



南开大学
Nankai University

慢查询日志

数据库系统上机

计算机学院&网络空间安全学院 乜鹏

<https://dbis.nankai.edu.cn/2019/0417/c12139a128118/page.htm>



声明：上机课程的内容偏向举例，通俗化，一些术语的准确定义请查看理论课程。

投票 最多可选4项



MySQL都有哪些种日志?

- A 查询日志
- B 慢查询日志
- C 快查询日志
- D 二进制文件

允公允能 日新月异

NANKAI UNIVERSITY



MySQL的慢查询日志是MySQL提供的一种日志记录，它用来记录在MySQL中响应时间超过阈值的语句，会记录哪些操作呢？

- A select
- B update
- C delete
- D call



什么是慢查询日志?



南开大学
Nankai University

- 不要被"慢查询日志"的名字误导, 错误的以为慢查询日志只会记录执行比较慢的SELECT语句, 其实不然, INSERT、DELETE、UPDATE、CALL等DML操作, 只要是超过了指定的时间, 都可以称之为"慢查询", 被记录在慢查询日志中。
- 默认情况下, 慢查询日志是不被开启的, 如果需要, 可以手动开启, 开启慢查询日志之后, 默认设置下, 执行时间超过10秒的语句才会被记录到慢查询日志中, 当然, 对于"慢"的定义, 见仁见智, 超过多长时间才是"慢", 可以自定义。
- 除了慢查询日志之外, MySQL中的日志还包括错误日志, 查询日志、二进制日志, 中继日志和innodb存储引擎级别的事务日志



慢查询日志相关参数



南开大学
Nankai University

- `log-slow-queries`: 表示是否开启慢查询日志, 5.6以前的版本使用此参数指定是否开启慢查询日志, 5.6以后的版本使用`slow_query_log`取代此参数, 如果你使用的mysql版本刚好是5.5, 那么可以看到这两个参数同时存在, 此时不需要同时设置它们, 设置这两个参数中的任何一个, 另一个也会自动保持一致。
- `log_output`: 日志存储方式, 可以设置为以下四种:
 - `log_output='FILE'`, 表示将日志存入文件, 默认值是'FILE'。
 - `log_output='TABLE'`, 表示将日志存入数据库, 这样日志信息就会被写入到mysql.slow_log表中。
 - `log_output='FILE, TABLE'`, 表示支持同时两种存储方式, 一般不会这样设置, 因为会徒增许多IO压力
 - `log_output='NONE'`, 表示不记录查询日志
- `long_query_time`: 慢查询阈值, 表示“多长时间的查询”被认定为“慢查询”, 此值得默认值为10秒, 表示超过10秒的查询被认定为慢查询, 需要记录日志。



慢查询日志相关参数



南开大学
Nankai University

- `slow-query-log-file`: 当使用文件存储慢查询日志时 (`log_output` 设置为 "FILE" 或者 "FILE, TABLE" 时), 指定慢查询日志存储于哪个日志文件中, 默认的慢查询日志文件名为 "主机名-slow.log", 慢查询日志的位置为 `datadir` 参数所对应的目录位置, 一般情况下为 `/var/lib/mysql`
- `log_queries_not_using_indexes`: 表示如果运行的sql语句没有使用到索引, 是否也被当做慢查询语句记录到慢查询日志中, OFF表示不记录, ON表示记录。
- `log_throttle_queries_not_using_indexes`: 5.6.5版本新引入的参数, 当`log_queries_not_using_indexes`设置为ON时, 没有使用索引的查询语句也会被当做慢查询语句记录到慢查询日志中, 使用`log_throttle_queries_not_using_indexes`可以限制这种语句每分钟记录到慢查询日志中的次数, 因为在生产环境中, 有可能有很多没有使用索引的语句, 此类语句频繁的被记录到慢查询日志中, 可能会导致慢查询日志快速不断的增长, 管理员可以通过此参数进行控制。



- 在当前数据库中开启
 - 在MySQL中输入命令查看状态: `show variables like '%slow_%';`

```
log_slow_admin_statements | OFF
log_slow_slave_statements | OFF
slow_launch_time          | 2
slow_query_log             | OFF
slow_query_log_file        | /var/lib/mysql/iz2ze2kju3zc4dilp2k9j7z-slow.log
```

- 若slow_query_log为OFF, 继续输入命令: `set global slow_query_log=1;`
- 再次查看状态, 变为ON

```
mysql> show variables where variable_name like "%slow_query_log%";
+-----+-----+
| Variable_name | Value |
+-----+-----+
| slow_query_log | ON    |
| slow_query_log_file | /var/lib/mysql/iz2ze2kju3zc4dilp2k9j7z-slow.log |
+-----+-----+
```

- 注意: 使用上述命令开启的慢查询日志只对当前数据库生效, MySQL重启后就会失效, 如果要永久生效需要修改my.cnf文件, (添加状态slow_query_log =1后, 重启MySQL服务)



- 设置慢查询日志存储方式

- 在MySQL中输入命令设置: set global log_output='TABLE';

```
mysql> mysql> set global log_output='TABLE';
Query OK, 0 rows affected (0.00 sec)

mysql> show variables where variable_name like "%log_output%";
+-----+-----+
| Variable_name | Value |
+-----+-----+
| log_output    | TABLE |
+-----+-----+
```

- 设置慢查询阈值

- 在MYSQl中输入命令设置阈值为3秒: set global long_query_time=3;

```
mysql> show variables where variable_name like "%long_query_time%";
+-----+-----+
| Variable_name | Value |
+-----+-----+
| long_query_time | 10.000000 |
+-----+-----+
```

- 注意: 使用上述命令直接修改之后进行查询发现变量值没有发生改变, 是因为在当前会话中并没有取出修改之后的变量值, 只需另起一个会话即可看到修改之后的值



- 故意使语句的执行时间超过3秒，查看日志记录
 - 在MySQL中输入命令：select sleep(5);
 - 查看慢查询日志记录

```
mysql> select sleep(5);
+-----+
| sleep(5) |
+-----+
|          0 |
+-----+
1 row in set (5.00 sec)

mysql> select * from mysql.slow_log
-> ;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| start_time | rows_examined | db | last_insert_id | user_host | server_id | sql_text | query_time | thread_id | lock_time |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 2020-04-29 20:39:38.455446 | 0 |  | root[root] @ localhost [] | 0 | 00:00:05.000687 | 00:00:00.000000 | 1527 | select sleep(5) |
```



慢查询日志使用小结



南开大学
Nankai University

- 什么样的SQL才会记录到慢查询日志中?
 - 由参数long_query_time控制，默认为10s，查询时间超过该值的会被记录
 - 可以通过命令修改：set global long_query_time=4;
 - 也可以在my.cnf中修改，修改方式同上
 - 如果修改了变量log_queries_not_using_indexes，那么未使用索引的查询也会被记录到慢查询日志中，在性能调优的时候可以开启这个选项
- 如何查询有多少慢查询记录?
 - 可以使用系统变量Slow_queries
 - 命令：show global status like '%slow_queries%';



mysqldumpslow命令



南开大学
Nankai University

- mysql自带了对慢查询日志的统计分析工具：**mysqldumpslow**，通过mysqldumpslow命令我们可以更加方便的从不同的维度对慢日志进行排序、查找、统计。
- 但是mysqldumpslow只能作用于慢查询日志文件，而不会统计slow_log表，换句话说，如果我们将log_output的值设置为“TABLE”，mysqldumpslow则无用武之地了，只有当log_output的值设置为“FILE”或者“FILE, TABLE”时，才可以通过mysqldumpslow查看对应的慢查询日志文件



mysqldumpslow命令基本用法



南开大学
Nankai University

- 首先将慢查询日志的存储方式改为'FILE'，并设置存储文件

```
mysql> show global variables where variable_name like '%slow_query_log_file%';
```

Variable_name	Value
slow_query_log_file	/var/lib/mysql/myhost-slow.log

- 使用mysqldumpslow命令直接指定对应的慢查询日志文件即可：
 - 在终端输入命令：mysqldumpslow -s t /var/lib/mysql/myhost-slow.log

```
[root@iz2ze2kju3zc4dilp2k9j7z jupyter]# mysqldumpslow -s t /var/lib/mysql/myhost-slow.log
```

```
Reading mysql slow query log from /var/lib/mysql/myhost-slow.log
Count: 3  Time=5.00s (15s)  Lock=0.00s (0s)  Rows=1.0 (3), root[root]@localhost
      select  sleep(N)
```

- 从上图分析可以看出：select sleep(N)这条语句一共执行了3次，平均时间5秒一次，合计时间15秒，返回记录共3条。



- 通过上面例子可知，-s选项代表排序，根据什么排序取决于-s选项后面添加的参数，下面是各个参数含义：
 - c: 执行计数
 - l: 锁定时间
 - r: 返回记录
 - t: 执行时间
 - al: 平均锁定时间
 - ar: 平均返回记录数
 - at: 平均执行时间