|  |  |
| --- | --- |
| 客户名称 | 福建朴朴信息技术有限公司（MySQL） |
| 问题描述 | 数据库服务器CPU发出告警，发现有并发高性能差的慢SQL |
| 处理过程 | 连接到数据库服务器执行show full processlist查看当前在跑的SQL进程  C:\Users\ADMINI~1\AppData\Local\Temp\1620435232.jpg  发现workflowactionview视图存在很高的并发，并且查询速度很慢。  我们看下这个视图的逻辑：  视图里面查询workflowactionset的数据，为了实现INTERFACETYPE字段不同值的转换，直接用了5此查询再union    完全可以使用如下的一次查询替换即可：    使用上面的方法，改写视图之后数据库压力释放，CPU回归5%不到  总结：  书写SQL的逻辑应尽量简化，不要想当然的把所有结果拼接到一起，这种操作是很低效不可取的；功能设计初期应避免使用视图，复杂SQL用拆分之后分批执行，而不是全都累积在一起。 |
| 客户名称 | 淄博市大数据局（MySQL） |
| 问题描述 | 正式上线，偶发的发现有SQL执行缓慢，进行整体优化 |
| 处理过程 | 分析slowlog之后发现如下SQL存在严重的性能问题，此处列出两个常见的典型例子：  1. 移动建模的一个问题SQL  ①MySQL里面匹配数据默认不区分大小写，此处增加upper函数进行转换，画蛇添足了，此类操作会导致查询过程不适用索引，引发性能问题  ②很常见的一个错误，formmodeid是int类型，这里匹配确用了char  2.知识文档的一个更新语句  不止是查询的时候可以用join，更新的时候同样也可以，此类更新①的写法会导致imagefileid使用全表扫描，改写为②的写法之后，速度会有很大的改善。    总结：  书写SQL的过程，应遵循SQL的优化建议规则，尽量让数据库用最少的代价得到你想要的结果；字段上加函数进行匹配是很低效的写法，数据库需要对这个字段所有数据进行运算，应避免此类操作。在进行多表关联时我们应该习惯用join的思维，而不是in或者exists的写法。 |
| 客户名称 | 广州仲裁委员会（MySQL） |
| 问题描述 | EM升级之后启动服务异常 |
| 处理过程 | 研发反馈数据库中查询不到表：    远程过程发现数据库中所有的表都丢失了！    此问题已经重复发生多次，原因是客户或者项目在安装好em之后，后续升级KB解压补丁程序时，误操作解压了安装包，导致将所有的程序和数据库覆盖，底层文件丢失，在没有备份的情况下此问题无法修复，只能重新配置EM。    总结：  我们提供的安装包，应该考虑上述的情况，默认文件解压之后的文件名不要与目标文件相同，比如解压之后文件名称为emp\_tmp,再重命名为emp，mv重命名的过程如果文件已存在会提示'是否覆盖'，但解压的过程是不会提示是否覆盖的。 |
| 客户名称 | 昆山农商行（MySQL） |
| 问题描述 | 程序初始化异常 |
| 处理过程 | 最近多个项目反馈初始化过程抛出如下异常：  此问题发生在MySQL5.7的版本中    报错的原因是此脚本重命名字段所使用的语法，在MySQL5.7版本中是不兼容的，  alter table tabname rename column1 to column2；此语法是Oracle中的，MySQL8.0对Oracle做了语法兼容，但是5.7版本中不被支持，导致部分使用5.7版本的客户出现异常，修改后的语法如下：  alter table tabname change column1 column2 text;  总结：  在编写脚本时SQL语句应使用每个数据库标准的SQL语法，尽量不要用偏门的写法；安装包的验证环境应该在多个数据库版本中进行测试验证，比如MySQL5.7/8.0,Oracle 11g/12c等，数据库大版本之间的语法还是有些许差异的。 |
| 客户名称 | 佳兆业集团控股有限公司（MySQL） |
| 问题描述 | 系统不稳定，偶发的缓慢，系统频繁告警 |
| 处理过程 | 架构部同事已协助优化了部分程序和系统硬件方面的功能，协助分析数据库日志时发现如下的操作问题较大：    沟通后得知，此SQL是二开的一个查询流程超时报表的功能，v\_workflow是客户自行编写的一个视图，打开视图的内容：    视图中，关联查询流程的四个常用的大表，又使用了union操作进行数据拼接，还有视图嵌套视图的方法去查询。。。  此视图每次查询需遍历400万条数据，客户月底拉取流程超时报表的操作很多，对系统的整体性能影响较大。  后建议直接与我们流程组同事对接，用更简便的方法查询。  总结：  二开的过程要合理评估需求，针对一些并发大，复杂度高的SQL更需要精简再精简，应尽量使用表之间的关系进行过滤数据，而不是通视图或SQL层层嵌套去拼出来想要的结果。 |
| 客户名称 | 松下家电（中国）有限公司（MySQL） |
| 问题描述 | 测试系统的流程监控一条流程也监控不到，权限配置正常 |
| 处理过程 | 1.开发定位是缺少hrmresourcevirtualview视图所致；  2.实际检测下来，客户环境缺少很多视图和存储过程，检测方法可以大致对比下数据库中的各类对象个数，来校验基本完整性，如下：    3.解决方式：通过从正式库重新备份，再还原到测试库上解决；  4.MySQL的备份还原请务必参考下我们提供的手册：  <https://www.e-cology.com.cn/spa/document/index.jsp?id=3405279>  特别注意：mysqldump命令行备份勿忘加-R参数，且不要用navicat工具备份 |
| 客户名称 | 广州庞大投资集团有限公司（SQLServer） |
| 问题描述 | 新建了一个文档，有完全控制权限，文档也是正常状态，但是就不能编辑 |
| 处理过程 | 1.开发定位ecology日志中有如下报错：    2.根据报错，是报ShareinnerDoc表是无效对象，即不存在这张表，在数据库直接查询也确实如此，然通过如下SQL查询，定位到当前该表所绑定的schema是  db\_owner的架构，不是我们常规默认的dbo架构：  select b.name,a.name From sys.tables a join sys.schemas b on a.schema\_id = b.schema\_id where a.name = 'ShareinnerDoc'    3.进一步通过如下SQL查询，发现有几百张表的所属架构是db\_owner，不是默认的dbo架构：  select b.name,count(1) as cnt from sys.tables a join sys.schemas b on a.schema\_id = b.schema\_id group by b.name  4.综上，需要修复成统一的架构，因为dbo架构占据绝大多数，故把所有db\_owner的架构修改为dbo，方式如下：  USE 实际库名  GO  ALTER USER 实际库名 WITH DEFAULT\_SCHEMA=[dbo]  GO  下面的SQL执行后，复制返回的结果在新建查询窗口再次执行即可完成修改  select 'alter schema dbo transfer db\_owner.'+a.name From sys.tables a join sys.schemas b on a.schema\_id = b.schema\_id where b.name = 'db\_owner'  5.完成以上修复后，再次测试，功能恢复正常。 |
| 客户名称 | 天下秀广告有限公司（MySQL） |
| 问题描述 | 特殊字符(表情)无法存储到mysql |
| 处理过程 | 1.严格来说并不是MySQL无法存储emoji表情符号，只是emoji表情符号需要占用4字节长度，而我们的OA程序都是UTF8编码，其下最大支持字符长度为3字节，所以才会导致报错，要想存储4字节的emoji表情符，需要UTF8MB4编码支持；  2.因为我们的ecology程序，MySQL不便于整库修改成utf8mb4的编码（考虑创建索引时长度问题，utf8mb4限制的长度更短，我们的程序升级时可能会报错、getpinyin函数问题，根据固定字节长度截取翻译的、主外键关系等的影响），故通过修改特定表的特定字段的编码为utf8mb4解决，如下：    3.该方法可作为修改特定需要表情符号输入，而又无法对整个库调整时的修改参考。 |
| 客户名称 | 中科信工程造价咨询（北京）有限责任公司（MySQL） |
| 问题描述 | 数据库恢复环境失败，提示Unknown collation: 'utf8mb4\_0900\_ai\_ci' |
| 处理过程 | 1.项目操作mysql数据库还原时，报错如下错：    2.原因在于客户的正式库环境是mysql8.0.18，测试库环境是mysql5.7.28，而mysql5.7的版本不支持utf8mb4\_0900\_ai\_ci的排序规则；  3.解决方式：通过对客户测试环境的mysql升级到8.0.18，进而彻底解决了还原报错问题，另如果不升级mysql版本，也可以直接替换备份文件后再还原解决，如下： sed -i "s/utf8mb4\_0900\_ai\_ci/utf8mb4\_general\_ci/g" 20210308.sql  4.强烈建议客户测试和正式环境数据库要保持版本的统一性，一方环境做了升级后，另一方环境也要相应升级。 |
| 客户名称 | 深圳华大北斗科技有限公司（MySQL） |
| 问题描述 | 华大北斗后台客户模块添加自定义字段报错 |
| 处理过程 | 1.项目反馈客户的后台客户模块添加自定义字段时，报如下错误：    2.原因在于对crm\_customerinfo表添加字段时，该表上的单行字节总长度超出了65535字节的限制所致；  3.通过SHOW CREATE TABLE crm\_customerinf;看到有dh,khdm,lxr,xsqd,yfszd,yx,zw七个为varchar(999)的字段类型，和项目商议后，实际并不需要那么长，故yfszd缩减为200，其它字段都缩减到100的长度；  3.在对crm\_customerinf表中实际字段类型修改后，也要对CRM\_CustomerDefinField表中usetable='CRM\_CustomerInfo'的对应字段类型也修改掉；  4.上面步骤操作完以后，再次添加字段，恢复正常；  5.特别说明：  MySQL定义的单行存储最大长度是65535字节，UTF8编码下最多存21845字符，表单行总长度不能超出21845字符，需要特别注意的是：如在Oracle中定义varchar(300)能够存储100个汉字，在MySQL中只需设置varchar(100)即可达到同样的效果，所以日常使用中，一定要合理规划字段的长度，避免不必要的字段长度开销。 |