

目录

| | |
|------------------|---|
| 目录 | 1 |
| 版本 | 5 |
| 云数据库MySQL版本支持 | 5 |
| 应用 | 5 |
| 云数据库MySQL应用案例 | 5 |
| 概览 | 5 |
| 操作总览 | 5 |
| 实例列表页 | 5 |
| 更配 | 5 |
| 重启 | 5 |
| 修改 | 5 |
| 删除 | 5 |
| 重置密码 | 5 |
| 创建只读实例 | 5 |
| 分配至项目 | 6 |
| 实例详情页 | 6 |
| 登录数据库 | 6 |
| 主备切换 | 6 |
| 迁移可用区 | 6 |
| 更改网络 | 6 |
| 更改VPC | 6 |
| 更改IP地址 | 6 |
| 更改端口 | 6 |
| 存储空间自动扩容 | 6 |
| 前提条件 | 6 |
| 配置存储空间自动扩容 | 7 |
| 存储空间自动扩容规则 | 7 |
| 数据库代理 | 7 |
| 概述 | 7 |
| 功能特性 | 7 |
| 适用场景 | 7 |
| 开通代理实例 | 7 |
| 自动读写分离 | 7 |
| 连接方式 | 8 |
| 监控与日志 | 8 |
| 关闭数据代理 | 8 |
| 版本升级 | 8 |
| 主版本升级 | 8 |
| 升级前注意事项: | 8 |
| 主版本升级步骤 | 8 |
| 小版本升级步骤 | 8 |
| 只读实例 | 8 |
| 前提条件 | 8 |
| 操作步骤 | 9 |
| 查看只读实例 | 9 |
| 在实例列表中查看只读实例 | 9 |
| 在主实例的实例详情页查看只读实例 | 9 |
| 账号管理 | 9 |
| 新建账号 | 9 |
| 修改权限 | 9 |

| | |
|---------------|----|
| 克隆账号 | 9 |
| 删除 | 9 |
| 数据库管理 | 10 |
| 新建数据库 | 10 |
| 修改权限 | 10 |
| 删除 | 10 |
| 实例监控 | 10 |
| 监控查看 | 10 |
| 修改监控粒度 | 10 |
| 操作步骤 | 10 |
| 配置告警 | 11 |
| 参数配置 | 11 |
| 新建参数组 | 11 |
| 应用参数组 | 11 |
| 编辑参数组 | 11 |
| 删除参数组 | 11 |
| 实例参数配置 | 11 |
| 安全组 | 11 |
| 新建安全组 | 11 |
| 删除安全组 | 11 |
| 克隆安全组 | 12 |
| 管理安全组 | 12 |
| 安全组添加/删除IP | 12 |
| 安全组添加/解绑实例 | 12 |
| 设置安全组 | 12 |
| 解绑安全组 | 13 |
| 导出 | 13 |
| 备份与恢复 | 13 |
| 手动备份 | 13 |
| 恢复数据 | 13 |
| 基于时间点恢复和库表级恢复 | 13 |
| 删除备份 | 14 |
| 自动备份设置 | 14 |
| 其他 | 14 |
| 日志管理 | 14 |
| 临时实例 | 14 |
| 外网访问 | 14 |
| 标签使用 | 15 |
| 编辑标签 | 15 |
| 标签显示及筛选 | 15 |
| 数据库审计 | 15 |
| 开启数据库审计 | 15 |
| 关闭数据库审计 | 15 |
| 审计日志查询 | 15 |
| 导出审计结果 | 16 |
| 查看导出历史 | 16 |
| 审计监控图表 | 16 |
| 审计日志分析 | 16 |
| 性能分析 | 16 |
| 慢SQL | 16 |
| 慢SQL统计 | 16 |
| 慢SQL详情 | 16 |

| | |
|---------------------------------|----|
| 热点表分析 | 17 |
| 测试环境 | 17 |
| 测试工具 | 17 |
| SysBench工具介绍 | 17 |
| 安装方法 | 17 |
| 测试方法 | 17 |
| 准备数据 | 17 |
| 压测性能 | 17 |
| 清理环境 | 17 |
| 测试结果 | 18 |
| 本文介绍KRDS MySQL 5.7实例的性能测试结果。 | 18 |
| 测试场景：磁盘IO型 | 18 |
| MySQL数据的备份与导入 | 18 |
| 数据的备份 | 18 |
| 数据恢复 | 19 |
| mysqldump常用参数 | 19 |
| MySQL Metadata lock的产生 | 20 |
| 简介 | 20 |
| 如何处理 | 20 |
| 如何避免 | 20 |
| MySQL查询空间使用情况 | 20 |
| 数据库占用空间 | 20 |
| 表占用空间 | 21 |
| 如何整理表碎片 | 21 |
| 如何通过binlog恢复数据库数据 | 21 |
| binlog简介 | 21 |
| binlog格式 | 21 |
| 查看、下载binlog | 22 |
| 通过binlog进行数据恢复 | 22 |
| terraform-ksyun | 22 |
| 使用案例 | 22 |
| Argument Reference | 24 |
| 会话管理 | 27 |
| 结束与 MySQL 服务器的连接 | 27 |
| 终止连接当前正在执行的语句 | 27 |
| 产品使用限制 | 28 |
| 产品内存核数关系 | 28 |
| 产品规格说明 | 28 |
| 使用Terraform管理RDS | 28 |
| 安装和配置 Terraform | 28 |
| 前往 Terraform 官网 下载适用于您的操作系统的程序包 | 28 |
| 创建 RAM 用户，并为其授权 | 28 |
| 登录 RAM 控制台 | 28 |
| 创建测试目录 | 29 |
| 创建配置文件 | 29 |
| 初始化工作目录 | 29 |
| 使用 Terraform 管理 RDS | 29 |
| SQL审批 | 30 |
| 前言 | 30 |
| 授权说明 | 30 |
| SQL审批 | 30 |
| 申请任务管理 | 30 |

| | |
|---------------|----|
| 创建SQL审批任务 | 30 |
| 查看SQL任务详情 | 30 |
| 执行结果 | 30 |
| 审批任务管理 | 31 |
| 详情 | 31 |
| 执行结果 | 31 |
| 自动审批设置 | 31 |
| 开启 | 31 |
| 关闭自动审批 | 31 |
| 数据库支持IPV6 | 32 |
| 使用限制 | 32 |
| 创建IPV6网段VPC | 32 |
| 创建IPV6网段终端子网 | 32 |
| 创建IPV6地址RDS实例 | 32 |
| 查看IPV6实例详情 | 32 |
| 申请外网IPV6地址 | 32 |
| IPV6实例关联安全组 | 32 |
| 数据库审计支持IPV6 | 32 |
| KingSQL 概述 | 32 |
| 线程池 | 33 |
| 限制条件 | 33 |
| 适用场景 | 33 |
| 性能测试结果 | 33 |
| 异步事务 | 33 |
| 限制条件 | 33 |
| 适用场景 | 34 |
| 锁优化 | 34 |
| 适用场景 | 34 |

版本

云数据库MySQL版本支持

目前MySQL数据库支持数据类型为MySQL 5.5、MySQL 5.6、MySQL 5.7、MySQL 8.0（部分地区）、Percona 5.6，后续会继续支持其他版本。

应用

云数据库MySQL应用案例

例1：通过云数据库MySQL一主多从，A公司轻松实现了业务高峰期的稳定运行。同时，金山云云数据库MySQL异地灾备服务，满足了A公司对数据高安全性和服务高可用性要求，让业务无后顾之忧。

例2：某MOBA手游通过使用云数据库MySQL全部基于PCIE SSD硬盘的数据库，获得超强数据库读写IO，轻松应对游戏中多人战斗的频繁数据访问。同时MySQL数据存储安全，回档机制灵活也为游戏运营带了稳定的数据支撑。

概览

1. 登录[云数据库MySQL控制台](#)。
2. 通过概览页，可查看当前实例个数、运行状态、区域概览等相关信息。



操作总览

实例列表页

更配

选中实例后，点击**更配**按钮，弹出**更改配置**窗口，选择配置后点击**更配**可进行实例的升级。



注意：升级期间，不能对实例进行重启、备份操作，同时数据库服务会出现30秒以内的闪断，且升级实例前会自动备份实例。

重启

选中实例后，点击实例操作的**重启**按钮，弹出提醒窗口，点击**确认**可进行实例的重启。



修改

选中实例后，点击实例操作的**修改**按钮，可进行实例配置的修改，包括实例名称、参数组、自动备份时间及管理员密码。



删除

选中实例后，点击实例操作的**删除**按钮可进行实例的删除。支持删除的实例包括高可用实例和只读实例。当删除实例类型为高可用时，关联的只读实例、临时实例也一并删除。

注意：计费方式为“包年包月”的实例不支持删除操作。



重置密码

选中实例后，点击实例操作的**重置密码**按钮，弹出**重置密码**窗口，输入新密码后点击**确认**可进行实例的密码重置。

创建只读实例

选中实例后，点击实例操作的**创建只读实例**按钮，跳转到**创建只读实例**页面，选择需要创建只读实例的配置等相关信息后，点击**购买**即可。

注意：创建只读实例的规格不能小于主实例，只读实例网络类型、数据库版本均继承主实例，删除主实例，只读实例同时删除。

分配至项目

选中实例后，点击实例操作的**分配至项目**按钮，弹出**分配至项目**窗口，选中想分配的项目后点击**确定**。

实例详情页

通过实例详情页，可查看当前实例的相关配置信息。

登录数据库

1. 在MySQL实例详情页左侧导航栏，点击**详情**，进入详情页面。
2. 点击**登录数据库**，打开**数据库登录**页面。

3. 填写用户名和密码，点击**执行**，进入数据库。

主备切换

1. 在实例详情页面，点击**更多**里的**主备切换**，弹出**实例主备可用区切换**窗口。
2. 显示当前主可用区和备可用区，点击**确定**，即可切换主备可用区。

注意：单机版不支持主备切换。

迁移可用区

1. 在实例详情页面，点击**更多**里的**迁移可用区**，弹出**迁移可用区**窗口。
2. 选择**可迁移备可用区**和**可迁移目标可用区**，点击**确定**，进行可用区迁移。

注意：可用区迁移过程期间不能对实例进行重启、备份操作，数据库服务会出现最长30秒的闪断，应用程序需要具有数据库重连机制自动重连的。

更改网络

更改VPC

1. 在实例详情页面，点击**更多**里的**更改网络**，弹出**更改网络**窗口。
2. 选择需要更改的VPC。
3. 选择终端子网。
4. 指定IP地址，并点击**确定**生效。

更改IP地址

1. 在实例详情页面，点击**更多**里的**更改网络**，弹出**更改网络**窗口。
2. 点击**更改IP**按钮。
3. 选择可用IP地址。
4. 点击**确定**生效。

更改端口

1. 在实例详情页面，点击**更多**里的**更改网络**，弹出**更改端口**窗口。
2. 输入自定义端口号。
3. 点击**确定**生效。

注意：更改网络IP和端口号会影响正在访问的数据库业务，需要及时更改程序连接数据库IP地址及端口号。

存储空间自动扩容

云数据库MySQL高可用版、云盘版、只读实例支持存储自动扩容，启用此功能会根据您设置的阈值自动扩展存储空间，降低由于存储空间不足导致服务不可用的风险。

前提条件

1. 当前实例状态为“运行中”。
2. 账户内余额充足。

配置存储空间自动扩容

1. 选择您需要配置的实例，进入详情，点击存储空间自动扩容功能右侧**设置**按钮进行存储空间自动扩容相关设置。



2. 设置存储空间自动扩容触发条件。若当实例可用存储空间达到设定阈值后，将自动对存储空间进行扩容。
3. 设置存储空间自动扩容上限。当存储空间达到该上限值后，即使剩余存储空间达到设定阈值，也不再自动扩容。
4. 添加自动扩容告警联系人。在自动扩容完成后，系统将通过短信/邮件的形式向您选定的联系人发送信息提醒。
5. 点击确定，即可保存并生效配置信息。

存储空间自动扩容规则

每次触发自动扩容，扩容步长为当前存储空间的15%。例如：实例存储空间为100G，自动扩容触发条件设置为存储使用空间的80%，当存储空间使用达到80G时，KRDS将进行自动扩容操作，当次扩容后存储空间为115G 【计算公式： $100 * (1 + 15\%)$ 】。

数据库代理

概述

KRDS for MySQL数据库代理是位于云数据库服务端和应用服务端之间的网络代理服务，用于代理应用服务访问云数据库时的所有请求。数据库代理通过服务地址提供服务，所有通过服务地址的请求将由代理实例根据请求类型转发至主实例或只读实例，实现自动读写分离，降低主库负载，提高业务请求处理能力。此外，数据库代理支持连接池功能，能有效解决短连接业务频繁建立新连接导致实例负载过高的问题。

注意：代理实例目前处于公测阶段，暂不遵循金山云关系型数据库服务等级协议（SLA），您可以免费使用，需要您发送工单申请权限。

功能特性

- 自动读写分离

通过在应用侧配置服务地址对应用服务访问云数据的请求进行代理，自动实现读写分离。所有的读请求将会根据读权重转发至只读实例，写请求将转发至主实例，降低主实例负载，提升整体实例负载能力。

- 多维服务监控

丰富的监控指标及图形化展示，能清晰反应数据库代理服务状态，为运维工作提供高效便捷的数据支持。


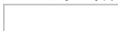
适用场景

1. 业务读请求数远大于写请求数，部分实例因读请求分配不均衡导致负载过高的场景。
2. 大量短连接为主要的业务频繁建连导致实例负载过高的场景。
3. 业务使用多个只读实例，且有业务隔离需求的场景。


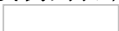

开通代理实例

注意：在开通数据库代理前，请确保当前主实例已关联至少一个只读实例。

1. 登录[云数据库MySQL控制台](#)。

2. 在MySQL实例详情页左侧导航栏，点击**数据库代理**，进入数据库代理页面。 3. 点击**开通数据库代理**，在服务开通页完成代理实例的选配与购买。 4. 代理实例创建完成后，即可进入代理服务页。在服务信息中，您可以查看当前数据库代理的基本信息；在读写分离中，您可以及时掌握已关联的实例信息及状态。

自动读写分离

1. 点击**配置读写分离**按钮，在弹出的配置窗口内进行读写分离配置。
2. 您可以在左侧的**未关联实例**列表中查看当前主实例下未关联到代理实例的只读实例，在右侧的**已关联实例**列表中查看代理实例已关联的只读实例。
3. 通过穿梭框中部的按钮您可以进行对只读实例进行关联与取消关联。您需要根据业务需求对需要关联的实例进行**读权重**配置。

4. 上述操作完成后，点击**确定**即可保存并生效配置信息。

连接方式

代理实例开通配置完成后，您需要在应用程序中配置代理服务地址及端口，默认服务端口为3306，配置完成后您的程序就可以使写请求自动转发到主实例，读请求根据您的权重配置自动转发到各个只读实例。

监控与日志

1. 在**性能监控**页，您可以通过实时的监控数据了解代理实例的运行状态，并根据业务需求及时调整代理实例规格。

2. 在**服务日志**页，您可以下载代理实例的服务日志用于排障与审计。

提示：当前KRDS for MySQL数据库代理的服务日志尚未上线，如需服务日志请提交工单。

关闭数据代理

警告：关闭数据库代理后，代理服务器地址将不可用，请您修改业务系统数据库的连接地址，避免由于代理服务地址不可用给您的业务带来影响。 1、在左侧导航栏点击数据库代理。 2、点击页面右上角**关闭数据代理**按钮。 3、点击弹出框右下角的**确定**按钮，释放代理实例。

版本升级

云数据库MySQL提供主版本和小版本升级功能，更新的版本可能包含数据库引擎的问题修复、性能提升、安全增强或新功能特性等。

主版本升级

当前支持云数据库MySQL如下形式的主版本升级：

MySQL 5.5 升级至 MySQL 5.6 MySQL 5.6 升级至 MySQL 5.7

升级前注意事项：

1. 检查新版数据库引擎的升级文档以了解是否存在可能会影响数据库或应用程序的兼容性问题。

- [MySQL 5.7中的改变](#)

2. 建议您在升级版本之前，通过备份恢复一个新实例进行业务测试验证，经过充分测试验证并确认您的业务功能正常后再升级原实例版本。 3. 在数据库升级过程中，云数据库MySQL服务会出现闪断，请您尽量在业务低峰期执行升级操作，或确保您的应用有自动重连机制。 4. 升级前请确保预留充足磁盘可用空间。

主版本升级步骤

1. 登录[RDS控制台](#)，选择需要升级的实例，进入详情页，在**配置信息**区域点击**升级数据库版本**。 2. 在弹出的对话框中，确认即将升级的目标版本，点击**开始升级**按钮进行主版本升级。您也可以点击**取消**按钮，取消本次升级操作。

小版本升级步骤

注意：当前仅MySQL 5.7版本支持小版本升级功能

1. 登录[RDS控制台](#)，选择需要升级的实例，进入详情页，在**配置信息**区域点击**升级数据库小版本**。若小版本号右侧未显示**升级数据库小版本**按钮，则表示您当前实例内核版本已经是最新版本。 2. 在弹出的对话框中，点击**确定**按钮进行小版本升级。您也可以点击**取消**按钮取消本次小版本升级操作。

只读实例

云数据库MySQL支持创建一个或多个只读实例，在对数据库有少量写请求，但有大量读请求的应用场景下，创建只读实例可以扩展数据库的读取能力。

前提条件

创建一个MySQL主实例。

操作步骤

1. 在MySQL实例列表页，选中目标实例，点击**实例操作**>**创建只读实例**。
2. 选择计费方式、可用区、内存和存储空间。
注意：只读实例的存储空间不得小于主实例。
3. 选择自动分配IP、指定IPV4或共享已有IP三种方式。只有主实例已经存在只读实例，才可以与先前创建的只读实例共享IP。
注意：从属于一个主实例的只读实例默认最多5个。

查看只读实例

在实例列表中查看只读实例

4. 在MySQL实例列表页，**类型**列选择**只读RDS**，查看只读实例。
5. 点击只读实例ID，查看示例详情，显示主实例信息。

在主实例的实例详情页查看只读实例

1. 在主实例列表页，点击主实例ID。
2. 在主实例的实例详情页，查看只读实例数量，将鼠标悬停于右侧图标，查看只读实例名称列表，点击只读实例ID。

账号管理

新建账号

1. 登录[云数据库MySQL控制台](#)。
2. 在MySQL实例详情页左侧导航栏，点击**账号管理**，进入**账号管理**页面。
3. 点击**新建账号**，进入**新建账号**页面。

4. 填写相关信息后，点击**确认**，账号创建成功。
5. 创建成功后，在**数据库管理**页面，对应授权数据库**绑定账号**下，显示刚刚新建的账号名称。

修改权限

1. 在账号管理页面，点击对应账号操作栏的**修改权限**。
2. 在左侧**未授权数据库**中，勾选要添加权限的账号，点击**授权**。
3. 在右侧**已授权数据库**中，可以选择权限类型，点击**确定**。
4. 在右侧**已授权数据库**中，选中要取消授权的账号，点击**移除**，删除账号权限。

克隆账号

1. 在账号管理页面，点击对应账号操作栏的**克隆账号**，弹出克隆账号窗口。
2. 其中**授权数据库**和**相关描述**与原账号相同，需要修改数据库账号（账号不能重复）。
3. 填写**密码**和**确认密码**，点击**确认**。

删除

1. 在账号管理页面，点击对应账号操作栏的**删除**按钮，弹出**删除账号**窗口。
2. 点击**确定**，即可删除相应账号。

注意：账号类型为管理员账号时，不支持删除，只支持删除普通账号。

3. 若想批量删除，勾选多个账号，点击操作栏的**删除**按钮。



4. 弹出**删除账号**窗口，点击**确定**，即可批量删除账号。

数据库管理

新建数据库

1. 登录[云数据库MySQL控制台](#)。
2. 在MySQL实例详情页左侧导航栏，点击**数据库管理**，进入**数据库管理**页面。
3. 点击**新建数据库**，进入**新建数据库**页面。
4. 填写相关信息后，点击**确认**，数据库创建成功。



注意：数据库权限栏显示的为账号管理中创建的普通账号。

修改权限

1. 在数据库管理页面，点击对应数据库操作栏的**修改权限**，弹出**修改数据库权限**窗口。
2. 在修改数据库权限页面中，勾选未授权账号，点击**授权**，为其添加权限。
3. 也可以在右侧**已授权账号**中，移除权限。



删除

1. 在数据库管理页面，点击对应数据库操作栏的**删除按钮**，弹出**删除数据库**窗口。
2. 点击**确定**，即可删除数据库。
3. 若想批量删除，勾选要删除的多个数据库。
4. 点击列表上方的**删除按钮**，即可批量删除。



实例监控

监控查看

1. 登录[云数据库MySQL控制台](#)。
2. 在MySQL控制台的左侧导航栏，点击**实例监控**，进入实例监控页面。在此您可以查看到系统层监控指标，引擎层监控指标以及连接信息监控指标。关于指标的含义介绍请参照[KRDS 关系型数据库](#)。 注意：目前云数据库MySQL最长支持查看近30天内的监控数据。

修改监控粒度

默认监控数据采集频率为每30秒采集一次，并且无法修改。如果需要更改实例监控粒度，您可以发起工单申请来进行监控粒度的调整。目前我们提供4种监控粒度：

- 5s
- 15s
- 30s
- 60s

操作步骤

1. 工单申请修改监控粒度申请通过后，您可以在实例监控页面的右上角查看到当前监控粒度。
2. 点击**修改**，选择您需要设置的监控粒度。
3. 点击**√确认**，完成监控粒度修改。

配置告警

金山云数据库产品告警配置，统一由云监控告警服务提供，请参照[新建告警策略](#)进行监控告警配置。

参数配置

新建参数组

1. 登录[云数据库MySQL控制台](#)。
2. 在MySQL控制台的左侧导航栏，点击**参数组**，进入参数组页面。
3. 点击**创建参数组**，进入新建参数组页面。
4. 填写相关信息后，点击**创建**，参数组创建成功。

应用参数组

1. 勾选要应用的参数组，点击**应用**。
2. 选择要应用的实例，点击**应用**即可应用参数组。

编辑参数组

点击参数组的**编辑**按钮，即可对参数组进行编辑。

删除参数组

1. 勾选要删除的参数组，点击操作栏下的**删除**按钮，弹出**删除参数组**窗口。
2. 点击**确定**，即可删除参数组。
3. 若想批量删除，勾选多个参数组，点击**删除**按钮。

注意：当参数组模板删除后，之前应用过此参数模板的实例依旧保留此参数组模板内配置参数。

实例参数配置

1. 在MySQL实例详情页左侧导航栏，点击**参数配置**，进入实例的参数配置页面。
2. 点击**修改**，进入修改参数配置页面，可对参数进行修改。
3. 点击**使用参数组**，弹出使用参数组窗口，在下拉框中选择已经存在的参数组，点击**应用**，即可使用该参数组。

注意：删除或修改参数组不影响实例参数。

4. 点击**生成参数组**，弹出生成参数组窗口，选择**新建参数组**或**覆盖已有参数组**，若选择**新建参数组**，填写相关信息，保存即可；若选择**覆盖已有参数组**，选择一个参数组，点击**确定**，即可覆盖。

安全组

• 新建安全组

1. 登录[云数据库MySQL控制台](#)
2. 在MySQL控制台的左侧导航栏，点击**安全组**，进入安全组页面。
3. 点击**新建**，弹出新建安全组窗口。
4. 填写安全组名称及描述后，点击**新建**，安全组新建成功。

• 删除安全组

1. 在安全组页面，点击对应安全组操作栏的**删除**按钮，弹出**删除安全组**窗口。
2. 点击**确定**，即可删除该安全组。

3. 若想批量删除，勾选要删除的多个安全组。
4. 点击列表上方的**删除**按钮，即可批量删除。

注意：有绑定实例的安全组，不可直接删除，需进入该安全组的详情页，点击实例管理，将所有绑定实例解绑后，方可删除。

● 克隆安全组

1. 在安全组页面，点击目标安全组的**克隆**。
2. 弹出**克隆安全组**对话框，填写**安全组名称**，**安全组描述**。
3. 点击**克隆**，即生成一个和原安全组IP规则一致的新安全组。

● 管理安全组

安全组添加/删除IP

1. 在安全组页面，点击列表中目标安全组或点击**管理**，进入到该安全组详情页。
2. 默认进入**安全组管理页** > **IP管理**分页。若想要添加此账号下已开通的主机IP，点击**添加主机IP**。
3. 弹出**添加主机IP**窗口，勾选列表中的IP，点击**确定**，即将所选IP添加至此安全组中。
4. 若想要手动添加IP，点击**添加IP**。弹出**添加IP**窗口，手动输入IP回车确认后，可继续输入下一条IP。输入完成后点击**确定**即可。

添加IP规则



输入IP段

IP段采用CIDR地址格式，CIDR地址中包含标准的32位IP地址和有关网络位数[0-32]的信息。如：其中“/24”表示其前面地址中的前24位代表网络部分，其余位代表主机部分

确定

取消

5. 您可以在此安全组中删除某些IP，在安全组**IP管理**分页，勾选IP，点击**删除**即可。

安全组添加/解绑实例

1. 若您需要在此安全组中添加实例，在**安全组详情页**，点击**实例管理**，点击**添加**。
2. 弹出**添加实例**窗口，列表显示此机房账号下的未绑定安全组的高可用、单机和只读实例，勾选实例，点击**确定**，即添加成功。

注意：此列表仅显示未绑定安全组的高可用、单机和只读实例，若想要添加已绑定安全组的实例，需用户进入目标实例已绑定的安全组管理页，将此实例手动解绑，方可进行绑定，或在目标实例详情的实例安全组页，点击**设置安全组**，手动为已绑定安全组的实例设置新安全组。

3. 如要在此安全组中解绑实例，在**安全组实例管理**页，勾选相应实例，点击**解绑**即可。

● 设置安全组

1. 在实例详情页，点击左侧菜单栏**实例安全组**，点击**设置安全组**。

2. 弹出设置安全组窗口，展示该账号此机房下已创建的安全组实例列表，选择一个安全组实例后，点击**确定**即可。

● 解绑安全组

1. 在**实例安全组**页面，选中实例点击**解绑安全组**即可。

2. 在安全组详情页中，勾选实例后，点击解绑即可，支持批量解绑。

● 导出

1. 若需要在本地查看安全组信息（安全组IP列表/绑定实例列表）的话，可以导出现在安全组信息到本地。勾选某个或者某些IP实例，甚至是全选。
2. 点击**导出**按钮之后，可以导出安全组信息到本地查看。

备份与恢复

1. 登录[云数据库MySQL控制台](#)。
2. 在MySQL实例详情页左侧导航栏，点击**备份恢复**，进入**备份恢复**页面。

注意：

3. 手动备份期间无法对实例进行重启、升级操作，且一个实例的手动备份最多有五个。
4. 建议单个实例表数量不超过100万，否则可能会造成升级、备份、监控失败，同时也可能会影响数据库监控。

手动备份

点击操作栏的**手动备份**按钮，弹出**备份实例**窗口，输入备份名称后点击**确定**可进行实例的备份。

注意：执行备份期间无法对实例进行重启、升级操作。

您也可以在实例列表页进行手动备份操作，勾选其中想要进行备份的一个或多个实例，点击**手动备份**按钮，添加**备份名称**之后，点击**确定**就可以单个或批量手动备份。一个实例只能最多创建5个手动备份。

恢复数据

1. 恢复某个备份集，点击该备份集的**恢复此备份**即可。
2. 备份可回档到本实例、临时实例、新实例。

3. 当恢复到本实例时，点击**确定**，即覆盖当前实例的数据，所以需谨慎操作。
4. 可以先恢复到临时实例，确认数据是否完整。

基于时间点恢复和库表级恢复

1. 点击操作栏的**库表恢复**，弹出**库表恢复**窗口。
2. 我们支持基于时间点恢复和库表级恢复，可以恢复到任意时间点以及任何一个指定库或表。

注意：

1. 含有只读实例的主实例不能回档到本实例。
2. 库表恢复每次最多恢复50个表。
3. 实例在非运行中时，无法使用库表级恢复。
4. 当实例中库表总数超过5万时，不支持库表恢复。

删除备份

1. 勾选备份，点击操作栏中的**删除**。
2. 弹出删除备份窗口，点击**确定**，即可删除备份，支持批量删除。

注意：自动备份的文件无法通过此种途径删除。

自动备份设置

1. 自动备份默认每24小时备份1次，可以自定义设置备份的时间，时间段时长为1小时。

注意：默认自动备份保留时间7天。

其他

备份下载链接有效期为30分钟，刷新页面可更新下载链接。

日志管理

1. 登录[云数据库MySQL控制台](#)。
2. 在MySQL实例详情页左侧导航栏，点击**日志管理**，进入日志管理页面。
3. 系统会列出实例7天内产生的慢日志、错误日志和binlog日志列表。点击**下载**按钮可以下载日志文件到本地，便于本地查看。

注意：控制台会自动清理上传超过一个月的binlog日志。

临时实例

1. 登录[云数据库MySQL控制台](#)。
2. 在MySQL实例详情页左侧导航栏，点击**临时实例**，进入临时实例页面。
3. 您可以进行**创建临时实例**、**升级正式实例**、**删除**操作。临时实例3日内有效，7日后系统自动删除。

外网访问

1. 如果您的云服务器与云数据库部署在不同地域或者在金山云以外的系统上，则您需开启外网IP来连接数据库。
2. 登录[云数据库MySQL控制台](#)，实例列表中，找到需要修改的实例，点击对应操作栏的**管理**。
3. 在配置信息里找到**外网地址**，点击**申请外网IP**。

注意：

1. 开启外网访问生效时间大概需要1分钟，启用成功后，您可以通过外网IP和端口访问云数据库MySQL。
2. 通过外网IP访问云数据库MySQL前，需设置访问白名单后方可正常连接云数据库MySQL。
3. 基于安全考虑建议云数据库MySQL外网访问仅用于辅助管理数据库或开发，业务访问建议使用内网访问。
4. 如使用同机房主机，不支持通过外网IP访问KRDS。

4. 开启成功后，即可在配置信息中查看到外网地址。

5. 开启成功后，若您希望关闭外网访问权限，则可以点击**释放外网IP**进行关闭外网访问权限。关闭后访问地址将回收，再次开启会重新分配。

注意：

1. 关闭外网访问后无法再通过外网IP和端口访问云数据库MySQL，关闭外网访问生效时间大概需要1分钟，请确认您的业务系统没有使用外网访问地址，以免给您带来不必要的损失。
2. 确认关闭外网访问后，建议及时删除安全组内的外网访问相关的白名单。
3. 若再次开启外网服务，则外网IP、端口都会改变。

目前金山云免费提供KRDS外网IP。

标签使用

编辑标签

1. 登录[云数据库MySQL控制台](#)。
2. 勾选需要添加标签的实例，点击**实例操作**按钮，选择菜单中的**编辑标签**。
3. 若您需要添加标签，在**编辑标签值**窗口，点击**添加**，选择**标签键**以及**标签值**，点击**确定**，即可添加成功。

4. 若您需要删除标签，在**编辑标签值**窗口，点击标签右侧的**删除**，点击**确定**，即可删除成功。

标签显示及筛选

1. 登录[云数据库MySQL控制台](#)。
2. 添加标签后，在实例列表**标签**列内会显示已添加的标签名称。

3. 点击**标签**名称，弹出**所属实例**窗体，显示同一标签实例的数量及实例列表。

数据库审计

数据库审计能够实时记录数据库实例活动，对数据库进行细粒度审计的合规性管理。通过对用户访问数据库行为的记录和分析，帮助用户对越权操作、恶意入侵等行为进行有效追溯和审计。

注意：数据库审计功能目前处于公测阶段，您可以免费使用，需要您发送工单申请权限。

开启数据库审计

数据库审计功能默认为关闭状态，在有需要时，可通过以下方式开启：

1. 提交工单申请数据库审计权限。
2. 申请成功后，登录[云数据库MySQL控制台](#)。
3. 在MySQL实例详情页左侧导航栏，点击**数据库审计**，进入审计日志查询页面。
4. 点击开启**数据库审计**按钮或开启**数据库审计**超链接，开启数据库审计功能。

关闭数据库审计

对于已开启数据库审计的实例对象，可在开启相同位置点击**关闭数据库审计**，停止数据库审计的功能。在数据库审计功能关闭期间审计数据将不会被记录。

审计日志查询

1. 数据库审计记录可通过**时间范围**、**关键字**、**数据库**、**用户**、**客户端IP**、**状态**、**SQL类型**进行查询。

2. 其中：**关键字**、**数据库**、**用户**、**客户端IP**可进行组合查询，字段间以“/”分隔。
3. **SQL类型**包括：SELECT、INSERT、UPDATE、DELETE、LOGIN、LOGOUT、MERGE、ALTER、CREATEINDEX、DROPINDEX、

CREATE、DROP、SET、DESC、REPLACE、CALL、BEGIN、DESCRIBE、FLUSH、USE、SHOW、START、COMMIT。

注意：由于数据库审计功能会记录所有数据库的操作，所以为了避免SQL语句过多而造成查询超时，数据库审计列表最多展示10000条审计结果，当导出结果大于10000条时，建议您通过导出审计结果功能导出后查看。

导出审计结果

1. 点击**导出审计结果**按钮，弹出**审计日志结果导出**窗口。



2. 其中，可通过勾选导出字段前的复选框组导出所需的审计日志。点击**导出**按钮，后台自动运行导出任务。



查看导出历史

1. 点击**查看导出历史**按钮，弹出**审计日志导出历史**窗口。
2. 当**导出进度**为导出完成时，点击**下载**超链接，可以将审计日志导出到本地查看。

审计监控图表

点击**监控图表**按钮进入审计监控图表页面，页面内会根据已有审计记录，自动生成所选时间维度，基于SQL类型和SQL操作命令的统计图表，以更加直观的方式辅助用户了解审计记录的整体情况。



DCL、DDL、DML、DQL、OTHER。（其中OTHER表示不在DCL、DDL、DML、DQL这四种类型中的其他所有操作语句）

审计日志分析

点击顶部的**审计日志分析**选项卡，进入**审计日志分析**页面。



通过可视化图表以及提取的SQL模板，帮助用户更好地分析审计日志结果。

注意：现有免费公测审计产品无法完整满足金融行业审计需求，可通过云市场选择专业审计产品满足需求。

性能分析

传统的企业对数据库运维、问题定位等工作多依赖于数据库运维人员，DBA的能力直接影响了数据库的“健康程度”。金山云性能分析服务能够很好地解决这一问题，帮助用户及时发现慢SQL等影响数据库的不稳定因素，保障数据库安全、高效的运行。

慢SQL

1. 登录[云数据库MySQL控制台](#)。
2. 在MySQL实例详情页左侧导航栏，点击**性能分析**>**慢SQL**，进入慢SQL分析页面，显示慢SQL趋势图。



注意：当没有开启数据库审计功能时，慢SQL趋势图不会展示慢SQL占比。

慢SQL统计

1. 点击下方的**慢SQL统计**选项卡，可以查看特定时间内以SQL模板为维度的慢SQL统计表。



2. 点击指定**慢SQL模板**名称，选项卡自动切换到**慢SQL详情**，并展示此模板内所有慢SQL语句。



慢SQL详情

1. 点击下方的慢SQL统计选项卡，可以查看特定时间内以SQL语句为维度的慢SQL详情表。
2. 还可以通过SQL类型（复选框）+SQL模板（下拉菜单）对特定时间内的慢SQL语句进行搜索。

热点表分析

1. 登录[云数据库MySQL控制台](#)。
2. 在MySQL实例详情页左侧导航栏，点击性能分析>热点表分析，进入热点表分析页面。
3. 分别展示选定时间范围内SQL执行次数TOP10与SQL执行耗时TOP10的表信息。



测试环境

本文介绍云数据库KRDS MySQL性能测试所使用的环境。

- 测试的KEC和KRDS实例均在同一地域、同一可用区。所有测试均在华北1（北京）可用区A完成。
- 测试的KEC和KRDS实例网络类型均为专有网络（VPC）且在同一VPC。
- 测试的KEC实例信息如下：实例规格为：标准型S4(8c16g)。实例镜像为：centos7.5 64位。

测试环境详细信息如下： 测试地区 华北1（北京）

| | |
|------|----------|
| 网络类型 | VPC网络 |
| 实例规格 | 控制台标准套餐 |
| 测试版本 | MySQL5.7 |

测试工具

本文介绍KRDS MySQL性能测试工具SysBench以及在KEC实例上安装SysBench。

SysBench工具介绍

本次使用的测试工具为sysbench，版本为1.0.19，是一个开源的、模块化的、跨平台的多线程性能测试工具。可以用来进行CPU、内存、磁盘I/O、线程、数据库的性能测试。目前支持的数据库有MySQL、Oracle和PostgreSQL。

安装方法

在KEC实例执行如下命令安装SysBench。

```
yum install gcc gcc-c++ autoconf automake make libtool bzip2 mysql-devel
wget https://github.com/akopytov/sysbench/archive/1.0.19.zip
unzip 1.0.19.zip
cd sysbench-1.0.19/
./autogen.sh
./configure --prefix=/usr --mandir=/usr/share/man
make
make install
sysbench -v
```

测试方法

准备数据

```
sysbench --threads=32 --time=1800 --events=999999999 --test=oltp_read_write.lua --table-size=10000000 --tables=10 --db-driver=mysql --mysql_storage_engine=innodb --mysql-host=xxx --mysql-port=3306 --mysql-user=xxx --mysql-password=xxx prepare
```

压测性能

```
sysbench --threads=32 --time=1800 --events=999999999 --test=oltp_read_write.lua --table-size=10000000 --tables=10 --db-driver=mysql --mysql_storage_engine=innodb --mysql-host=xxx --mysql-port=3306 --mysql-user=xxx --mysql-password=xxx run
```

清理环境

```
sysbench --threads=32 --time=1800 --events=999999999 --test=oltp_read_write.lua --table-size=10000000 --tables=10 --db-driver=mysql
```

```
ql --mysql_storage_engine=innodb --mysql-host=xxx --mysql-port=3306 --mysql-user=xxx --mysql-password=xxx cleanup
```

| SysBench参数说明 | 参数 | 说明 |
|--------------------------|----|--|
| threads | | 测试线程数。 |
| time | | 测试时间。 |
| events | | 测试请求数量。 |
| test=oltp_read_write.lua | | 表示调用oltp_read_write.lua脚本进行 oltp 模式测试。 |
| table-size | | 测试表大小。 |
| tables | | 测试表数量。 |
| db-driver | | 指定数据库驱动程序。 |
| mysql_storage_engine | | 数据库存储引擎。 |
| mysql-host | | KRDS实例连接地址。 |
| mysql-port | | KRDS实例连接端口。 |
| mysql-user | | KRDS实例账号。 |
| mysql-password | | KRDS实例账号对应的密码。 |

测试结果

本文介绍KRDS MySQL 5.7实例的性能测试结果。

测试场景：磁盘I/O型

磁盘I/O型场景指只有部分数据可以放到缓存里，查询过程中需要读写磁盘更新缓存。

SysBench参数：--table-size=10000000 --tables=10。

测试指标为TPS和QPS。

- TPS:每秒执行事务数(Transactions Per Second)，数据库每秒执行的事务数，以COMMIT成功次数为准。
- QPS:每秒执行请求数QPS(Queries Per Second)，数据库每秒执行的SQL数，包含INSERT、SELECT、UPDATE、DELETE、COMMIT等。

| 测试实例内存 | 最大连接数 | SysBench线程数 | QPS | TPS |
|--------|-------|-------------|----------|---------|
| 1G | 1200 | 32 | 2715.63 | 135.78 |
| 2G | 2400 | 32 | 6271.46 | 313.57 |
| 4G | 4800 | 32 | 15623.33 | 781.16 |
| 8G | 9600 | 32 | 28181.81 | 1409.09 |
| 12G | 14400 | 32 | 30133.76 | 1506.68 |
| 16G | 19200 | 32 | 48858.67 | 2442.93 |
| 24G | 19200 | 32 | 57689.44 | 2884.47 |
| 32G | 19200 | 32 | 61024.08 | 2903.67 |
| 48G | 19200 | 32 | 68173.43 | 3403.67 |
| 96G | 19200 | 32 | 78213.42 | 3810.67 |
| 128G | 19200 | 32 | 87320.79 | 4396.03 |

MySQL数据的备份与导入

mysqldump是MySQL官方提供的逻辑备份工具，它可以将原数据库中的数据对象和数据转换为SQL语句，并将这些SQL语句重放到目的数据库中，从而达到迁移数据的目的。同时，mysqldump支持将结果输出为CSV，带分隔符的文本，XML等格式。

详细的使用手册可以参见官方文档：<https://dev.mysql.com/doc/refman/5.6/en/mysqldump.html>

数据的备份

备份整个数据库

1. 进入到MySQL的bin目录下执行

```
mysqldump -h 192.168.1.1 -uroot -ppwd dbname &gt; \temp\db.sql
```

-h 选项指定机器的ip

-u 选项指定使用的用户名

-p 选项指定密码，密码和选项间不要有空格

dbname 需要备份的数据库名

2. 如果需要指定数据库端口，也可以使用下面语句来备份数据库表结构

```
mysqldump -h ip -P3306 -uroot -ppwd dbname &gt; \temp\db.sql
```

3. 只备份表结构，不备份数据

```
mysqldump -uroot -ppwd -d dbname &gt; \temp\db.sql •
```

4. 备份单张数据表

```
mysqldump -uroot -ppwd dbname tablename &gt; \temp\db.sql
```

5. 备份多张数据表

```
mysqldump -uroot -ppwd dbname table1 table2 &gt; \temp\db.sql
```

6. 远程备份

```
mysqldump -h ip -uroot -ppwd dbname &gt; \temp\db.sql
```

数据恢复

将上一步导出的数据导入到数据库中。可以通过这两种方式进行：

1. 利用管道重定向

```
mysql -h ip -uroot -ppwd -f &lt; \temp\db.sql
```

2. 用MySQL客户端

先登录MySQL客户端，然后执行：

```
source \temp\db.sql;
```

mysqldump常用参数

- --compress 客户端服务端均支持压缩协议时开启。（暂时不需要）
- --host=host_name 远程MySQL服务器的host。
- --password=password 远程MySQL服务器用户的密码。
- --port=port_num 远程MySQL服务器端口号。
- --user=user_name 远程MySQL服务器用户名。
- --max_allowed_packet 最大包长度，保证服务器客户端相适应。（暂时不需要）
- --net_buffer_length 服务端的该参数至少要大于等于客户端参数。（暂时不需要）
- --no-defaults 禁止读取配置文件，以免实例中有未知配置文件影响执行结果。
- --force 在dump过程中遇到错误也继续执行。
- --log-error=file_name 错误日志输出文件名设置。
- --character-sets-dir=dir_name 指定已安装字符集目录。
- --default-character-set=charset_name 指定默认字符集。
- --no-set-names 禁用输出文件中包含set names语句。
- --set-charset 在输出文件中添加set names语句。
- --apply-slave-statements 在slave执行备份时，会在change master to 语句前执行stop slave，在备份结束后执行start slave。
- --dump-slave=2 在slave执行备份，并附带change master to 信息，该信息是注释，并不会生效。（这个参数没有用处，

因为我们接下来增量的备份还是从这个节点，master的信息并没有什么用)

- `--master-data=2` 备份，以注释的形式导出master信息。该参数默认关闭`--lock-tables`，打开`--lock-all-tables`。开启`--delete-master-logs`。需要`reload`权限。删除binlog，这个参数慎重使用。
- `--all-databases` 备份全部表。
- `--databases` 备份指定数据库。
- `--events` 从mysql.event 表备份事件，这时帐号需要对mysql库有访问权限。
- `--no-data` 只备份表结构。
- `--routines` 从mysql.proc表备份函数和存储过程。需要对mysql库有访问权限。
- `--tables` 备份指定表。
- `--triggers` 备份触发器，默认开启，必须对备份的表有TRIGGER权限。
- `--skip-triggers` 关闭触发器备份。
- `--extended-insert` 使用`multiple-row insert` 语法。
- `--add-locks` 在结果文件中每个表前后分别添加LOCK TABLES，UNLOCK TABLES。这个会加速insert。
- `--lock-all-tables` 备份开启前使用全局锁。默认关闭`--single-transaction`，`--lock-tables`。
- `--lock-tables` 备份每个数据库前，在会对该库下所有表加读锁。数据库级别一致性。

MySQL Metadata lock的产生

简介

MySQL为了在并发环境下维护表元数据的数据一致性，在表上有活动事务（显式或隐式）的时候，不可以对元数据进行写入操作。因此 MySQL 引入了 metadata lock ，来保护表的元数据信息。

支持事务的 InnoDB 引擎表和 不支持事务的 MyISAM 引擎表，都会出现 metadata lock wait 等待现象。一旦出现 metadata lock wait 等待现象，后续所有对该表的访问都会阻塞在该等待上，导致连接堆积，业务受影响。

因此在对表进行上述操作时，如果表上有活动事务（未提交或回滚），请求写入的会话会等待在 metadata lock wait，会导致metadata lock wait出现的场景有：

1. 创建、删除索引。
2. 修改表结构、删除表。
3. 获取表上表级写锁 (`lock table tab_name write`)。

如何处理

`show processlist` 查看会话有长时间未完成的查询，使用`kill` 命令终止该查询。

如何避免

出现长时间 metadata lock wait 会导致表上相关查询阻塞，影响业务。我们需要尽量避免类此情况。

在业务低峰期执行创建表、修改表，或者创建索引等操作。

- 在到RDS的数据库连接建立后，设置会话变量`autocommit`为1或者on，比如`set autocommit=1;`或`set autocommit=on;`。
- 考虑使用事件来终止长时间运行的事务，比如下面的例子中会终止执行时间超过60分钟的事务。
- 执行上述1中操作前，设置会话变量`lock_wait_timeout`为较小值，比如`set lock_wait_timeout=30`。命令可以设置 metadata lock wait 的最长时间为 30 秒。避免长时间等待元数据锁影响表上其他业务查询。

MySQL查询空间使用情况

数据库占用空间

统计数据库占用空间可使用以下SQL:

```
select TABLE_SCHEMA,concat\(truncate\(\sum\(\DATA\_LENGTH+DATA\_FREE+INDEX\_LENGTH\)/1024/1024,2\),' MB'\) as total\_size, concat\(\truncate\(\sum\(\DATA\_LENGTH\)/1024/1024,2\),' MB'\) as data\_size,concat\(\truncate\(\sum\(\DATA\_FREE\)/1024/1024,2\),' MB'\) as data\_free, concat\(\truncate\(\sum\(\INDEX\_LENGTH\)/1024/1024,2\),' MB'\) as index\_size from information\_schema.tables group by TABLE_SCHEMA order by data\_length desc;
```

其结果中total_size为总占用空间, data_size为表占用空间, data_free为表碎片占用空间, index_size为表索引占用空间。

表占用空间

统计表占用空间方法可使用以下SQL:

```
select CONCAT(TABLE_SCHEMA, '.', TABLE_NAME) as table\_name,concat\(\truncate\(\sum\(\DATA\_LENGTH+DATA\_FREE+INDEX\_LENGTH\)/1024/1024,2\),' MB'\) as total\_size, concat\(\truncate\(\sum\(\DATA\_LENGTH\)/1024/1024,2\),' MB'\) as data\_size,concat\(\truncate\(\sum\(\DATA\_FREE\)/1024/1024,2\),' MB'\) as data\_free, concat\(\truncate\(\sum\(\INDEX\_LENGTH\)/1024/1024,2\),' MB'\) as index\_size from information\_schema.tables group by TABLE_NAME order by data\_length desc;
```

其结果中total_size为总占用空间, data_size为表占用空间, data_free为表碎片占用空间, index_size为表索引占用空间。

如何整理表碎片

```
optimize table table\_name;
```

备注:

1. MySQL官方建议不要经常(每小时或每天)进行碎片整理, 一般根据实际情况, 只需要每周或者每月整理一次即可。
2. OPTIMIZE TABLE只对MyISAM, BDB和InnoDB表起作用, 尤其是MyISAM表的作用最为明显。此外, 并不是所有表都需要进行碎片整理, 一般只需要对包含上述可变长度的文本数据类型的表进行整理即可。
3. 在OPTIMIZE TABLE运行过程中, MySQL会锁定表。

如何通过binlog恢复数据库数据

binlog简介

binlog,即二进制日志,它记录了数据库上的所有改变。改变数据库的SQL语句,执行结束时,将在binlog文件的末尾写入一条记录,同时通知语句解析器,语句执行完毕。

MySQL binlog日志有三种格式,分别为statement, mixed, 以及row。

binlog格式

1. statement

每一条会修改数据的SQL都会记录在binlog中。

- 优点

不需要记录每一行的变化,减少了binlog日志量,节约了IO,提高性能。相比row能节约多少性能与日志量,这个取决于应用的SQL情况,正常同一条记录修改或者插入row格式所产生的日志量还小于statement产生的日志量,但是考虑到如果带条件的update操作,以及整表删除,alter表等操作,row格式会产生大量日志,因此在考虑是否使用row格式日志时应该跟据应用的实际情况,其所产生的日志量会增加多少,以及带来的IO性能问题。

- 缺点

由于记录的只是执行语句,为了这些语句能在slave上正确运行,因此还必须记录每条语句在执行时的一些相关信息,以保证所有语句能在slave上得到和在在master端执行时候的相同结果。另外MySQL的复制,像一些特定函数功能,slave可与master上要保持一致会有很多相关问题(如sleep()函数, last_insert_id(), 以及user-defined functions(udf)会出现问题)。使用以下函数的语句也无法被复制:

- LOAD_FILE()
- UUID()
- USER()
- FOUND_ROWS()
- SYSDATE() (除非启动时启用了 --sysdate-is-now 选项) 同时在INSERT ... SELECT 会产生比 RBR 更多的行级锁。

2. row

不记录SQL语句上下文相关信息，仅保存哪条记录被修改。对于大部分场景一般都推荐客户使用row格式日志。

- 优点

binlog中可以不记录执行的SQL语句的上下文相关的信息，仅需要记录那一条记录被修改成什么了。所以row的日志内容会非常清楚的记录下每一行数据修改的细节。而且不会出现某些特定情况下的存储过程，function，以及trigger的调用和触发无法被正确复制的问题。

- 缺点

所有的执行的语句当记录到日志中的时候，都将以每行记录的修改来记录，这样可能会产生大量的日志内容，比如一条update语句，修改多条记录，则binlog中每一条修改都会有记录，这样造成binlog日志量会很大，特别是当执行alter table之类的语句的时候，由于表结构修改，每条记录都发生改变，那么该表每一条记录都会记录到日志中。

3. mixed (MySQL默认)

是以上两种level的混合使用，一般的语句修改使用statement格式保存binlog，如一些函数，statement无法完成主从复制的操作，则采用row格式保存binlog，MySQL会根据执行的每一条具体的SQL语句来区分对待记录的日志形式，也就是在statement和row之间选择一种。新版本的MySQL中对row level模式也做了优化，并不是所有的修改都会以row level来记录，像遇到表结构变更的时候就会以statement模式来记录。至于update或者delete等修改数据的语句，还是会记录所有行的变更。KRDS中默认使用mixed格式。

查看、下载binlog

KRDS在数据库服务器本地保存了最近1天的binlog文件，同时会每一个小时把binlog上传到KS3云存储系统（只保存7天以内的数据）。通过本地+KS3的binlog KRDS可实现7天内任意时间点的数据回档。目前KS3上的binlog暂不支持下载，后续会陆续支持。但是用户可以通过一些工具查看服务器上的最近一天的binlog。

1. 查看

在MySQL命令行中，用show master logs命令查看已有的binlog文件。

查看指定binlog文件的内容：

```
show binlog events in 'binlog.000001';
```

2. 下载

通过mysqlbinlog工具可以下载binlog文件到本地

```
mysqlbinlog -uusername -ppassword -h192.168.1.1 -P3306 -R binlog.000001 > log.txt
```

上述方式会以文本方式下载binlog到log.txt，如果想以二进制的方式下载binlog可以使用：

```
mysqlbinlog -uusername -ppassword -h192.168.1.1 -P3306 --raw -R binlog.000001
```

注意：请注意MySQL Client的版本与server一致，否则可能会出现下载失败的情况。可以通过命令查看客户端版本mysql -v (5.1版本过低，建议使用5.6.36或以上版本)。

通过binlog进行数据恢复

下载了binlog后，我们可以通过将binlog灌入数据库，让数据库重新执行binlog中的操作。

```
mysqlbinlog /usr/local/mysql/data/mysql-bin.000001 | mysql -h192.168.1.1 -uroot -p123456
```

terraform-ksyun

资源 resource ksyun_krds Provides an RDS instance resource. A DB instance is an isolated database environment in the cloud. A DB instance can contain multiple user-created databases. 提供RDS实例资源。数据库实例是云中的隔离数据库环境。一个数据库实例可以包含多个用户创建的数据库。

使用案例

```
»Create a RDS MySQL instance  
创建一个RDS实例
```

```
provider "ksyun" {  
  region = "cn-shanghai-3"  
  access_key = ""  
  secret_key = ""  
}
```

```
variable "available_zone" {
```

```
    default = "cn-shanghai-3a"
  }
  resource "ksyun_vpc" "default" {
    vpc_name = "ksyun-vpc-tf"
    cidr_block = "10.7.0.0/21"
  }
  resource "ksyun_subnet" "foo" {
    subnet_name = "ksyun-subnet-tf"
    cidr_block = "10.7.0.0/21"
    subnet_type = "Reserve"
    dhcp_ip_from = "10.7.0.2"
    dhcp_ip_to = "10.7.0.253"
    vpc_id = "${ksyun_vpc.default.id}"
    gateway_ip = "10.7.0.1"
    dns1 = "198.18.254.41"
    dns2 = "198.18.254.40"
    availability_zone = "${var.available_zone}"
  }

  resource "ksyun_krds_security_group" "krds_sec_group_14" {
    output_file = "output_file"
    security_group_name = "terraform_security_group_14"
    security_group_description = "terraform-security-group-14"
    security_group_rule {
      security_group_rule_protocol = "182.133.0.0/16"
      security_group_rule_name = "asdf"
    }
    security_group_rule {
      security_group_rule_protocol = "182.134.0.0/16"
      security_group_rule_name = "asdf2"
    }
  }

  resource "ksyun_krds" "my_rds_xx" {
    output_file = "output_file"
    db_instance_class = "db.ram.2|db.disk.21"
    db_instance_name = "houbin_terraform_1-n"
    db_instance_type = "HRDS"
    engine = "mysql"
    engine_version = "5.7"
    master_user_name = "admin"
    master_user_password = "123qweASD123"
    vpc_id = "${ksyun_vpc.default.id}"
    subnet_id = "${ksyun_subnet.foo.id}"
    bill_type = "DAY"
    security_group_id = "${ksyun_krds_security_group.krds_sec_group_14.id}"
    preferred_backup_time = "01:00-02:00"
    availability_zone_1 = "cn-shanghai-3a"
    availability_zone_2 = "cn-shanghai-3b"
  }
  »Create a RDS MySQL instance with specific parameters
  创建一个带有明确参数的RDS实例
  provider "ksyun" {
    region = "cn-shanghai-3"
    access_key = ""
    secret_key = ""
  }

  variable "available_zone" {
    default = "cn-shanghai-3a"
  }
  resource "ksyun_vpc" "default" {
    vpc_name = "ksyun-vpc-tf"
    cidr_block = "10.7.0.0/21"
  }
  resource "ksyun_subnet" "foo" {
    subnet_name = "ksyun-subnet-tf"
    cidr_block = "10.7.0.0/21"
    subnet_type = "Reserve"
    dhcp_ip_from = "10.7.0.2"
    dhcp_ip_to = "10.7.0.253"
    vpc_id = "${ksyun_vpc.default.id}"
    gateway_ip = "10.7.0.1"
    dns1 = "198.18.254.41"
    dns2 = "198.18.254.40"
    availability_zone = "${var.available_zone}"
  }

  resource "ksyun_krds_security_group" "krds_sec_group_14" {
    output_file = "output_file"
    security_group_name = "terraform_security_group_14"
    security_group_description = "terraform-security-group-14"
    security_group_rule {
      security_group_rule_protocol = "182.133.0.0/16"
```

```

    security_group_rule_name = "asdf"
  }
  security_group_rule {
    security_group_rule_protocol = "182.134.0.0/16"
    security_group_rule_name = "asdf2"
  }
}

resource "ksyun_krds" "my_rds_xx" {
  output_file = "output_file"
  db_instance_class = "db.ram.2|db.disk.21"
  db_instance_name = "houbin_terraform_l-n"
  db_instance_type = "HRDS"
  engine = "mysql"
  engine_version = "5.7"
  master_user_name = "admin"
  master_user_password = "123qweASD123"
  vpc_id = "${ksyun_vpc.default.id}"
  subnet_id = "${ksyun_subnet.foo.id}"
  bill_type = "DAY"
  security_group_id = "${ksyun_krds_security_group.krds_sec_group_14.id}"
  preferred_backup_time = "01:00-02:00"
  parameters {
    name = "auto_increment_increment"
    value = "8"
  }

  parameters {
    name = "binlog_format"
    value = "ROW"
  }

  parameters {
    name = "delayed_insert_limit"
    value = "108"
  }
  parameters {
    name = "auto_increment_offset"
    value = "2"
  }
  availability_zone_1 = "cn-shanghai-3a"
  availability_zone_2 = "cn-shanghai-3b"
}

```

Argument Reference

The following arguments are supported: 支持下列参数

- `output_file` (Required) 将返回内容存储的文件名
- `db_instance_class` (Required) 数据库规格，
- 正则格式为 `db.ram.d{1,3}|db.disk.d{1,5}`，
- `db.ram`表示实例的运行内存大小，`db.disk`表示运行硬盘大小

-this value regex `db.ram.d{1,3}|db.disk.d{1,5}` , `db.ram` is rds random access memory size, `db.disk` is disk size

`db_instance_name` (Required) 实例名称
`db_instance_type` (Required) 实例类型支持HRDS
`engine` (Required) 数据库引擎名称mysql|percona
-engine is db type, only support mysql|percona
`engine_version` (Required) 数据库引擎版本, 修改时只支持升级版本
db engine version only support 5.5|5.6|5.7|8.0
`master_user_name` (Required) 数据库主账号名称
`master_user_password` (Required) 主账号密码
`vpc_id` (Required) 虚拟私有网络的id
`subnet_id` (Required) 子网id
`bill_type` (Required) 账单类型,
YEAR_MONTH (包年包月),
DAY (按日计费), 默认值: YEAR_MONTH
`duration` (Optional) 购买时长, 以月为单位
`security_group_id` (Optional) 安全组id
`preferred_backup_time` (Optional) 备份时间
`availability_zone_1` (Optional) 可用区1
`availability_zone_2` (Optional) 可用区2
`project_id` (Optional) 子项目id
`parameters` (Optional) 数据库参数
`port` (Optional) 端口号

NOTE: Because of data backup and migration, change DB instance type and storage would cost 15~30 minutes, or even more. Please make full preparation before changing them.

注意: 数据库实例在修改时如果实例正在备份状态下请稍后重试, 备份由于可能时间较长请您耐心等待

»Attributes Reference The following attributes are exported:

id - The RDS instance ID. port - RDS database connection port.

»Timeouts NOTE: Available in 1.52.1+.

The timeouts block allows you to specify timeouts for certain actions:

create - (Defaults to 30 mins) Used when creating the db instance (until it reaches the initial Running status). update - (Defaults to 30 mins) Used when updating the db instance (until it reaches the initial Running status). delete - (Defaults to 10 mins) Used when terminating the db instance.

ksyun_krds_rr Provides an RDS Read Only instance resource. A DB read only instance is an isolated database environment in the cloud. 提供RDS只读实例资源。数据库只读实例是云中的隔离数据库环境。 »使用案例 »Create a RDS RR MySQL instance resource "ksyun_krds_rr" "my_rds_rr" { output_file = "output_file" db_instance_identifier = "***" db_instance_class = "db.ram.2|db.disk.50" db_instance_name = "houbin_terraform_888_rr_1" bill_type = "DAY" security_group_id = "***" }

parameters { name = "auto_increment_increment" value = "7" }

parameters { name = "binlog_format" value = "ROW" } } »Argument Reference

The following arguments are supported:

支持下列参数

output_file (Required) 将返回内容存储的文件名

db_instance_identifier (Required) 传入RDS 高可用实例的实例id, 一个RDS高可用实例最多3个只读实例

db_instance_class (Required) 数据库规格, 正则格式为 db.ram.d{1,3}|db.disk.d{1,5}, db.ram表示实例的运行内存大小, db.disk表示运行硬盘大小, 只读实例的规格不能小于高可用实例的规格

-this value regex db.ram.d{1,3}|db.disk.d{1,5}, db.ram is rds random access memory size, db.disk is disk size

db_instance_name (Required) 实例名称

bill_type (Required) 账单类型, YEAR_MONTH (包年包月), DAY (按日计费), 默认值: YEAR_MONTH

duration (Optional) 购买时长, 以月为单位

security_group_id (Optional) 安全组id

project_id (Optional) 子项目id

parameters (Optional) 数据库参数

NOTE:RDS RR do not support modify

RDS只读实例只支持创建和删除

»Attributes Reference The following attributes are exported:

id - The RDS instance ID. port - RDS database connection port.

»Timeouts NOTE: Available in 1.52.1+.

The timeouts block allows you to specify timeouts for certain actions:

create - (Defaults to 30 mins) Used when creating the db instance (until it reaches the initial Running status). update - (Defaults to 30 mins) Used when updating the db instance (until it reaches the initial Running status). delete - (Defaults to 10 mins) Used when terminating the db instance.

ksyun_sqlserver

Provides an SqlServer instance resource. A DB instance is an isolated database environment in the cloud. A DB instance can contain multiple user-created databases.

提供SqlServer 实例资源。SqlServer 数据库实例是云中的隔离数据库环境。 一个数据库实例可以包含多个用户创建的数据库。

»使用案例

»Create a RDS MySQL instance 创建一个RDS实例 provider "ksyun" { region = "cn-shanghai-3" access_key = "" secret_key = "" }

variable "available_zone" { default = "cn-shanghai-3a" } resource "ksyun_vpc" "default" { vpc_name = "ksyun-vpc-tf" cidr_block = "10.7.0.0/21" } resource "ksyun_subnet" "foo" { subnet_name = "ksyun-subnet-tf" cidr_block = "10.7.0.0/21" subnet_type = "Reserve" dhcp_ip_from = "10.7.0.2" dhcp_ip_to = "10.7.0.253" vpc_id = "\${ksyun_vpc.default.id}" gateway_ip = "10.7.0.1" dns1 = "198.18.254.41" dns2 = "198.18.254.40" availability_zone = "\${var.available_zone}" }

resource "ksyun_sqlserver" "sqlserver-1" { output_file = "output_file" dbinstanceclass = "db.ram.2|db.disk.20" dbinstancename = "ksyun_sqlserver_1" dbinstancetype = "HRDS_SS" engine = "SQLServer" engineversion = "2008r2" masterusername = "admin" masteruserpassword = "123qweASD" vpc_id = "\${ksyun_vpc.default.id}" subnet_id = "\${ksyun_subnet.foo.id}" billtype = "DAY" } »Argument Reference

The following arguments are supported:

支持下列参数

output_file (Required) 将返回内容存储的文件名
 db_instance_class (Required) 数据库规格, 正则格式为 db.ram.d{1,3}|db.disk.d{1,5}, db.ram表示实例的运行内存大小, db.disk表示运行硬盘大小
 -this value regex db.ram.d{1,3}|db.disk.d{1,5}, db.ram is rds random access memory size, db.disk is disk size
 db_instance_name (Required) 实例名称
 db_instance_type (Required) 实例类型支持HRDS_SS
 engine (Required) 数据库引擎名称SQLServer
 -engine is db type, only support SQLServer
 engine_version (Required) 数据库引擎版本
 db engine version only support 2008r2,2012,2016
 master_user_name (Required) 数据库主账号名称
 master_user_password (Required) 主账号密码
 vpc_id (Required) 虚拟私有网络的id
 subnet_id (Required) 子网id
 bill_type (Required) 账单类型, , YEAR_MONTH (包年包月), DAY (按日计费), 默认值: YEAR_MONTH
 duration (Optional) 购买时长, 以月为单位
 security_group_id (Optional) 安全组id
 preferred_backup_time (Optional) 备份时间
 project_id (Optional) 子项目id
 parameters (Optional) 数据库参数
 port (Optional) 端口号

NOTE: SQLServer not support modify
 SQLServer 暂时不支持修改

»Attributes Reference The following attributes are exported:

id - The RDS instance ID. port - RDS database connection port.

»Timeouts NOTE: Available in 1.52.1+.

The timeouts block allows you to specify timeouts for certain actions:

create - (Defaults to 30 mins) Used when creating the db instance (until it reaches the initial Running status). update - (Defaults to 30 mins) Used when updating the db instance (until it reaches the initial Running status). delete - (Defaults to 10 mins) Used when terminating the db instance.

ksyun_krds_security_group
 提供RDS安全组功能
 »使用案例

```
»Create a RDS Security Group instance resource "ksyun_krds_security_group" "krds_sec_group_13" {
  output_file = "output_file"
  security_group_name = "terraform_security_group_13"
  security_group_description = "terraform-security-group-13"
  security_group_rule {
    security_group_rule_protocol = "182.133.0.0/16"
    security_group_rule_name = "asdf"
  }
  security_group_rule {
    security_group_rule_protocol = "182.134.0.0/16"
    security_group_rule_name = "asdf2"
  }
}
```

»Argument Reference

The following arguments are supported:
 支持下列参数

output_file (Required) 将返回内容存储的文件名
 security_group_name (Required)
 security_group_description (Optional)

security_group_rule (Optional) 安全组规则

--security_group_rule_protocol (Required) 0.0.0.0/32格式

--security_group_rule_name (Required) 不超过256个字节, 仅支持中文、大小写字母、数字、减号和下划线

»Attributes Reference The following attributes are exported:

id - The RDS instance ID. port - RDS database connection port.

»Timeouts NOTE: Available in 1.52.1+.

The timeouts block allows you to specify timeouts for certain actions:

create - (Defaults to 10 mins) Used when creating the db instance (until it reaches the initial Running status). update - (Defaults to 10 mins) Used when updating the db instance (until it reaches the initial Running status). delete - (Defaults to 10 mins) Used when terminating the db instance.

数据查询资源 data source

ksyun_krds

通过ksyun_krds查询HRDS、RDS-RR

```
data "ksyun_krds" "search-krds" {
  output_file = "output_file"
  db_instance_identifier = "***"
  db_instance_type = "HRDS,RR,TRDS"
  keyword = ""
  order = ""
  project_id = ""
  marker = ""
  max_records = ""
}
```

output_file (Required) 将返回内容存储的文件名

db_instance_identifier (Optional) 实例ID (传入实例ID, 获取的是该实例的详情, 否则则获取list)

db_instance_type (Optional) HRDS (高可用), RR (只读实例), TRDS (临时实例)

db_instance_status (Optional) ACTIVE (运行中) / INVALID (请续费)

keyword (Optional) 按名称/VIP模糊过滤

order (Optional) 区分大小写, 取值范围: DEFAULT (默认排序方式), GROUP (按复制组排序, 会把只读实例排在所属主实例的后面)

project_id (Optional) 默认值为所有项目

Marker (Optional) 记录开始偏移量

MaxRecords (Optional) 每页结果中包含的最大条数 取值范围: 1-100

ksyun_sqlservers

通过ksyun_sqlservers查询HRDS-SS

```
data "ksyun_sqlservers" "search-sqlservers" {
  output_file = "output_file"
  db_instance_identifier = "***"
  db_instance_type = "HRDS-SS"
  keyword = ""
  order = ""
  project_id = ""
  marker = ""
  max_records = ""
}
```

output_file (Required) 将返回内容存储的文件名

db_instance_identifier (Optional) 实例ID (传入实例ID, 获取的是该实例的详情, 否则则获取list)

db_instance_type (Optional) HRDS (高可用), RR (只读实例), TRDS (临时实例)

db_instance_status (Optional) ACTIVE (运行中) / INVALID (请续费)

keyword (Optional) 按名称/VIP模糊过滤

order (Optional) 区分大小写, 取值范围: DEFAULT (默认排序方式), GROUP (按复制组排序, 会把只读实例排在所属主实例的后面)

project_id (Optional) 默认值为所有项目

Marker (Optional) 记录开始偏移量

MaxRecords (Optional) 每页结果中包含的最大条数 取值范围: 1-100

ksyun_krds_security_groups

通过ksyun_krds_security_groups查询安全组

```
data "ksyun_sqlservers" "search-sqlservers" {
  output_file = "output_file"
  security_group_id = 123
}
```

output_file (Required) 将返回内容存储的文件名

security_group_id (Optional) 安全组id

会话管理

您可以通过执行相应的存储过程来结束非自身用户发起的连接线程或查询语句。

结束与 MySQL 服务器的连接

- **语法:** `call mysql.admin_kill($ProcessID);`
- **参数详解:** \$ProcessID: 连接的线程 ID, 您可以通过MySQL客户端连接到您的KRDS服务, 执行show processlist来获取线程ID信息。
- **示例** 以下示例结束线程 ID 为 296的连接: `call mysql.admin_kill(296);`

终止连接当前正在执行的语句

- **语法:** `call mysql.admin_kill_query($ProcessID);`
- **参数详解:** \$ProcessID: 连接的线程 ID, 您可以通过MySQL客户端连接到您的KRDS服务, 执行show processlist来获取线程ID信息。

- **示例** 以下示例终止ID 为 296的查询语句：`call mysql.admin_kill(296);`

产品使用限制

| 限制项 | 限制描述 | 申请例外支持 |
|----------|--|--------|
| 内存最大规格 | 内存最大规格为48G | 工单申请 |
| 磁盘最大规格 | 磁盘最大规格为2000G | 工单申请 |
| 最大连接数 | 每G内存提供200的最大连接数 | 工单申请 |
| 读写IOPS | 每G内存读写均为1200 | 不支持例外 |
| 购买数量 | 最大批量购买数量为20个 | 工单申请 |
| 只读实例数量 | 从属于一个高可用实例的只读实例默认最多5个 | 工单申请 |
| 数据库备份 | 手动备份最多5个 控制台默认提供7天备份 高可用实例15天 | 不支持例外 |
| 默认过期删除时间 | 单机版实例7天 临时实例1天 只读实例1天 | 不支持例外 |
| 慢日志、错误日志 | 保留时间30天 | 不支持例外 |
| 受限权限 | 基于稳定性及安全性考虑，默认限制super, file, Shutdown, Create_tablespace权限 | 工单申请 |
| 监控粒度 | 默认监控数据采集频率为每30秒采集一次 | 工单申请 |

产品内存核数关系

产品规格说明

| 编号 | CPU核心数 (Core) | 内存 (GB) | 备注 |
|----|---------------|---------|---------------|
| 1 | 2 | 1 | RDS for MySQL |
| 2 | 2 | 2 | RDS for MySQL |
| 3 | 4 | 4 | RDS for MySQL |
| 4 | 6 | 8 | RDS for MySQL |
| 5 | 6 | 12 | RDS for MySQL |
| 6 | 10 | 16 | RDS for MySQL |
| 7 | 10 | 24 | RDS for MySQL |
| 8 | 12 | 32 | RDS for MySQL |
| 9 | 16 | 48 | RDS for MySQL |

使用Terraform管理RDS

本文主要介绍如何安装配置 Terraform 及使用 Terraform 管理 RDS。

安装和配置 Terraform

使用 Terraform 前，您需要按照以下步骤安装并配置 Terraform。

前往 Terraform 官网 下载适用于您的操作系统的程序包

本文以 Linux 系统为例。将程序包解压到 /usr/local/bin。如果将可执行文件解压到其他目录，则需要将路径加入到全局变量。运行 Terraform 验证路径配置，若显示可用的 Terraform 选项的列表，表示安装完成。 [root@test bin]#terraform Usage: terraform [-version] [-help] [args]

创建 RAM 用户，并为其授权

登录 RAM 控制台

创建名为 Terraform 的 RAM 用户，并为该用户创建 AccessKey。具体步骤参见创建 RAM 用户。为 RAM 用户授权。您可以根据实际的情况为 Terraform 授予合适的管理权限。具体步骤参见为 RAM 用户授权。*注意 请不要使用主账号的 AccessKey 配置Terraform 工具

创建测试目录

因为每个 Terraform 项目都需要创建 1 个独立的执行目录，所以先创建一个测试目录 terraform-test。 [root@test bin]#mkdir terraform-test 进入 terraform-test 目录。 [root@test bin]#cd terraform-test [root@test terraform-test]#

创建配置文件

Terraform 在运行时，会读取该目录空间下所有 .tf 和 .tfvars 文件。因此，您可以按照实际用途将配置信息写入到不同的文件中。下面列出几个常用的配置文件： provider.tf -- provider 配置 terraform.tfvars -- 配置 provider 要用到的变量 variable.tf -- 通用变量 resource.tf -- 资源定义 data.tf -- 包文件定义 output.tf -- 输出 例如创建 provider.tf 文件时，您可按以下格式配置您的身份认证信息： provider.tf provider "ksyun" { region = "cn-beijing-6" access_key = "LTA**N02" secret_key = "M0k8x0*****wwff" }

初始化工作目录

```
[root@test terraform-test]#terraform init
```

Initializing provider plugins...

- Checking for available provider plugins on <https://releases.hashicorp.com...>
- Downloading plugin for provider "ksyun" ...

注意 每个 Terraform 项目在新建 Terraform 工作目录并创建配置文件后，都需要初始化工作目录。以上操作完成之后，您就可以使用 Terraform 工具了。

使用 Terraform 管理 RDS

Terraform 安装完成之后，您就可以通过 Terraform 的操作命令管理 RDS了，下面介绍几个常用的操作命令： terraform plan: 预览功能，允许在正式执行之前查看将要执行那些操作。 例如，您添加了一个创建 RDS的配置文件 test.tf :

```
test.tf
resource "ksyun_krds" "test-ss-1" {
  output_file = "output_file"
  dbinstanceclass= "db.ram.16|db.disk.500"
  dbinstancename = "ksyun_rds_1"
  dbinstancetype = "HRDS"
  engine = "mysql"
  engineversion = "5.7"
  masterusername = "admin"
  masteruserpassword = "123qweASD"
  vpcid = "cb*****8aae"
  subnetid = "87*****eb47"
  billtype = "DAY"
}
```

使用 terraform plan 可查看到将会执行的操作。

```
[root@test terraform-test]# terraform plan
Refreshing Terraform state in-memory prior to plan..
The refreshed state will be used to calculate this plan, but will not be
persisted to local or remote state storage.
```

An execution plan has been generated and is shown below. Resource actions are indicated with the following symbols:

- create Terraform will perform the following actions: Plan: 1 to add, 0 to change, 0 to destroy.

terraform apply: 执行工作目录中的配置文件。 例如您想创建一个名为test-ss-1 的 RDS 之后使用 terraform apply 命令执行配置文件即可。

```
[root@test terraform-test]#terraform apply
```

An execution plan has been generated and is shown below. Resource actions are indicated with the following symbols:

- create Terraform will perform the following actions: Apply complete! Resources: 1 added, 0 changed, 0 destroyed. 说明 此配置运行后，若 test-ss-1 这个 RDS不存在，则创建一个 RDS。

如果想删除此RDS则将test-ss-1资源从文件中删除。

Terraform支持更新RDS实例。

SQL 审批

前言

授权说明

1. 登录[云数据库MySQL控制台](#)。
2. 输入完成后将会进入您的金山云主页面，之后将鼠标移入右上角账号时，将会出现下拉框，点击**账户总览**。
3. 点击完毕后进入**账户总览**页面。
4. 进入**用户总览**页面后继续将鼠标移入右上角的账号时，当出现下拉框继续点击**访问控制**后进入该页面。

5. 进入该页面后点击左菜单栏中的人员管理下的子用户并在该页面中选择预授权的用户，点击**添加授权**。

6. 点击完毕后将会弹出**权限添加**页面。

7. 当进入该页面时：在**全部策略**中填写**workflow**后将会出现两种策略状态：
 - 子账号：只拥有对工作进行提交的权限。
 - 主账号：拥有对工作进行审批的权限。
8. 根据自己工作所需，可为用户设置权限，勾选预赋予的权限，点击 > 后点击**确定**即可。

SQL 审批

申请任务管理

创建SQL审批任务

1. 登录[云数据库MySQL控制台](#)。
2. 在MySQLSQL审批页，选择**申请任务管理**，点击**创建SQL审批任务**。

3. 进入**创建SQL审批任务**页面。

4. 输入相关信息后点击提交即可。
 - 保存：创建任务及ID;任务状态置为待提交。
 - 提交：创建任务及ID; 任务状态置为审批中。
 - 取消：弹窗确认是否需要保存，否为直接返回。

注意：标题字数不得超过20个字符。审批人可选多人。

查看SQL任务详情

1. 登录[云数据库MySQL控制台](#)。
2. 点击**标题/任务ID**在操作列表下点击**详情**，进入到任务的详情页面。

3. 在此页面，可详细查看到该任务的具体信息。

执行结果

注意：该执行结果是在任务执行失败的时候才会出现。 选择**标题/任务ID**在任务状态为执行失败后，找到**执行结果**点击进入。

进入该页面可查看任务状态及审批人等相关信息，如想查看结果，请点击**详情**即可，然后将会出现一弹窗，告知用户执行结果。

审批任务管理

在MySQLSQL审批页，选择**审批任务管理**。

详情

进入**审批任务管理**页选择一**标题/任务ID**后的详情，点击进入。

在该页面可查看到该**标题/任务ID**的具体信息，查看完毕后点击**返回**即可。

• 执行结果

注意： 执行结果只在**任务状态**为执行成功或失败的情况下显示。 选择一个执行成功或任务失败的标题点击**执行结果**。

点击进入**任务执行结果**页面。

1. 在此页面可查看到该任务的具体信息，如想查看执行失败的原因，可点击**查看结果**下的**详情**。

自动审批设置

点击**SQL审批**进入SQL审批页面后点击**审批任务管理** > **自动审批设置**。

开启

1. 进入**自动审批设置**页面时，可根据您自身的需求对实例的状态进行关闭或开启，当您要开启对实例审批时。
2. 点击开启将出现**自动审批设置**弹窗。

在该页面输入相关信息即可。

1、数据库账号及密码：请输入本实例的数据库账号。

2、自动审批类型：分为两类。

1) 周期性审批：每日所选时间范围内执行自动审批。

2) 时间范围审批：所选时间范围内执行自动审批。

时间范围可进行逐条增加或删除。

注意： 请填写本实例的数据库账号，为确保SQL可执行请保证所填数据库账号具备执行所审批SQL的权限。

关闭自动审批

勾选已开启自动审批时间段的实例，点击**关闭自动审批**。

注意： 为开启自动审批时间段的实例，复选框为置灰状态，是无法点击的，只有已开启自动审批时间段的实例复选框可进行勾选。

数据库支持IPV6

使用限制

1. IPV6当前为内测功能，如需使用请发工单进行权限申请。
2. IPV6仅开放广州机房的使用权限，其他机房暂不支持。

创建IPV6网段VPC

详情请见[新建IPV4/IPV6双栈VPC](#)（因RDS仅支持广州机房的IPV6服务；请在广州机房创建对应的IPV6和VPC）。

创建IPV6网段终端子网

在IPV6网段VPC下创建IPV6网段终端子网。

创建IPV6地址RDS实例

1. 登录[云数据库MySQL控制台](#)。
2. 当您具备IPV6功能权限后可选中广州机房创建目标规格实例。
3. 网络配置中选择您预先创建好的具备IPV6网段的VPC及终端子网。

4. 选择支持IPV6选项并选择IPV6安全组进行配置（安全组可提前创建）。

查看IPV6实例详情

在广州机房实例列表中，点击您创建的具备IPV6地址的实例，进入实例详情。

申请外网IPV6地址

当实例具备IPV6地址时，点击[申请外网IP](#)即可为实例获取外网IPV6地址。

IPV6实例关联安全组

1. 登录[云数据库MySQL控制台](#)。
2. 当您创建IPV6实例时需要为实例关联IPV6安全组，只有在IPV6安全组内的IP方可访问实例。
3. 在创建实例前，将相关IP配置到IPV6安全组中，并在创建实例时进行关联（当前暂不支持创建实例后修改IPV6安全组）。
4. 点击安全组创建IPV6安全组。

5. 创建IPV6安全组后，进入安全组详情配置所需的IP。

6. 当您为实例配置完IP后，创建实例时即可为实例关联该安全组，后续您也可以随时修改安全组IP。

数据库审计支持IPV6

当您的账号具备IPV6权限及审计权限后，您的IPV6实例即可审计到IPV6相关信息（注意IPV6服务当前仅支持广州机房）。

KingSQL 概述

KingSQL 是金山云数据库团队基于MySQL研发的企业级关系型数据库内核，拥有自主知识产权。具有高性能、高可用、易使用、低成本等特点。

当前为内测阶段，可联系客服申请内测。

线程池

在大连接高并发场景下，MySQL性能会出现严重衰减，金山云KingSQL通过线程池功能，大幅提高在高并发场景下的处理效率，保障数据库高效稳定输出。

MySQL默认为一个连接创建一个线程，在大连接高并发场景下，会创建较多的工作线程，导致各线程互相竞争，CPU时间片在大量线程间频繁调度，上下文反复切换，造成较多的CPU资源消耗，严重降低MySQL吞吐。

KingSQL通过线程池，使工作线程和连接不再一一绑定，通过工作线程在连接间的复用，避免创建过多线程，始终将工作线程数控制在最佳线程数附近，从而在不影响中低并发处理效率的同时，大幅提高了高并发场景下的处理能力，同时给予DDL、监控信息查询等操作更高的优先级，确保在业务高负载时能够对数据库进行稳定的监控和管理维护工作。

限制条件

实例版本为KingSQL内核版

如需升级，请联系客服。

适用场景

- 连接多但请求并发低时，通过连接复用，避免创建大量空闲线程，减少系统资源开销。
- 连接多且请求并发高时，限制创建超过数据库服务处理能力的线程，减少线程间上下文切换，从而提高CPU资源利用效率，增加整体吞吐。

参数配置

| 参数 | 含义及配置值 |
|-------------------------|---|
| thread_handling | 配置线程池选项 取值范围： pool-of-threads：开启线程池 one-thread-per-connection：一个连接一个线程 no-threads：一个线程处理所有请求 默认值：one-thread-per-connection |
| threadpool_max_threads | 整个线程池最大线程数，总线程数超过此值时，不会再创建线程（极端场景除外） 取值范围：1 - 10000 默认值：10000 |
| threadpool_idle_timeout | 工作线程空闲的超时时间，超过此时间会自动销毁 取值范围：1 - UINT_MAX 默认值：60（秒） |

性能测试结果

使用sysbench测试，结果如下 `oltp_read_write`、`oltp_write_only`

`oltp_point_select`

异步事务

MySQL组提交（group commit）过程中，大部分工作线程处于阻塞状态，等待相关业务的处理完成，如binlog的生成、binlog的sync刷盘及数据修改为已经提交状态等。当相关业务处理完成，阻塞等待的线程被激活，返回结果。半同步或者全同步场景下，工作线程进入事务提交过程，需要等待主从复制的应答，此时工作线程处于空闲状态，当从库应答后，工作线程被激活，完成事务提交并返回结果。阻塞等待的工作线程不能有效的使用，浪费了工作线程资源，降低了数据库的性能。

KingSQL在线程池模式下，工作线程和提交线程异步化，工作线程在事务提交的过程中不需要阻塞等待。当接收到用户的事务提交请求后，工作线程将事务推入提交等待队列，由后台线程异步处理事务提交逻辑，工作线程无需等待，继续处理新的用户请求。充分释放工作线程，减少工作线程的等待，保证高并发场景下的性能。

限制条件

实例版本为KingSQL内核版，且使用线程池。

适用场景

- 在读写混合场景中，特别是半同步或全同步场景，可以充分提高工作线程的使用率，提升数据库的吞吐性能。
- 需要与线程池配合使用。

参数配置

| 参数 | 含义及配置值 |
|---------------------------------|--------------------------------------|
| <code>async_group_commit</code> | 是否开启异步事务 取值范围： on/off 默认值： off |

锁优化

在mysql5.7.x及以下版本，当出现锁冲突，会同步死锁检测，判断此操作是否会形成死锁。工作线程需要等待死锁检测结果，在锁冲突较高的场景，死锁检测成为性能瓶颈点，降低了数据库的性能。数据读写过程中，锁系统的互斥锁会保护锁（行锁，表锁等）资源。在高并发场景下，线程间互斥锁形成锁冲突，影响数据库的性能。

KingSQL将死锁检测由同步检测替换为异步检测，提升了数据读写性能，在无死锁的情况下无需同步等待，降低数据处理延时。通过把锁系统的互斥锁修改为多个桶锁，降低了多线程间并发读写的锁冲突，提升数据库的并发性能。

适用场景

- 适用于互联网模式下的秒杀场景。
- 适用于热点数据表高并发数据更新的场景。

参数配置

| 参数 | 含义及配置值 |
|-------------------------------------|---|
| <code>innodb_lock_hash_cells</code> | 锁粒度，这个值跟buffer_pool大小相关，如果buffer_pool大于64GB则需要调整该值 取值范围： 1 - 1024 默认值： 64 修改后需重启 |