MySQL的日志管理

**错误日志**：记录mysql服务的启动、运行、停止mysql服务时出现的问题

**查询日志**：记录建立的客户端连接和执行的语句

**二进制日志**：记录所有更改数据的语句，可以用于数据复制。

**慢查询日志**：记录所有执行时间超过long\_query\_time的所有查询或不使用索引的查询

**日志表**：启用了普通日志和慢查询日志时候确定输出目标。在mysql中的配置选项为log\_output(FILE(默认),TABLE,NONE)

注意：默认情况下，所有日志创建于mysql数据目录中。通过刷新日志，可以强制mysql关闭和重新打开日志文件（或者在某些情况下切换到一个新的日志）。当执行一个FLUSH LOGS语句或执行mysqladmin flush-logs 或mysqladmin refresh 时，将刷新日志。如果使用mysql复制功能，在复制服务器上可以维护更多日志文件，这种日志称为接替日志。其他日志功能会降低mysql数据库的性能。例如，在查询非常频繁的mysql数据库系统中，如果开启了通用查询日志和慢查询日志，mysql数据库会花费很多时间记录日志。同时，日志会占用大量的磁盘空间。

# 出错日志

错误日志文件包含了当mysqld启动和停止时，以及服务器在运行过程中发生任何严重错误时的相关信息。在MYSQL中，错误日志也是非常重要的，mysql将启动和停止数据库信息以及一些错误信息记录到错误日志中。在默认情况下，错误日志会记录到数据库的数据目录下。如果没有在配置文件中指定文件名，则文件名默认为hostname.err，log-error可以配置指定路径。

|  |
| --- |
| **#查看错误日志**  show variables LIKE 'log\_error';#找到日志文件位置  **#删除错误日志（手动），并重建日志文件**  mysqladmin -u root -p flush-logs 或者 flush logs; |

# 普通查询日志

普通查询日志记录了mysql的所有用户操作，包括启动和关闭服务、执行查询和更新语句等。mysql服务器默认情况下并没有开启通用查询日志。如果需要通用查询日志，可以通过修改my.cnf配置文件来开启。在my.cnf的[mysqld]组下加入选项：

|  |
| --- |
| general\_log=1  general\_log\_file=普通查询日志位置 |

# 慢查询日志

慢查询日志是记录查询时长超过指定时间的日志。慢查询日志主要用来记录执行时间较长的查询语句。通过慢查询日志，可以找出执行时间较长、执行效率较低的语句，然后进行优化。mysql中慢查询日志默认是关闭的，可以在mysqld中添加如下配置：

|  |
| --- |
| slow\_query\_log = 1  long\_query\_time =1  slow\_query\_log\_file = 慢日志位置 |

# 二进制日志

mysql创建二进制日志文件时，首先创建一个以filename为名称，以index为后缀的文件；再创建一个以filename为名称，以“.000001”为后缀的文件。当mysql服务重新启动一次，以“.000001”为后缀的文件会增加一个，并且后缀名加1递增；如果日志长度超过了max\_binlog\_size的上限（默认是1GB）也会创建一个新的日志文件。

## 查看二进制日志

show binary logs语句：可以查看当前二进制日志文件个数和文件名。

mysqlbinlog bin-log文件：查看二进制文件内容

## 删除二进制日志

删除所有的二进制日志文件使用：RESET MASTER;

输出部分二进制日志：

PURGE MASTER LOGS TO 'log\_name'

PURGE MASTER LOGS BEFORE 'date'

|  |
| --- |
| purge master logs before "20160414";  PURGE MASTER LOGS TO 'mysql-bin.000006'; |

## 使用二进制日志还原数据库

mysqlbinlog恢复数据的语法如下：

mysqlbinlog [option] filename | mysql -uuser -ppass

option是一些可选项，filename是日志文件名

比较重要的两对option参数是：

* --start-datetime：恢复数据库的起始时间点
* --stop-datetime：恢复数据库的结束时间点
* --start-position：恢复数据的开始位置
* --stop–position：恢复数据的结束位置

## 暂时停止二进制日志

SET sql\_log\_bin={0|1}

# 日志管理

### 自动完成日志过期处理

#手动修改mysql-log-rotate

cp /opt/mysql/support-files/mysql-log-rotate /etc/logrotate.d/

/usr/sbin/logrotate -f /etc/logrotate.d/mysql-log-rotate

### 二进制日志文件和中继日志文件的过期处理

二进制日志处理：从从服务器中找到最小binlog位置，然后执行

purge master logs to ' ####.bin'

中继日志文件：在处理完后会自动删除