

MySQL InnoDB MONITOR 性能监控

Cuihulong MySQL从1开始 2025年07月18日 11:38 上海

MySQL中碰到问题，排除故障必不可少解读InnoDB内部状态（SHOW ENGINE INNODB STATUS），InnoDB监视器提供的信息：比如事务，死锁，信号量，io，自适应hash，缓存等现状。

一般DBA采取方式是通过SQL命令行在需要时获得标准InnoDB Monitor输出到客户端程序中。

因为目前提供的方式是innodb自动循环捕获信息并覆盖原有的信息。所以往往事故过后，无法获取当时现状信息。除此之外，InnoDB内部状态对于性能调优非常有用。

其实在INNODB也提供日志方式记录，当InnoDB监视器开启时，InnoDB大约每15秒将输出到错误日志或指定日志文件。

1.启动跟踪

- 日志输出方面比较单一。innodb_status_output和innodb_status_output_locks变量用于开启标准InnoDB Monitor和InnoDB Lock Monitor。

```
mysql> SHOWVARIABLESLIKE'%innodb_status%';

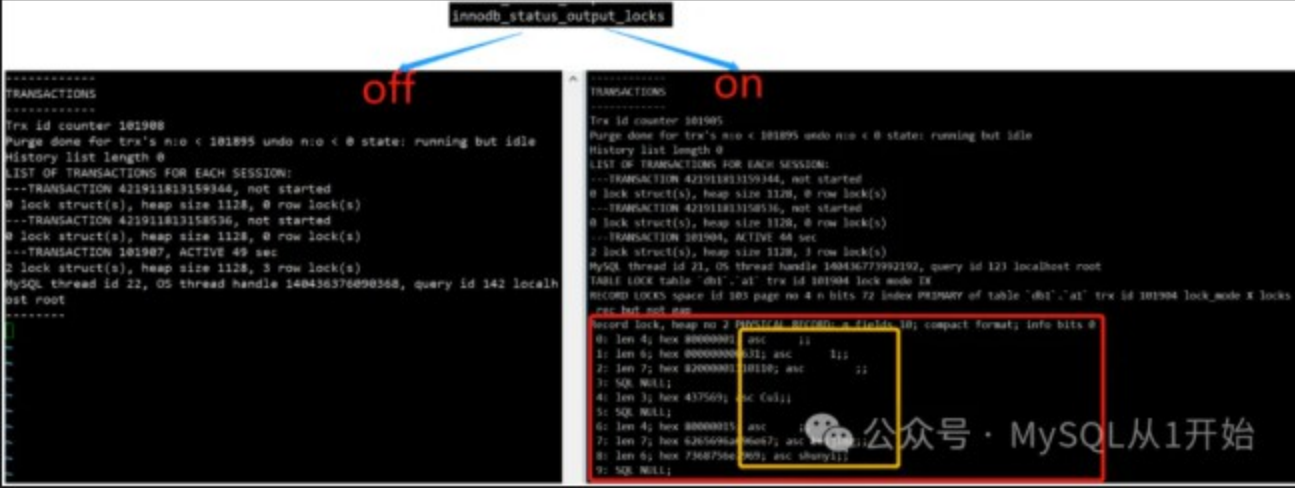
+-----+-----+
| Variable_name | Value |
+-----+-----+
| innodb_status_output | ON |
| innodb_status_output_locks | OFF |
+-----+-----+

2 rows in set (0.00 sec)

#备注：开启/关闭InnoDB监视器需要PROCESS权限

mysql>SETGLOBAL innodb_status_output=ON;
mysql>SETGLOBAL innodb_status_output=ON;
```

- Lock monitor差异
- 如果后用Lock monitor，就会打开单个输出流，流包括额外的锁信息。
- 做个对比图：提供了额外操作的数据信息，可以准确的定位到对应的数据操作



- 文件输出
- 在不开启innodb_status_output参数的情况，通过启动时指定innodb-status-file选项，可以后用标准InnoDB Monitor输出并将其指向一个状态文件。当使用此选项时，datadir下生成一个InnoDB会创建一个名为innodb_status的文件Pid，并大约每15秒向其写入输出。
- 注意：正常关闭数据库自动删除，异常关闭就不会进行删除。

非动态参数, 可以再my.cnf中添加innodb_status_file=1启用

```
[mysqld]
innodb_status_file=1

#或 启动方式
mysqld --defaults-file=/etc/my8.0.cnf --innodb-status-file=on --user=mysql
```

```
[root@schouse mysql]#  
[root@schouse mysql]# ll innodb_stat*  
-rw-r----- 1 mysql mysql 4300 Oct 26 11:28 innodb_status 78250
```

- 自定义脚本
作为15s周期输出监控的替代方案,间隔有点太短，容易对性能有影响，可以通过SHOW ENGINE INNODB STATUS 这一SQL语句来获得InnoDB的标准监控输出。通过自定义脚本和crontab进行抽取。

```
#!/bin/bash  
#####  
# crontab #  
# */10 * * * * sh /root/innodb_statu.sh #  
# sed -i 's/\r//g' awr.sh #  
#####  
source /etc/profile  
HOSTNAME="127.0.0.1"  
PORT="3380"  
USERNAME="root"  
PASSWORD="123456"  
SOCK="/opt/data8.0/mysql/mysql.sock"  
  
mysql_innobdstatus(){  
    AWR_PATH="/opt/data8.0/logs/"$1  
    if [ ! -d $AWR_PATH ] ;then  
        mkdir -p $AWR_PATH  
    fi  
  
    FINENAME=$AWR_PATH/$(date +%mysql_awr_%Y-%m-%d).log  
    if [ ! -f $FINENAME ] ;then  
        touch -p $FINENAME  
    fi  
  
    now=`date +%Y-%m-%d %H:%M:%S`  
  
    echo "++++++"  
+++>> $FINENAME  
    echo "innodb engine">> $FINENAME  
    echo "++++++"  
+++>> $FINENAME  
    #mysql -h$HOSTNAME-P$PORT -u$USERNAME-p$PASSWORD -e"show engine  
innodb status\G" >> $FINENAME  
    mysql --user=$USERNAME --password=$PASSWORD --socket=$SOCK -A  
-e"show engine innodb status\G" >> $FINENAME  
  
    #delete file 10 day age  
    find $AWR_PATH -type f -mtime +10 | xargs rm -f  
}  
  
mysql_innobdstatus 3380
```

2.指标说明

Innobd监控目前主要关注点是信号量，transation，死锁， 外键等问题。之前写的一片文章中有些指标介绍，可以参考下。
<https://www.modb.pro/db/80471>

3.InnoDB Monitor会自动打开：

InnoDB在以下情况下暂时启用标准InnoDB Monitor输出:

- A long semaphore wait信号量等待
- InnoDB无法在缓冲池中找到空闲的块
- 超过67%的缓冲池被锁堆或自适应哈希索引占用

4.总结

InnoDB监视器要是只在真正需要的时候启用，那可能关键信息已经被刷新掉。所以对mysql负载不高的情况下可以开启。目前推荐方式采用脚本。

- InnoDB监视输出会导致性能下降：经过sysbench多次测试普遍情况下性能损耗大概0.2%~3%。目前不存在致命的穷住数据库问题，但还需要谨慎。
- 日志刷新是每15秒周期性输出一次，但是由于状态收集与输出也会占用一些时间。因此，两次日志时间并不是规律的间隔15秒。
- 如监视器输出指向错误日志，错误日志会变的非常大。需要定期FLUSH ERROR LOGS命令行进行切割
- 定向指定日志innodb-status-file方式也会存在大文件问题。还有个人觉得频率确实太高。如提供参数控制频率，不删除就应该更友好。

