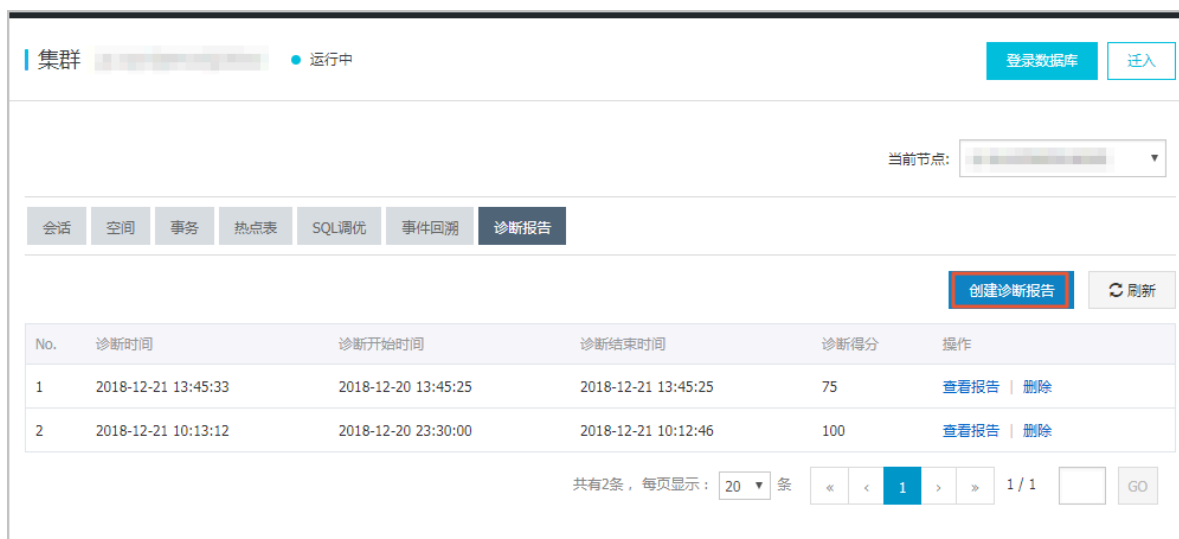
14.3.7 查看节点性能诊断报告

诊断报告中会列出节点中存在的问题详情及问题等级，并给出相应的优化建议，使您能够对节点有一个全面的了解并针对问题进行相应优化以提升节点性能。诊断报告中会包含节点基本信息、节点性能综合评分、节点核心指标（CPU、内存、空间、IOPS、连接数）的状态和问题、SQL分析等内容。本文将介绍如何创建、查看节点性能诊断报告。

操作步骤

1. 进入[POLARDB控制台](#)。
2. 选择地域。
3. 找到目标集群，单击集群名称列的集群ID。
4. 在左侧导航栏中选择诊断与优化 > 问题分析。

5. 选择诊断报告页签，单击右侧创建诊断报告。



6. 设置诊断报告的开始和结束时间，单击创建报告，等待报告生成。

7. 报告生成后，在操作列单击查看报告，在弹出的对话框中单击点击查看。

14.4 慢SQL

POLARDB for MySQL提供慢SQL分析功能，能够查看慢日志趋势和统计信息，并且提供SQL建议和诊断分析。

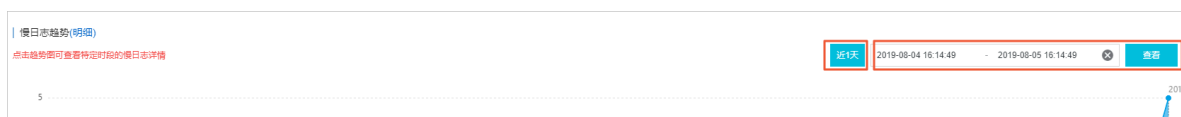
查看慢日志趋势和慢日志统计

1. 登录POLARDB控制台。
2. 在控制台左上角，选择集群所在地域。
3. 单击目标集群ID。
4. 在左侧导航栏中，选择诊断与优化 > 慢SQL。
5. 您可以通过以下两种方式查看慢日志趋势：
 - 单击近1天，查看最近24小时的慢日志情况。
 - 设置查询起止日期，然后单击查看，可以查看最多7天的慢日志情况。



说明：

默认展示近1天的慢日志情况。



6. 查看慢日志趋势折线图，单击慢SQL发生的时间节点，即可在下方看到慢日志统计。



7. 您可以通过如下两种方式排查慢SQL问题：

- 单击目标慢SQL操作列的样本，可查看该次慢SQL的详细信息。

慢日志样本

执行时间	SQL模板	库名	客户端	用户	执行耗时 (秒)	锁等待耗时 (秒)	扫描行	返回行
2019-08-05 16:11:34	select * from company where n...		110		2.0	0.0	854.9K	313.8K

关闭

- 单击目标慢SQL操作列的优化，可查看诊断优化信息。

SQL诊断优化 (专家服务)

SQL

复制

执行计划

ID	select_type	table	extra	rows	possible_k...	key	key_len	ref
1	SIMPLE	company		945820				

诊断结果

诊断时间: 2019-08-05 16:25:11

确认 取消

说明:

您也可以单击专家服务，付费购买数据库专家服务，为您提供云数据库产品本身能力范围之外的专业的数据库专家服务，主要包括紧急救援、健康诊断、性能调优、护航保障、数据迁移等。

导出慢日志

您可以单击导出慢日志将慢日志信息保存到本地查看。

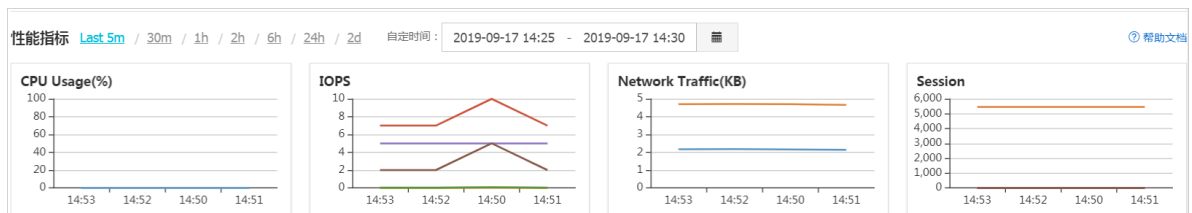
慢日志统计 (时间范围: 2019-08-05 16:00:00 - 2019-08-05 17:00:00)											导出慢日志
① 当1个小时慢日志数量多于15000条时,仅显示前15000条慢日志统计											×
SQL模板	库名	执行次数	平均执行时间(秒)	最大执行时间(秒)	平均锁等待时间(秒)	最大锁等待时间(秒)	平均扫描行	最大扫描行	平均返回行	最大返回行	操作
SELECT * FROM company ...		1	2.00	2	0.00	0	854.9K	855K	313.84K	314K	样本 优化
SELECT * FROM company ...		1	1.00	1	0.00	0	854.9K	855K	103.63K	104K	样本 优化

14.5 性能洞察

性能洞察（Performance Insight）是POLARDB诊断与优化里专注于集群负载监控、关联分析、性能调优的利器，以简单直观的方式帮助您迅速评估数据库负载，找到性能问题的源头，提升数据库的稳定性。

操作步骤

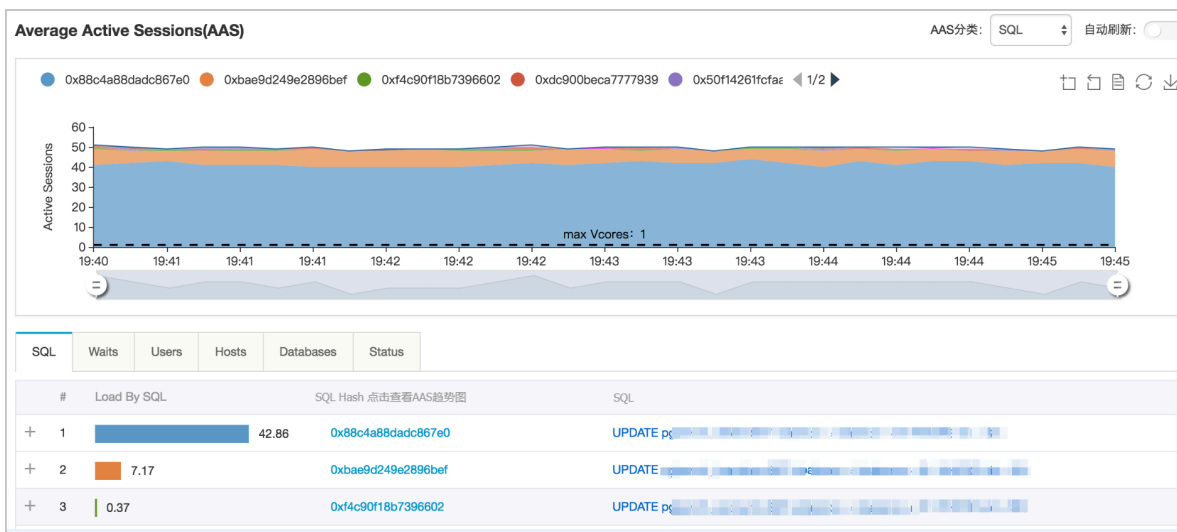
1. 登录POLARDB控制台。
2. 在页面左上角，选择集群所在地域。
3. 单击目标集群ID。
4. 在左侧导航栏中，选择诊断与优化 > 性能洞察。
5. 查看性能指标趋势图，确认数据库的负载情况。



说明:

- 性能指标包括CPU使用率、每秒读写次数（IOPS）、网络流量和会话数。
- 查询时间只能为最近7天，且时间间隔小于2天。

6. 查看平均活跃会话（Average Active Sessions）变化趋势图及相关多维负载信息，确定性能问题源头。



说明:

以上图为例，可以直观地看到在选定的时间段内，标识为蓝色的SQL占用了最多的活跃会话资源，还能看到SQL模板和样本。您也可以切换到其他页签（Waits、Users、Hosts、Database、Status）查看到更多信息帮助您确定问题源头。

15 性能监控与报警

为方便您掌握实例的运行状态，POLARDB控制台提供了丰富的性能监控项。

性能监控

1. 登录[POLARDB控制台](#)。
 2. 在控制台左上角，选择集群所在地域。
 3. 单击目标集群ID。
 4. 在左侧导航栏中，选择诊断与优化 > 性能监控。
 5. 根据您的需求，可以查看集群或节点的监控信息。详细说明请参见[监控项说明](#)。
- 集群性能监控：单击集群，在右侧设置时间段后单击确定。



- 节点性能监控：单击节点，在右侧选择节点并设置时间段后单击确定。



说明:

节点页面下方单击显示更多，会显示更多监控项。



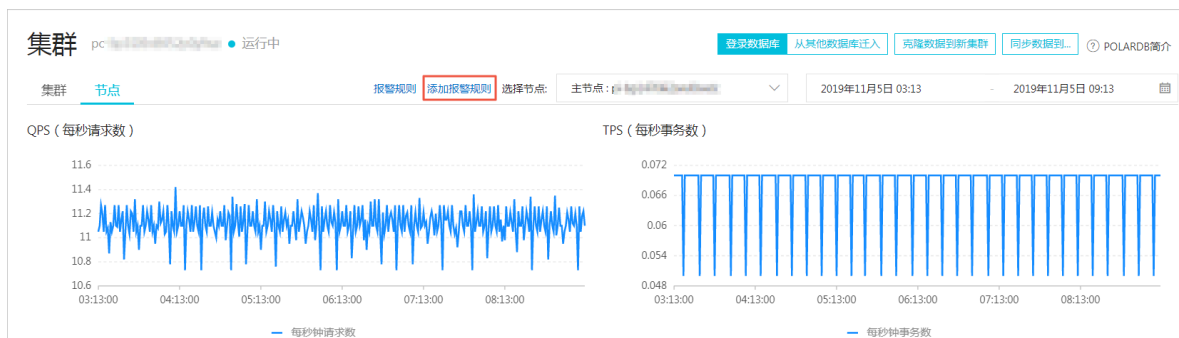
监控项说明

类别	监控项	说明
集群	存储空间	展示本地Binlog、Redolog和其他日志文件的大小，以及数据空间、系统空间和临时空间的使用量。
	QPS	展示各节点的每秒请求数。
	TPS	展示各节点的每秒事务数。
	CPU	展示各节点的CPU使用率。
	内存	展示各节点的内存使用率。
节点	QPS	展示所选择节点的每秒请求数。
	TPS	展示所选择节点的每秒事务数。

类别	监控项	说明
	CPU	展示所选择节点的CPU使用率。
	内存	展示所选择节点的内存使用率。
	连接	展示所选择节点的当前总连接数和活跃连接数。
	操作	展示所选择节点的每秒操作数，包括DELETE、INSERT、UPDATE、REPLACE等等。
	内存缓冲池	展示所选择节点的缓冲池的脏块率、读命中率、利用率。
	I/O吞吐量	展示所选择节点的总I/O吞吐量、读I/O吞吐量、写I/O吞吐量。
	IOPS	展示所选择节点的每秒读写次数，包括每秒读写总次数、每秒读次数、每秒写次数。
	网络	展示所选择节点的每秒输入输出流量。
	扫描行数	展示所选择节点每秒插入、读取、更新、删除的行数。
	InnoDB读写数据量	展示所选择节点每秒从存储引擎读取的数据量和每秒往存储引擎写入的数据量。
	InnoDB缓冲池请求数	展示所选择节点缓冲池的每秒读写次数。
	InnoDB日志写	展示所选择节点日志的每秒写请求数量和每秒同步数据到磁盘的次数。
	临时表	展示所选择节点每秒创建的临时表数量。

添加报警规则

1. 登录POLARDB控制台。
2. 在控制台左上角，选择集群所在地域。
3. 单击目标集群ID。
4. 在左侧导航栏中，选择诊断与优化 > 性能监控。
5. 单击添加报警规则。



6. 在创建报警规则页面，选择资源范围，设置报警规则和通知方式后，单击确认即可。



说明:

如果资源范围选择集群，还需要选择集群所在地域和目标集群。更多报警规则参数说明，请参见[#unique_121](#)。

创建报警规则 [← 返回](#)

1 关联资源

产品: 云数据库POLARDB-MySQL (新版)

资源范围: 集群

地域: 华东1 (杭州)

集群: 监控测试

2 设置报警规则

规则名称: CPU跑高报警

规则描述: CPU使用率 5分钟周期 连续3周期 平均值 >= 70 %

+ 添加报警规则

通道沉默周期: 24 小时

生效时间: 00:00 至 23:59

成功添加报警规则后，当集群被监控的指标超过设定阈值后自动发送报警通知给指定的联系人。

报警规则管理

1. 登录[POLARDB控制台](#)。
2. 在控制台左上角，选择集群所在地域。
3. 单击目标集群ID。
4. 在左侧导航栏中，选择诊断与优化 > 性能监控。
5. 单击报警规则，进入报警规则列表页面。

报警规则列表 刷新								
阈值报警 事件报警								
创建报警规则 请输入进行查询 搜索								
规则名称	状态 (全部)	启用	监控项 (全部)	维度 (全部)	报警规则	产品名称 (云数据库POLARDB)	通知对象	操作
CPU跑高报警	正常状态	已启用	CPU使用率	clusterId:pc-xxxxxx	CPU使用率 >= 50 Warn 连续3次就报警	云数据库POLARDB-MySQL (新版)	POLARDB报警联系人组 查看	查看 报警历史 修改 禁用 删除
shukun测试	正常状态	已启用	内存使用率	clusterId:pc-xxxxxx	内存使用率 >= 5 Warn 连续1次就报警	云数据库 POLARDB	shukun云账号报警联系人 查看	查看 报警历史 修改 禁用 删除
启用 禁用 删除 共 2条 10 < < 1 > >								

6. 在报警规则列表页面，您可以进行报警规则管理，具体操作如下。

- 单击操作栏中的查看，可查看报警规则的基本信息。
- 单击操作栏中的报警历史，可查看报警历史。
- 单击操作栏中的修改，可修改报警规则。
- 单击操作栏中的禁用，可禁用报警规则。
- 单击操作栏中的删除，可删除报警规则的。
- 单击通知对象栏中的查看，可查看报警联系组、报警联系人及报警通知方式。

相关API

API	描述
#unique_122	查询POLARDB集群的性能数据。
#unique_123	查询POLARDB集群节点的性能数据。

16 SQL洞察

SQL洞察功能为您的数据库提供安全审计、性能诊断等增值服务。

费用说明

- 试用版：免费使用，审计日志仅保存一天，即只能查询一天范围内的数据；不支持数据导出等高级功能；不保障数据完整性。
- 30天或以上：详情请参见[#unique_20](#)。

功能说明

- SQL审计日志。

记录对数据库执行的所有操作。通过审计日志记录，您可以对数据库进行故障分析、行为分析、安全审计等操作。

- 增强搜索。

可以按照数据库、用户、客户端IP、线程ID、执行耗时、执行状态等进行多维度检索，并支持导出和下载搜索结果。

SQL洞察

服务设置

搜索

分析

设置查询条件

关键字

可多字段组合查询，字段间以空格分隔

or

▼

时间范围

2018-12-21 12:57:55 - 2018-12-21 13:12:55

自定义

▼

用户

可组合查询，如：user1 user2 user3

数据库

可组合查询，如：DB1 DB2 DB3

操作类型

☐ SELECT ☐ INSERT ☐ UPDATE ☐ DELETE

更多

客户端IP

可组合查询，如：IP1 IP2 IP3

线程ID

可组合查询，如：ThreadId1 ThreadId2 ThreadId3

执行状态

☐ 成功 ☐ 失败

执行耗时

-

扫描记录数

-

关闭高级查询

查询

日志列表

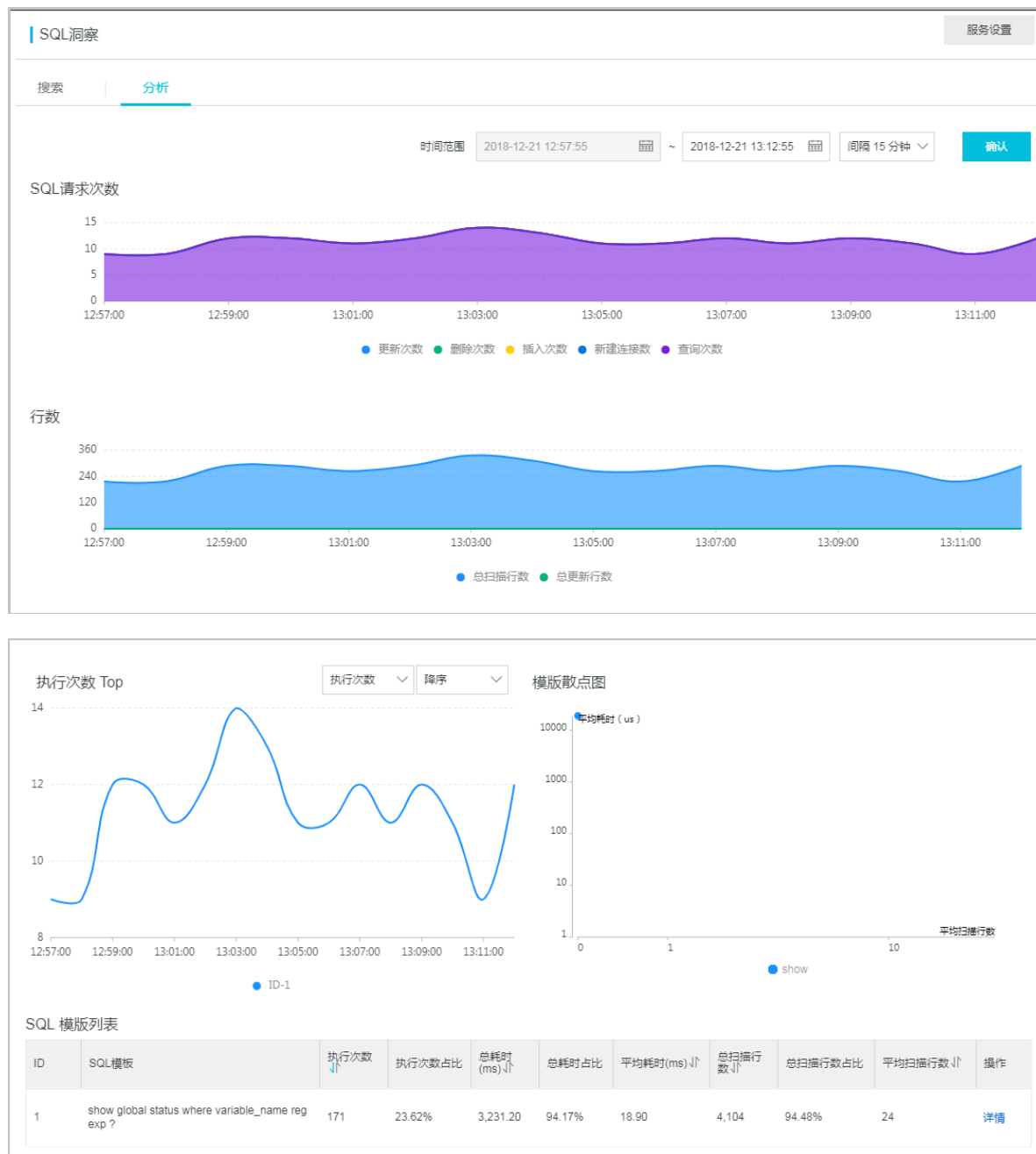
导出

查看导出列表

SQL语句	数据库	线程ID	用户	客户端IP	操作	状态	耗时 (ms) ↓↑	执行时间 ↓↑	更新行 数 ↓↑	扫描行 数 ↓↑
SQL语句: SELECT * FROM ...										

· SQL分析。

可以对指定时间段的SQL日志进行可视化交互式分析，找出异常SQL，定位性能问题。



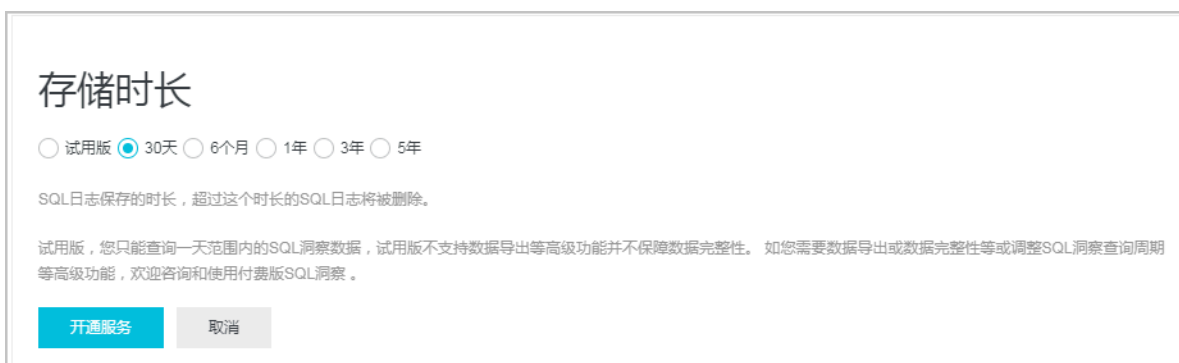
开通SQL洞察

1. 登录POLARDB控制台。
2. 在控制台左上角，选择集群所在地域。
3. 单击目标集群ID。
4. 在左侧导航栏中，选择日志与审计 > SQL洞察。

5. 单击立即开通。



6. 选择SQL审计日志的保存时长，单击开通服务。



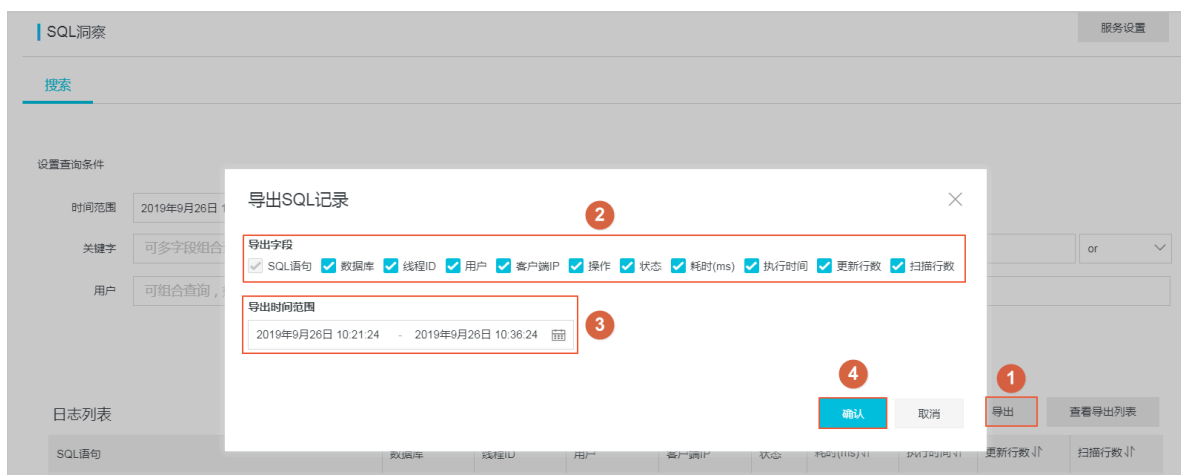
修改SQL日志的存储时长

1. 登录[POLARDB控制台](#)。
2. 在控制台左上角，选择集群所在地域。
3. 单击目标集群ID。
4. 在左侧导航栏中，选择日志与审计 > SQL洞察。
5. 单击右上角服务设置。
6. 修改存储时长并单击确认。

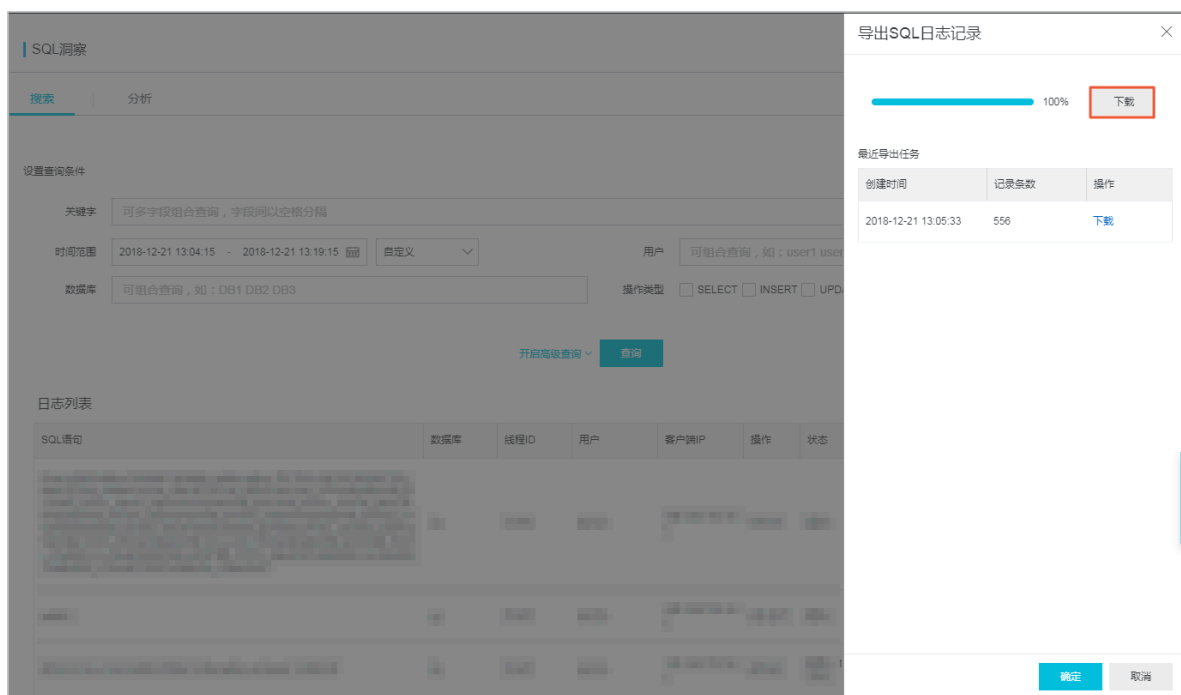
导出SQL记录

1. 登录[POLARDB控制台](#)。
2. 在控制台左上角，选择集群所在地域。
3. 单击目标集群ID。
4. 在左侧导航栏中，选择日志与审计 > SQL洞察。
5. 单击右侧导出。

6. 在弹出的对话框中，勾选需要导出的选项，单击确认。



7. 导出完成后，在导出SQL日志记录中，下载已导出的文件并妥善保存。



关闭SQL洞察



说明:

SQL洞察功能关闭后，SQL审计日志会被清空。请将SQL审计日志导出后，再关闭SQL洞察功能。

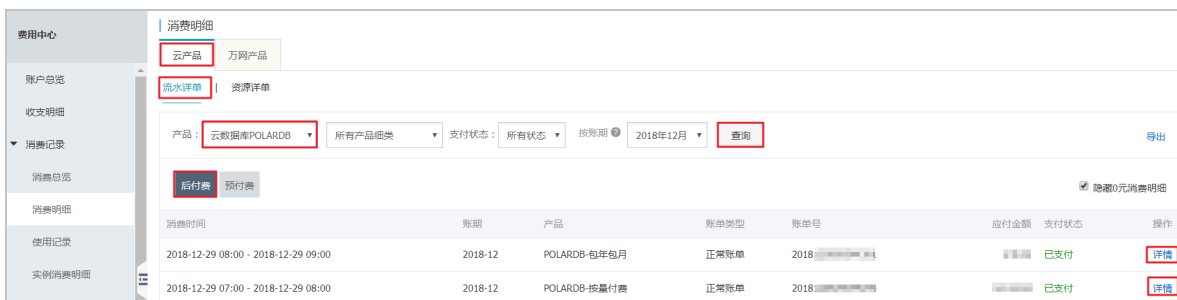
1. 登录POLARDB控制台。
2. 在控制台左上角，选择集群所在地域。
3. 单击目标集群ID。
4. 在左侧导航栏中，选择日志与审计 > SQL洞察。
5. 单击服务设置。

6. 单击滑块关闭SQL洞察。



查看审计日志的大小和消费明细

1. 登录[阿里云管理控制台](#)。
2. 在页面右上角，选择费用 > 进入费用中心。
3. 在左侧导航栏中，选择消费记录 > 消费明细。
4. 选择云产品页签。



5. 选择流水详单。
6. 选择后付费。
7. 设置查询条件，然后单击查询。



说明：

若要查询超过12个月前的记录，请提交工单。

8. 根据POLARDB集群的计费方式，找到POLARDB-按量付费或POLARDB-包年包月，单击最右侧的详情。

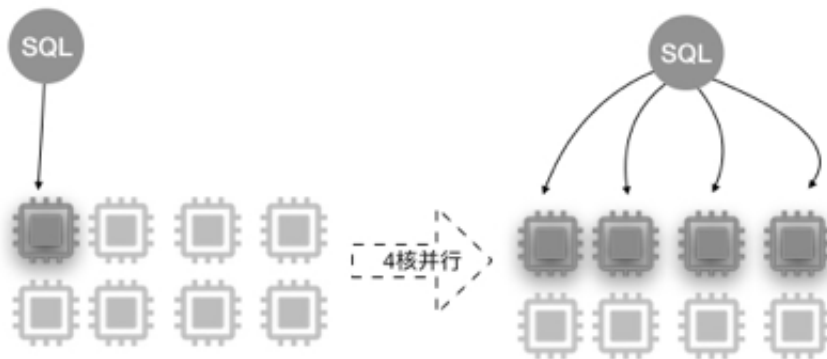
17 并行查询

17.1 并行查询

POLARDB for MySQL 8.0重磅推出并行查询框架，当您的查询数据量到达一定阈值，就会自动启动并行查询框架，从而使查询耗时指数级下降。

在存储层将数据分片到不同的线程上，多个线程并行计算，将结果流水线汇总到总线程，最后总线程做些简单归并返回给用户，提高查询效率。

并行查询（Parallel Query）利用多核CPU的并行处理能力，以8核 32G 配置为例，示意图如下所示：



应用场景

并行查询适用于大部分SELECT语句，例如大表查询、多表连接查询、计算量较大的查询。对于非常短的查询，效果不太显著。

- 轻分析类业务

报表查询通常SQL复杂而且比较耗费时间，通过并行查询可以加速单次查询效率。

- 系统资源相对空闲

并行查询会使用更多的系统资源，只有当系统的CPU较多、IO负载不高、内存够大的时候，才可以充分使用并行查询来提高资源利用率和查询效率。

性能优势

关于并行查询的性能优势，请参见[并行查询使用示例](#)。

并行查询的使用方法

- 通过系统参数来控制并行查询

POLARDB通过Global全局参数 `max_parallel_degree` 来控制每一条SQL最多使用多少个线程并行执行，默认值是4，您可以在使用过程中随时修改该参数（参见[设置集群参数](#)），无需重启数据库。

除了Global集群级别，您也可以单独调整某Session内SQL查询的并行度。例如，通过Session级环境变量，加到JDBC的连接串配置中，则可以对某个应用程序单独设置并行度。

```
set max_parallel_degree = n
```

- 使用 Hint

使用Hint开启并行查询（n代表最大的并行度，也就是最大的worker个数）：

```
SELECT /*+ SET_VAR(max_parallel_degree=n) */ * FROM ...
```



说明：

为了发挥最佳性能，并行查询并不是对所有的SQL都生效，POLARDB优化器会根据具体的查询SQL生成合适的执行计划。

- 强制优化器选择并行执行

POLARDB优化器可能不选择并行执行查询，如果希望优化器忽略代价，尽可能选择并行计划，可以设置如下参数：

```
set force_parallel_mode = on
```



说明：

这是一个调试参数，不建议在生产环境中使用。由于并行查询使用场景的限制，有些情况下即便设置了该参数，优化器也可能不选择并行。

相关参数和变量

表 17-1: 系统参数

参数名	级别	描述
max_parallel_degree	Global、Session	单个查询的最大并行度，即最多使用多少个 worker 进行并行执行。 - 取值范围：[0-1024] - 默认值：0，表示关闭并行计算。  说明： POLARDB 优化器可能会对主查询和子查询分别并行执行，如果同时并行执行，它们的最大 worker 数不能超过 max_parallel_degree，整个查询使用的 worker 数将会是主查询和子查询使用的 worker 数之和。
force_parallel_mode	Session	强制 POLARDB 优化器忽略代价，尽可能的使用并行查询。 - 取值范围：[ON, OFF] - 默认值：OFF  注意： 这是个调试参数，打开后，POLARDB 优化器会尽可能的选择并行查询，但不能保证一定使用并行查询。

表 17-2: 状态变量

变量名	级别	描述
Parallel_workers_created	Session、Global	从 Session 启动开始，生成 Parallel Worker 的个数。
Gather_records	Session、Global	Gather 记录总数。
PQ_refused_over_memory_soft_limit	Session、Global	由于内存限制没有启用并行的查询数。

变量名	级别	描述
PQ_refused_over_total_workers	Session、Global	由于总Worker数限制没有启用并行的查询数。
Total_used_query_memory	Global	当前总的已使用的查询内存（Virtual Memory）。
Total_running_parallel_workers	Global	当前正在运行的Parallel Worker的数目。

并行执行计划

以下为您介绍EXPLAIN执行计划输出中与并行查询相关的内容。

• 并行扫描

在并行扫描中，每个Worker并行独立扫描数据表中的数据。Worker扫描产生的中间结果集将会返回给Leader线程，Leader线程通过Gather操作收集产生的中间结果，并将所有结果汇返回到客户端。

• 多表并行连接

并行查询会将多表连接操作完整的下推到Worker上去执行。POLARDB优化器只会选择一个自认为最优的表进行并行扫描，而除了该表外，其他表都是一般扫描。每个Worker会将连接结果集返回给Leader线程，Leader线程通过Gather操作进行汇总，最后将结果返回给客户端。

• 并行排序

POLARDB优化器会根据查询情况，将ORDER BY下推到每个Worker里执行，每个Worker将排序后的结果返回给Leader，Leader通过Gather Merge Sort操作进行归并排序，最后将排序结果返回到客户端。

• 并行分组

POLARDB优化器会根据查询情况，将GROUP BY下推到Worker上去并行执行。每个Worker负责部分数据的GROUP BY。Worker会将GROUP BY的中间结果返回给Leader，Leader通过Gather操作汇总所有数据。这里POLARDB优化器会根据查询计划情况来自动识别是否需要再次在Leader上进行GROUP BY。例如，如果GROUP BY使用了Loose Index Scan，Leader上将不会进行再次GROUP BY；否则Leader会再次进行GROUP BY操作，然后把最终结果返回到客户端。

• 并行聚集

并行查询执行聚集函数下推到Worker上并行执行。并行聚集是通过两次聚集来完成的。第一次，参与并行查询部分的每个Worker执行聚集步骤。第二次，Gather或Gather Merge节点

将每个Worker产生的结果汇总到Leader。最后，Leader会将所有Worker的结果再次进行聚集得到最终结果。

并行执行计划示例

以下以使用pq_test表来测试并行查询为例。

表结构如下：

```
mysql> SHOW CREATE TABLE pq_test\G
***** 1. row *****
      Table: pq_test
Create Table: CREATE TABLE `pq_test` (
  `id` BIGINT(20) NOT NULL AUTO_INCREMENT,
  `help_topic_id` INT(10) UNSIGNED NOT NULL,
  `name` CHAR(64) NOT NULL,
  `help_category_id` SMALLINT(5) UNSIGNED NOT NULL,
  `description` TEXT NOT NULL,
  `example` TEXT NOT NULL,
  `url` TEXT NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=21495809 DEFAULT CHARSET=utf8
1 row in set (0.00 sec)
```

表大小如下：

```
mysql> SHOW TABLE STATUS\G
***** 1. row *****
      Name: pq_test
      Engine: InnoDB
      Version: 10
      Row_format: Dynamic
      Rows: 20064988
      Avg_row_length: 1898
      Data_length: 38085328896
      Max_data_length: 0
      Index_length: 0
      Data_free: 4194304
      Auto_increment: 21495809
      Create_time: 2019-07-30 01:35:27
      Update_time: NULL
      Check_time: NULL
      Collation: utf8_general_ci
      Checksum: NULL
      Create_options:
      Comment:
1 row in set (0.02 sec)
```

查询SQL：

```
SELECT COUNT(*) FROM pq_test;
```

• 不使用并行查询的EXPLAIN输出如下：

```
mysql> SET max_parallel_degree=0; EXPLAIN SELECT COUNT(*) FROM
pq_test\G
Query OK, 0 rows affected (0.02 sec)
***** 1. row *****
```

```

      Id: 1
    Select_type: SIMPLE
      Table: pq_test
    Partitions: NULL
      Type: index
Possible_keys: NULL
      Key: PRIMARY
     Key_len: 8
       Ref: NULL
      Rows: 20064988
    Filtered: 100.00
     Extra: Using index
1 row in set, 1 warning (0.03 sec)

```

- 使用并行查询的EXPLAIN输出如下：

```

mysql> EXPLAIN SELECT COUNT(*) FROM pq_test\G
***** 1. row *****
      Id: 1
    Select_type: SIMPLE
      Table: <gather>
    Partitions: NULL
      Type: ALL
Possible_keys: NULL
      Key: NULL
     Key_len: NULL
       Ref: NULL
      Rows: 20064988
    Filtered: 100.00
     Extra: NULL
***** 2. row *****
      Id: 2
    Select_type: SIMPLE
      Table: pq_test
    Partitions: NULL
      Type: index
Possible_keys: NULL
      Key: PRIMARY
     Key_len: 8
       Ref: NULL
      Rows: 10032494
    Filtered: 100.00
     Extra: Parallel scan (2 workers); Using index
2 rows in set, 1 warning (0.00 sec)

```

从该EXPLAIN可以看到，并行计划中包含Gather操作，该操作负责汇总所有Worker返回的中间结果。另外，从执行计划输出的Extra信息中看到：pq_test表使用了Parallel scan（并行扫描），期望用2个Workers来并行执行。

17.2 并行查询使用示例

本文以TPC-H为例，为您介绍并行查询使用示例。案例中所有示例，使用的数据量是TPC-H中SF=100GB，使用POLARDB节点规格为88核 710G的配置在主节点进行测试。

- [GROUP BY & ORDER BY支持](#)
- [AGGREGATE函数支持 \(SUM/^{AVG}/COUNT\)](#)

- [JOIN支持](#)
- [BETWEEN函数 & IN函数支持](#)
- [LIMIT支持](#)
- [INTERVAL函数支持](#)
- [CASE WHEN支持](#)
- [LIKE支持](#)

GROUP BY & ORDER BY支持

未开启并行查询，耗时1563.32秒，开启并行查询后，耗时只用49.65秒，提升31.48倍。

原始SQL语句，如下所示：

```
SELECT    l_returnflag,
          l_linestatus,
          Sum(l_quantity)                                AS
sum_qty,
          Sum(l_extendedprice)                          AS
sum_base_price,
          Sum(l_extendedprice * (1 - l_discount))        AS
sum_disc_price,
          Sum(l_extendedprice * (1 - l_discount) * (1 + l_tax)) AS
sum_charge,
          Avg(l_quantity)                                AS
avg_qty,
          Avg(l_extendedprice)                          AS
avg_price,
          Avg(l_discount)                                AS
avg_disc,
          Count(*)                                       AS
count_order
FROM      lineitem
WHERE     l_shipdate <= date '1998-12-01' - INTERVAL '93' day
GROUP BY l_returnflag,
          l_linestatus
ORDER BY l_returnflag,
          l_linestatus ;
```

未开启并行查询，耗时1563.32秒。

```
mysql> SELECT /*+ SET_VAR(max_parallel_degree=0) */ l_returnflag,
->          l_linestatus,
->          Sum(l_quantity)                                AS sum_qty,
->          Sum(l_extendedprice)                          AS sum_base_price,
->          Sum(l_extendedprice * (1 - l_discount))        AS sum_disc_price,
->          Sum(l_extendedprice * (1 - l_discount) * (1 + l_tax)) AS sum_charge,
->          Avg(l_quantity)                                AS avg_qty,
->          Avg(l_extendedprice)                          AS avg_price,
->          Avg(l_discount)                                AS avg_disc,
->          Count(*)                                       AS count_order
-> FROM      lineitem
-> WHERE     l_shipdate <= date '1998-12-01' - INTERVAL '93' day
-> GROUP BY l_returnflag,
->          l_linestatus
-> ORDER BY l_returnflag,
->          l_linestatus ;
^@+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| l_returnflag | l_linestatus | sum_qty | sum_base_price | sum_disc_price | sum_charge | avg_qty | avg_price | avg_disc | count_order |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| A           | F           | 3775127758.00 | 5660776097194.45 | 5377736398183.9374 | 5592847429515.927026 | 25.499370 | 38236.116984 | 0.050002 | 148047881 |
| N           | F           | 98553062.00 | 147771098385.98 | 140384965965.0348 | 145999793032.775829 | 25.501557 | 38237.199389 | 0.049985 | 3864590 |
| N           | O           | 7421794698.00 | 11128967412861.24 | 10572526740167.5066 | 10995437028892.436612 | 25.500030 | 38237.248190 | 0.049998 | 291050427 |
| R           | F           | 3775724970.00 | 5661603032745.34 | 5378513563915.4097 | 5593662252666.916161 | 25.500066 | 38236.697258 | 0.050001 | 148067261 |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
4 rows in set (26 min 3.32 sec)
```

开启并行查询后，耗时只用49.65秒，提升31.48 倍。

```
mysql> SELECT /*+ SET_VAR(max_parallel_degree=32) */ l_returnflag,
-> l_linestatus,
-> Sum(l_quantity) AS sum_qty,
-> Sum(l_extendedprice) AS sum_base_price,
-> Sum(l_extendedprice * (1 - l_discount)) AS sum_disc_price,
-> Sum(l_extendedprice * (1 - l_discount) * (1 + l_tax)) AS sum_charge,
-> Avg(l_quantity) AS avg_qty,
-> Avg(l_extendedprice) AS avg_price,
-> Avg(l_discount) AS avg_disc,
-> Count(*) AS count_order
-> FROM lineitem
-> WHERE l_shipdate <= date '1998-12-01' - INTERVAL '93' day
-> GROUP BY l_returnflag,
-> l_linestatus
-> ORDER BY l_returnflag,
-> l_linestatus ;
```

l_returnflag	l_linestatus	sum_qty	sum_base_price	sum_disc_price	sum_charge	avg_qty	avg_price	avg_disc	count_order
A	F	3775127758.00	5660776097194.45	5377736398183.9374	5592847429515.927026	25.499370	38236.116984	0.050002	148047881
N	F	98553062.00	147771098385.98	140384965965.0348	145999793032.775829	25.501557	38237.199389	0.049985	3864590
N	O	7421794698.00	11128967412861.24	10572526740167.5066	10995437028892.436612	25.500030	38237.248190	0.049998	291050427
R	F	3775724970.00	5661603032745.34	5378513563915.4097	5593662252666.916161	25.500066	38236.697258	0.050001	148067261

4 rows in set (49.65 sec)

AGGREGATE函数支持 (SUM/AVG/COUNT)

未开启并行查询，耗时1563.32秒，开启并行查询后，耗时只用49.65秒，提升31.48 倍。

原始SQL语句，如下所示：

```
SELECT l_returnflag,
       l_linestatus,
       Sum(l_quantity) AS
sum_qty,
       Sum(l_extendedprice) AS
sum_base_price,
       Sum(l_extendedprice * (1 - l_discount)) AS
sum_disc_price,
       Sum(l_extendedprice * (1 - l_discount) * (1 + l_tax)) AS
sum_charge,
       Avg(l_quantity) AS
avg_qty,
       Avg(l_extendedprice) AS
avg_price,
       Avg(l_discount) AS
avg_disc,
       Count(*) AS
count_order
FROM lineitem
WHERE l_shipdate <= date '1998-12-01' - INTERVAL '93' day
GROUP BY l_returnflag,
         l_linestatus
ORDER BY l_returnflag,
         l_linestatus ;
```

未开启并行查询，耗时1563.32秒。

```
mysql> SELECT /*+ SET_VAR(max_parallel_degree=0) */ l_returnflag,
-> l_linestatus,
-> Sum(l_quantity) AS sum_qty,
-> Sum(l_extendedprice) AS sum_base_price,
-> Sum(l_extendedprice * (1 - l_discount)) AS sum_disc_price,
-> Sum(l_extendedprice * (1 - l_discount) * (1 + l_tax)) AS sum_charge,
-> Avg(l_quantity) AS avg_qty,
-> Avg(l_extendedprice) AS avg_price,
-> Avg(l_discount) AS avg_disc,
-> Count(*) AS count_order
-> FROM lineitem
-> WHERE l_shipdate <= date '1998-12-01' - INTERVAL '93' day
-> GROUP BY l_returnflag,
-> l_linestatus
-> ORDER BY l_returnflag,
-> l_linestatus ;
```

l_returnflag	l_linestatus	sum_qty	sum_base_price	sum_disc_price	sum_charge	avg_qty	avg_price	avg_disc	count_order
A	F	3775127758.00	5660776097194.45	5377736398183.9374	5592847429515.927026	25.499370	38236.116984	0.050002	148047881
N	F	98553062.00	147771098385.98	140384965965.0348	145999793032.775829	25.501557	38237.199389	0.049985	3864590
N	O	7421794698.00	11128967412861.24	10572526740167.5066	10995437028892.436612	25.500030	38237.248190	0.049998	291050427
R	F	3775724970.00	5661603032745.34	5378513563915.4097	5593662252666.916161	25.500066	38236.697258	0.050001	148067261

4 rows in set (26 min 3.32 sec)

开启并行查询后，耗时只用49.65秒，提升31.48倍。

```
mysql> SELECT /*+ SET_VAR(max_parallel_degree=32) */ l_returnflag,
-> l_linestatus,
-> Sum(l_quantity) AS sum_qty,
-> Sum(l_extendedprice) AS sum_base_price,
-> Sum(l_extendedprice * (1 - l_discount)) AS sum_disc_price,
-> Sum(l_extendedprice * (1 - l_discount) * (1 + l_tax)) AS sum_charge,
-> Avg(l_quantity) AS avg_qty,
-> Avg(l_extendedprice) AS avg_price,
-> Avg(l_discount) AS avg_disc,
-> Count(*) AS count_order
-> FROM lineitem
-> WHERE l_shipdate <= date '1998-12-01' - INTERVAL '93' day
-> GROUP BY l_returnflag,
-> l_linestatus
-> ORDER BY l_returnflag,
-> l_linestatus ;
```

l_returnflag	l_linestatus	sum_qty	sum_base_price	sum_disc_price	sum_charge	avg_qty	avg_price	avg_disc	count_order
A	F	3775127758.00	5660776097194.45	5377736398183.9374	5592847429515.927026	25.499370	38236.116984	0.050002	148047881
N	F	98553062.00	147771098385.98	140384965965.0348	145999793032.775829	25.501557	38237.199389	0.049985	3864590
N	O	7421794698.00	11128967412861.24	10572526740167.5066	10995437028892.436612	25.500030	38237.248190	0.049998	291050427
R	F	3775724970.00	5661603032745.34	5378513563915.4097	5593662252666.916161	25.500066	38236.697258	0.050001	148067261

4 rows in set (49.65 sec)

JOIN支持

未开启并行查询，耗时21.73秒，开启并行查询后，耗时1.37秒，提升15.86倍。

原始SQL语句，如下所示：

```
select sum(l_extendedprice* (1 - l_discount)) as revenue
from lineitem, part
where ( p_partkey = l_partkey and p_brand = 'Brand#12'
and p_container in ('SM CASE', 'SM BOX', 'SM PACK', 'SM PKG')
and l_quantity >= 6 and l_quantity <= 6 + 10
and p_size between 1 and 5
and l_shipmode in ('AIR', 'AIR REG')
and l_shipinstruct = 'DELIVER IN PERSON' )
or ( p_partkey = l_partkey and p_brand = 'Brand#13'
and p_container in ('MED BAG', 'MED BOX', 'MED PKG', 'MED PACK'
')
and l_quantity >= 10 and l_quantity <= 10 + 10
and p_size between 1 and 10
and l_shipmode in ('AIR', 'AIR REG')
and l_shipinstruct = 'DELIVER IN PERSON' )
or ( p_partkey = l_partkey and p_brand = 'Brand#24'
and p_container in ('LG CASE', 'LG BOX', 'LG PACK', 'LG PKG')
and l_quantity >= 21 and l_quantity <= 21 + 10
and p_size between 1 and 15
and l_shipmode in ('AIR', 'AIR REG'))
```

```
and l_shipinstruct = 'DELIVER IN PERSON' );
```

未开启并行查询，耗时21.73秒。

```
mysql> select /*+ SET_VAR(max_parallel_degree=0) */ sum(l_extendedprice* (1 - l_discount)) as revenue
-> from lineitem, part
-> where ( p_partkey = l_partkey and p_brand = 'Brand#12'
->         and p_container in ('SM CASE', 'SM BOX', 'SM PACK', 'SM PKG')
->         and l_quantity >= 6 and l_quantity <= 6 + 10
->         and p_size between 1 and 5
->         and l_shipmode in ('AIR', 'AIR REG')
->         and l_shipinstruct = 'DELIVER IN PERSON' )
-> or ( p_partkey = l_partkey and p_brand = 'Brand#13'
->         and p_container in ('MED BAG', 'MED BOX', 'MED PKG', 'MED PACK')
->         and l_quantity >= 10 and l_quantity <= 10 + 10
->         and p_size between 1 and 10
->         and l_shipmode in ('AIR', 'AIR REG')
->         and l_shipinstruct = 'DELIVER IN PERSON' )
-> or ( p_partkey = l_partkey and p_brand = 'Brand#24'
->         and p_container in ('LG CASE', 'LG BOX', 'LG PACK', 'LG PKG')
->         and l_quantity >= 21 and l_quantity <= 21 + 10
->         and p_size between 1 and 15
->         and l_shipmode in ('AIR', 'AIR REG')
->         and l_shipinstruct = 'DELIVER IN PERSON' );

+-----+
| revenue |
+-----+
| 317693789.3851 |
+-----+
1 row in set (21.73 sec)
```

开启并行查询后，耗时1.37秒，提升15.86倍。

```
mysql> select /*+ SET_VAR(max_parallel_degree=32) */ sum(l_extendedprice* (1 - l_discount)) as revenue
-> from lineitem, part
-> where ( p_partkey = l_partkey and p_brand = 'Brand#12'
->         and p_container in ('SM CASE', 'SM BOX', 'SM PACK', 'SM PKG')
->         and l_quantity >= 6 and l_quantity <= 6 + 10
->         and p_size between 1 and 5
->         and l_shipmode in ('AIR', 'AIR REG')
->         and l_shipinstruct = 'DELIVER IN PERSON' )
-> or ( p_partkey = l_partkey and p_brand = 'Brand#13'
->         and p_container in ('MED BAG', 'MED BOX', 'MED PKG', 'MED PACK')
->         and l_quantity >= 10 and l_quantity <= 10 + 10
->         and p_size between 1 and 10
->         and l_shipmode in ('AIR', 'AIR REG')
->         and l_shipinstruct = 'DELIVER IN PERSON' )
-> or ( p_partkey = l_partkey and p_brand = 'Brand#24'
->         and p_container in ('LG CASE', 'LG BOX', 'LG PACK', 'LG PKG')
->         and l_quantity >= 21 and l_quantity <= 21 + 10
->         and p_size between 1 and 15
->         and l_shipmode in ('AIR', 'AIR REG')
->         and l_shipinstruct = 'DELIVER IN PERSON' );

+-----+
| revenue |
+-----+
| 317693789.3851 |
+-----+
1 row in set (1.37 sec)
```

BETWEEN函数 & IN函数支持

未开启并行查询，耗时21.73秒，开启并行查询后，耗时1.37秒，提升15.86倍。

原始SQL语句，如下所示：

```
select sum(l_extendedprice* (1 - l_discount)) as revenue
from   lineitem,   part
where  ( p_partkey = l_partkey and p_brand = 'Brand#12'
        and p_container in ('SM CASE', 'SM BOX', 'SM PACK', 'SM PKG')
        and l_quantity >= 6 and l_quantity <= 6 + 10
        and p_size between 1 and 5
        and l_shipmode in ('AIR', 'AIR REG')
        and l_shipinstruct = 'DELIVER IN PERSON' )
      or ( p_partkey = l_partkey and p_brand = 'Brand#13'
        and p_container in ('MED BAG', 'MED BOX', 'MED PKG', 'MED PACK
        ')
        and l_quantity >= 10 and l_quantity <= 10 + 10
        and p_size between 1 and 10
        and l_shipmode in ('AIR', 'AIR REG')
        and l_shipinstruct = 'DELIVER IN PERSON' )
      or ( p_partkey = l_partkey and p_brand = 'Brand#24'
        and p_container in ('LG CASE', 'LG BOX', 'LG PACK', 'LG PKG')
        and l_quantity >= 21 and l_quantity <= 21 + 10
        and p_size between 1 and 15
        and l_shipmode in ('AIR', 'AIR REG')
        and l_shipinstruct = 'DELIVER IN PERSON' );
```

未开启并行查询，耗时21.73秒。

```
mysql> select /*+ SET_VAR(max_parallel_degree=0) */ sum(l_extendedprice* (1 - l_discount)) as revenue
-> from   lineitem,   part
-> where  ( p_partkey = l_partkey and p_brand = 'Brand#12'
->          and p_container in ('SM CASE', 'SM BOX', 'SM PACK', 'SM PKG')
->          and l_quantity >= 6 and l_quantity <= 6 + 10
->          and p_size between 1 and 5
->          and l_shipmode in ('AIR', 'AIR REG')
->          and l_shipinstruct = 'DELIVER IN PERSON' )
->      or ( p_partkey = l_partkey and p_brand = 'Brand#13'
->          and p_container in ('MED BAG', 'MED BOX', 'MED PKG', 'MED PACK')
->          and l_quantity >= 10 and l_quantity <= 10 + 10
->          and p_size between 1 and 10
->          and l_shipmode in ('AIR', 'AIR REG')
->          and l_shipinstruct = 'DELIVER IN PERSON' )
->      or ( p_partkey = l_partkey and p_brand = 'Brand#24'
->          and p_container in ('LG CASE', 'LG BOX', 'LG PACK', 'LG PKG')
->          and l_quantity >= 21 and l_quantity <= 21 + 10
->          and p_size between 1 and 15
->          and l_shipmode in ('AIR', 'AIR REG')
->          and l_shipinstruct = 'DELIVER IN PERSON' );

+-----+
| revenue |
+-----+
| 317693789.3851 |
+-----+
1 row in set (21.73 sec)
```

开启并行查询后，耗时1.37秒，提升15.86倍。

```
mysql> select /*+ SET_VAR(max_parallel_degree=32) */ sum(l_extendedprice* (1 - l_discount)) as revenue
-> from   lineitem, part
-> where ( p_partkey = l_partkey and p_brand = 'Brand#12'
->         and p_container in ('SM CASE', 'SM BOX', 'SM PACK', 'SM PKG')
->         and l_quantity >= 6 and l_quantity <= 6 + 10
->         and p_size between 1 and 5
->         and l_shipmode in ('AIR', 'AIR REG')
->         and l_shipinstruct = 'DELIVER IN PERSON' )
-> or ( p_partkey = l_partkey and p_brand = 'Brand#13'
->         and p_container in ('MED BAG', 'MED BOX', 'MED PKG', 'MED PACK')
->         and l_quantity >= 10 and l_quantity <= 10 + 10
->         and p_size between 1 and 10
->         and l_shipmode in ('AIR', 'AIR REG')
->         and l_shipinstruct = 'DELIVER IN PERSON' )
-> or ( p_partkey = l_partkey and p_brand = 'Brand#24'
->         and p_container in ('LG CASE', 'LG BOX', 'LG PACK', 'LG PKG')
->         and l_quantity >= 21 and l_quantity <= 21 + 10
->         and p_size between 1 and 15
->         and l_shipmode in ('AIR', 'AIR REG')
->         and l_shipinstruct = 'DELIVER IN PERSON' );
+-----+
| revenue |
+-----+
| 317693789.3851 |
+-----+
1 row in set (1.37 sec)
```

LIMIT支持

未开启并行查询，耗时339.22秒，开启并行查询后，耗时29.31秒，提升11.57倍。

原始SQL语句，如下所示：

```
select l_shipmode, sum(case when o_orderpriority = '1-URGENT' or
o_orderpriority = '2-HIGH' then 1
else 0
end) as high_line_count, sum(case when o_orderpriority <> '1-URGENT'
and o_orderpriority <> '2-HIGH' then 1
else 0
end) as low_line_count
from   orders, lineitem
where o_orderkey = l_orderkey
and l_shipmode in ('MAIL', 'TRUCK')
and l_commitdate < l_receiptdate
and l_shipdate < l_commitdate
and l_receiptdate >= date '1996-01-01'
and l_receiptdate < date '1996-01-01' + interval '1' year
group by l_shipmode
order by l_shipmode limit 10;
```

未开启并行查询，耗时339.22秒。

```
mysql> select /** SET_VAR(max_parallel_degree=0) */
  l_shipmode, sum(case when o_orderpriority = '1-URGENT' or o_orderpriority = '2-HIGH' then 1
    else 0
  end) as high_line_count, sum(case when o_orderpriority <> '1-URGENT' and o_orderpriority <> '2-HIGH' then 1
    else 0
  end) as low_line_count
from orders, lineitem
where o_orderkey = l_orderkey
and l_shipmode in ('MAIL', 'TRUCK')
and l_commitdate < l_receiptdate
and l_shipdate < l_commitdate
and l_receiptdate >= date '1996-01-01'
and l_receiptdate < date '1996-01-01' + interval '1' year
group by l_shipmode
order by l_shipmode limit 10;
```

l_shipmode	high_line_count	low_line_count
MAIL	625339	937117
TRUCK	625580	937993

2 rows in set (339.22 sec)

开启并行查询后，耗时29.31秒，提升11.57倍。

```
mysql> select /** SET_VAR(max_parallel_degree=32) */
  l_shipmode, sum(case when o_orderpriority = '1-URGENT' or o_orderpriority = '2-HIGH' then 1
    else 0
  end) as high_line_count, sum(case when o_orderpriority <> '1-URGENT' and o_orderpriority <> '2-HIGH' then 1
    else 0
  end) as low_line_count
from orders, lineitem
where o_orderkey = l_orderkey
and l_shipmode in ('MAIL', 'TRUCK')
and l_commitdate < l_receiptdate
and l_shipdate < l_commitdate
and l_receiptdate >= date '1996-01-01'
and l_receiptdate < date '1996-01-01' + interval '1' year
group by l_shipmode
order by l_shipmode limit 10;
```

l_shipmode	high_line_count	low_line_count
MAIL	625339	937117
TRUCK	625580	937993

2 rows in set (29.31 sec)

INTERVAL函数支持

未开启并行查询，耗时220.87秒，开启并行查询后，耗时7.75秒，提升28.5倍。

原始SQL语句，如下所示：

```
select
  100.00 * sum(case when p_type like 'PROMO%' then l_extendedprice *
    (1 - l_discount)
    else 0
  end) / sum(l_extendedprice * (1 - l_discount)) as promo_revenue
from lineitem, part
where l_partkey = p_partkey
and l_shipdate >= date '1996-01-01'
and l_shipdate < date '1996-01-01' + interval '1' month limit 10;
```

未开启并行查询，耗时220.87秒。

```
mysql> select /*+ SET_VAR(max_parallel_degree=0) */
-> 100.00 * sum(case when p_type like 'PROMO%' then l_extendedprice * (1 - l_discount)
-> else 0
-> end) / sum(l_extendedprice * (1 - l_discount)) as promo_revenue
-> from lineitem, part
-> where l_partkey = p_partkey
-> and l_shipdate >= date '1996-01-01'
-> and l_shipdate < date '1996-01-01' + interval '1' month limit 10;

^@+-----+
| promo_revenue |
+-----+
| 16.6415897388 |
+-----+
1 row in set (3 min 40.87 sec)
```

开启并行查询后，耗时7.75秒，提升28.5倍。

```
mysql>
mysql> select /*+ SET_VAR(max_parallel_degree=32) */
-> 100.00 * sum(case when p_type like 'PROMO%' then l_extendedprice * (1 - l_discount)
-> else 0
-> end) / sum(l_extendedprice * (1 - l_discount)) as promo_revenue
-> from lineitem, part
-> where l_partkey = p_partkey
-> and l_shipdate >= date '1996-01-01'
-> and l_shipdate < date '1996-01-01' + interval '1' month limit 10;

+-----+
| promo_revenue |
+-----+
| 16.6415897388 |
+-----+
1 row in set (7.75 sec)
```

CASE WHEN支持

未开启并行查询，耗时220.87秒，开启并行查询后，耗时7.75秒，提升28.5倍。

原始SQL语句，如下所示：

```
select
  100.00 * sum(case when p_type like 'PROMO%' then l_extendedprice *
    (1 - l_discount)
    else 0
  end) / sum(l_extendedprice * (1 - l_discount)) as promo_revenue
from lineitem, part
where l_partkey = p_partkey
and l_shipdate >= date '1996-01-01'
and l_shipdate < date '1996-01-01' + interval '1' month limit 10;
```

未开启并行查询，耗时220.87秒。

```
mysql> select /*+ SET_VAR(max_parallel_degree=0) */
-> 100.00 * sum(case when p_type like 'PROMO%' then l_extendedprice * (1 - l_discount)
-> else 0
-> end) / sum(l_extendedprice * (1 - l_discount)) as promo_revenue
-> from lineitem, part
-> where l_partkey = p_partkey
-> and l_shipdate >= date '1996-01-01'
-> and l_shipdate < date '1996-01-01' + interval '1' month limit 10;

^@+-----+
| promo_revenue |
+-----+
| 16.6415897388 |
+-----+
1 row in set (3 min 40.87 sec)
```

开启并行查询后，耗时7.75秒，提升28.5倍。

```
mysql>
mysql> select /*+ SET_VAR(max_parallel_degree=32) */
-> 100.00 * sum(case when p_type like 'PROMO%' then l_extendedprice * (1 - l_discount)
-> else 0
-> end) / sum(l_extendedprice * (1 - l_discount)) as promo_revenue
-> from lineitem, part
-> where l_partkey = p_partkey
-> and l_shipdate >= date '1996-01-01'
-> and l_shipdate < date '1996-01-01' + interval '1' month limit 10;

+-----+
| promo_revenue |
+-----+
| 16.6415897388 |
+-----+
1 row in set (7.75 sec)
```

LIKE支持

未开启并行查询，耗时427.46秒，开启并行查询后，耗时33.72秒，提升12.68倍。

原始SQL语句，如下所示：

```
select s_name, s_address from
  supplier, nation where
s_suppkey in
  ( select ps_suppkey from partsupp where
    ps_partkey in ( select p_partkey from part where p_name
like 'dark%')
    and ps_availqty>(select 0.0005 * sum(l_quantity) as col1
from lineitem, partsupp
where l_partkey = ps_partkey and l_suppkey = ps_suppkey
and l_shipdate >= date '1993-01-01' and l_shipdate < date '1993-
01-01' + interval '1' year)
)
and s_nationkey = n_nationkey and n_name = 'JORDAN'
order by s_name limit 10;
```

未开启并行查询，耗时427.46秒。

```
mysql>
mysql> select /*+ SET_VAR(max_parallel_degree=0) */ s_name, s_address from
-> supplier, nation where
-> s_suppkey in
-> ( select ps_suppkey from partsupp where
->      ps_partkey in ( select p_partkey from part where p_name like 'dark%')
->      and ps_availqty>(select 0.0005 * sum(l_quantity) as col1
->      from lineitem, partsupp
->      where l_partkey = ps_partkey and l_suppkey = ps_suppkey
->      and l_shipdate >= date '1993-01-01' and l_shipdate < date '1993-01-01' + interval '1' year)
-> )
-> and s_nationkey = n_nationkey and n_name = 'JORDAN'
-> order by s_name limit 10;
ct p_partkey from part where p_name like 'dark%')
      and ps_availqty>(select 0.0005 * sum(l_quantity) as col1
      from lineitem, partsupp
      where l_partkey = ps_partkey and l_suppkey = ps_suppkey
      and l_shipdate >= date '1993-01-01' and l_shipdate < date '1993-01-01' + interval '1' year)
      )
and s_nationkey = n_nationkey and n_name = 'JORDAN'
order by s_name limit 10;
^@^@^@^@^@Empty set (7 min 7.46 sec)
```

开启并行查询后，耗时33.72秒，提升12.68倍。

```
mysql> select /*+ SET_VAR(max_parallel_degree=32) */ s_name, s_address from
-> supplier, nation where
-> s_suppkey in
-> ( select ps_suppkey from partsupp where
->      ps_partkey in ( select p_partkey from part where p_name like 'dark%')
->      and ps_availqty>(select 0.0005 * sum(l_quantity) as col1
->      from lineitem, partsupp
->      where l_partkey = ps_partkey and l_suppkey = ps_suppkey
->      and l_shipdate >= date '1993-01-01' and l_shipdate < date '1993-01-01' + interval '1' year)
-> )
-> and s_nationkey = n_nationkey and n_name = 'JORDAN'
-> order by s_name limit 10;
^@Empty set (33.72 sec)
```

17.3 并行查询使用限制

本文为您介绍并行查询的使用限制，帮助您正确使用并行查询功能。

并行查询的使用限制

目前，以下情况暂时无法享受并行查询带来的性能提升：

- 表记录小于20000条
- 系统表或临时表
- 查询语句SELECT...FOR UPDATE/SHARE
- 使用全文索引的查询
- 存储过程Procedures
- 自定义函数UDF
- Recursive CTE
- Window functions

- GIS
- XML函数
- GROUP BY WITH ROLLUP
- 带锁的相关函数
- 非Btree 的INDEX
- INDEX MERGE
- 串行化隔离级别事务内的查询语句
- 当前并行的工作线程数已经等于或超过4倍CPU核数

18 如何开启Binlog

本文介绍如何为POLARDB集群开启Binlog功能。

前提条件

集群创建于2019年4月5日之后。如果是在2019年4月5日之前创建的集群，暂时需要[提交工单](#)进行小版本升级，之后即可在控制台手动开启Binlog。

背景信息

POLARDB是一款完全兼容MySQL的云原生数据库，默认使用了更高级别的物理日志代替Binlog，但为了更好地与MySQL生态融合，POLARDB上线了开启Binlog的功能，开启Binlog后您就可以连接[ElasticSearch](#)、[AnalyticDB](#)等数据产品，也可以搭建[POLARDB到RDS](#)、[RDS到POLARDB](#)或POLARDB之间的数据实时同步。

收费说明

Binlog的空间属于集群存储空间的一部分，需要收取[存储费用](#)。

注意事项

- 开启后Binlog默认保存2周，超出两周的Binlog文件会被自动删除。您可以修改参数`loose_expire_logs_hours`（取值范围0~2376，单位：小时）以设置Binlog的保存时长。0表示不自动删除Binlog文件。
- Binlog功能默认关闭，开启Binlog需要重启实例，会造成连接中断，重启前请做好业务安排，谨慎操作。
- 开启Binlog后会导致写性能下降，读性能不受影响。

- 拉取、订阅或同步Binlog（例如使用DTS等工具）时，建议使用POLARDB的主地址，因为直接指向生成Binlog的主节点，具有更好的兼容性和稳定性。您可以在基本信息页面查看主地址，如下图所示。



开启Binlog的操作步骤

1. 登录POLARDB控制台。
2. 选择地域。
3. 找到目标集群，单击集群名称列的集群ID。
4. 在左侧导航栏中选择配置与管理 > 参数配置。
5. 搜索loose_polar_log_bin，修改当前值，并单击提交修改。



说明:

- 数据库引擎为MySQL 5.6的POLARDB集群，当前值修改为ON_WITH_GTID；
- 数据库引擎为MySQL 8.0的POLARDB集群，当前值修改为ON。

6. 在右侧提示重启的对话框中单击确定。



说明:

如果报错提示Custins minor version does not support current action, 请提交工单开启。

常见问题

- Binlog能保存多久？

答：Binlog默认保存2周，超出两周的Binlog文件会被自动删除。您可以修改参数`loose_expire_logs_hours`（取值范围0~2376，单位：小时）以设置Binlog的保存时长。0表示不自动删除Binlog文件。

- 开启Binlog后可以关闭吗？

答：将参数`loose_polar_log_bin`修改为OFF并提交即可关闭。关闭后已有的Binlog不会被删除。

- 开启Binlog对性能影响有多大？

答：测试数据显示，在64线程并发下，开启Binlog后会有30%~40%的写性能衰减（性能衰减随并发增加而减少，后续将会持续优化），读性能不受影响。对于读多写少的业务场景，整体数据库性能影响较小，例如读写比是4:1的数据库，整体性能影响大约是10%。

19 POLARDB for MySQL常见问题

删除数据库后为什么还是占用很多空间？

答：这是由于redolog日志文件占用了空间，通常在2G~11G左右，最多时会占用11G：8G（缓冲池中的8个redolog日志）+1G（正在写的redolog日志）+1G（提前创建的redolog日志）+1G（最后一个redolog日志）。

缓冲池内的redolog日志文件数量由参数`loose_innodb_polar_log_file_max_reuse`控制，默认是8，您可以修改这个参数从而减少日志空间占用量，但是在压力大的情况下，性能可能会出现周期性的小幅波动。



说明：

调整参数`loose_innodb_polar_log_file_max_reuse`后，缓冲池不会立刻被清空，随着DML被执行，才会慢慢减少。如果需要立即清空，请联系售后服务。

参数配置	log_throttle_queries_not_using_indexes ⓘ	0	否	0	[0-4294967295]
诊断与优化	long_query_time ⓘ	1	否	1	[0.03-10]
集群总览	loose_expire_logs_hours ⓘ	336	否	336	[0-2376]
性能监控	loose_innodb_polar_log_file_max_reuse ⓘ	8	否	8	[0-100]
问题分析	loose_innodb_primary_abort_ddl_wait_replica_timeout ⓘ	3600	否	3600	[1-31536000]
慢 SQL					

磁盘空间无法选择怎么办？

答：存储空间无需手动选择，系统根据数据量自动伸缩。

POLARDB底层使用存储集群的方式，可以做到磁盘动态扩容，且磁盘扩容过程对用户无感知，当磁盘空间使用了70%，系统就会自动扩容一部分空间，而且扩容不需要停止实例。通过这种机制，POLARDB的存储可以做到按照使用量来收费。

读写分离怎么保证读一致性？

答：读写分离链路会记录日志序号（Log sequence number, LSN），读请求会发往LSN符合要求的只读节点，详情请参见[读写分离](#)。

如何实现POLARDB的读写分离？

答：只需在应用程序中使用集群地址，即可根据配置的读负载节点实现读写分离。您也可以[自定义集群地址](#)。

访问信息 ②

▼ 白名单 ②

白名单列表  [配置](#)

▼ 主地址 ②

私网  [修改](#)

公网 [申请](#)

▼ 集群地址【推荐】 ② [新增自定义地址](#)

• 默认地址 () [修改配置](#)

读写模式 [可读可写 \(自动读写分离\)](#)

私网  [修改](#)

公网 [申请](#)

> 节点配置

> 高级配置

如果有多个只读节点，如何设置指定的ECS访问指定的只读节点？

答：您可以设置[自定义集群地址](#)，自行选择需要接入的只读节点，然后在ECS上使用该自定义集群地址。

只用了主地址，但是发现只读节点也有负载，是否主地址也支持读写分离？

答：主地址不支持读写分离，始终只连接到主节点。只读节点有少量QPS是正常现象，与主地址无关。

如何发现慢SQL

答：`#unique_131`后执行`show processlist;`找出执行时间过长的SQL。

```
mysql> show processlist;
+-----+-----+-----+-----+-----+-----+-----+-----+
| Id    | User | Host                                | db    | Command | Time | State | Info                                |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 33554490 | acc  | 192.168.1.10:10358 | NULL | Query   | 0    | starting | show processlist |
| 33554499 | acc  | 192.168.1.10:10358 | NULL | Query   | 250  | User sleep | select sleep(600) |
| 33554499 | acc  | 192.168.1.10:10358 | NULL | Sleep   | 253  | | NULL |
+-----+-----+-----+-----+-----+-----+-----+-----+
3 rows in set, 13312 warnings (0.00 sec)
```

如何终止慢SQL

答：发现慢SQL后，查看慢SQL的Id，执行`kill <Id>`终止慢SQL。

```
mysql> kill 33554499;  
Query OK, 0 rows affected (0.01 sec)
```

20 隐藏目录

20.1 查看日志

系统自动记录实例的错误日志和慢查询日志（以下称为慢日志）。您可以查看错误日志列表、慢日志明细，和慢日志统计信息，帮助故障定位分析。

查看日志

1. 进入[POLARDB控制台](#)。
2. 选择目标实例所在的地域。
3. 在左侧导航栏中选择实例列表。
4. 单击目标实例的ID，或者在操作列中选择管理。
5. 在左侧导航栏中选择日志管理。
6. 选择错误日志、慢日志明细或者慢日志统计，选择时间范围，然后单击刷新。

查询项	内容
错误日志	记录1个月内数据库中执行出错的SQL语句。错误日志每一分钟自动刷新一次。
慢日志明细	· 记录1个月内数据库中执行时间超过1秒（可以在参数设置中修改long_query_time参数来设置）的SQL语句，并进行相似语句去重。 · 记录最多1万条慢日志。超出1万条时，将删除最早的慢日志。 · 如果一条慢SQL语句长度超过2048个字符，相应的慢日志仅记录其前2048个字符。
慢日志统计	对1个月内数据库中执行时间超过1秒（可以在参数设置中修改long_query_time参数来设置）的SQL语句进行统计汇总，给出慢查询日志的分析报告。