

# MySQL压测常用方法

## 一、mysql自带的mysqlslap工具

通常，我们会出于以下几个目的对MySQL进行压力测试：

- 1、确认新的MySQL版本性能相比之前差异多大，比如从5.6变成5.7，或者从官方版本改成Percona分支版本；
- 2、确认新的服务器性能是否更高，能高多少，比如CPU升级了、阵列卡cache加大了、从机械盘换成SSD盘了；
- 3、确认一些新的参数调整后，对性能影响多少，比如 innodb\_flush\_log\_at\_trx\_commit、sync\_binlog 等参数；
- 4、确认即将上线的新业务对MySQL负载影响多少，是否能承载得住，是否需要服务器进行扩容或升级配置；

针对第四种，需要和线上业务结合起来，这时候就需要自行开发测试工具，或者利用 [tcpcopy](#) 将线上实际用户请求导向测试环境，进行仿真模拟测试。

对于前三种，我们通常采用基准测试就可以。比较常用的MySQL基准压力测试工具有 [tpcc-mysql](#)、[sysbench](#)、[mysqlslap](#) 等几个。

--concurrency代表并发数量，多个可以用逗号隔开，concurrency=10, 50, 100，并发连接线程数分别是10、50、100个并发。

--engine代表要测试的引擎，可以有多个，用分隔符隔开。

--iterations代表要运行这些测试多少次。

--auto-generate-sql 代表用系统自己生成的SQL脚本来测试。

--auto-generate-sql-load-type 代表要测试的是读还是写还是两者混合的（read,write,update,mixed）

--number-of-queries 代表总共要运行多少次查询。每个客户运行的查询数量可以用查询总数/并发数来计算。

--debug-info 代表要额外输出CPU以及内存的相关信息。

--number-int-cols：创建测试表的 int 型字段数量

--auto-generate-sql-add-autoincrement：代表对生成的表自动添加auto\_increment列，从5.1.18版本开始

--number-char-cols 创建测试表的 char 型字段数量。

--create-schema 测试的schema，MySQL中schema也就是database。

--query 使用自定义脚本执行测试，例如可以调用自定义的一个存储过程或者sql语句来执行测试。

--only-print 如果只想打印看看SQL语句是什么，可以用这个选项。

```
mysqlslap -umysql -p123 --concurrency=100 --iterations=1 --auto-generate-sql --auto-generate-sql-add-autoincrement --auto-generate-sql-load-type=mixed --engine=myisam --number-of-queries=10 --debug-info
```

或：

指定数据库和sql语句：

```
mysqlslap -h192.168.3.18 -P4040 --concurrency=100 --iterations=1 --create-schema='test' --query='select * from test;' --number-of-queries=10 --debug-info -umysql -p123
```

要是看到底做了什么可以加上：--only-print

Benchmark

Average number of seconds to run all queries: 25.225 seconds

Minimum number of seconds to run all queries: 25.225 seconds

Maximum number of seconds to run all queries: 25.225 seconds

Number of clients running queries: 100

Average number of queries per client: 0

以上表明100个客户端同时运行要25秒

再如：

```
mysqlslap -uroot -p123456 --concurrency=100 --iterations=1 --engine=myisam --create-schema='haodingdan112' --query='select * From order_boxing_transit where id = 10' --number-of-queries=1 --debug-info
```

## 二、关于压测的其他建议：

### 1、如何避免压测时受到缓存的影响

【老叶建议】有2点建议

a、填充测试数据比物理内存还要大，至少超过 [innodb buffer pool size](#) 值，不能将数据全部装载到内存中，除非你的本意就想测试全内存状态下的MySQL性能。

b、每轮测试完成后，都重启mysqld实例，并且用下面的方法删除系统cache，释放swap（如果用到了swap的话），甚至可以重启整个OS。

```
[root@imysql.com]# sync -- 将脏数据刷新到磁盘
```

```
[root@imysql.com]# echo 3 > /proc/sys/vm/drop_caches -- 清除OS Cache
```

```
[root@imysql.com]# swapoff -a && swapon -a
```

### 2、如何尽可能体现线上业务真实特点

【老叶建议】有2点建议

a、其实上面已经说过了，就是自行开发测试工具或者利用 [tcpcopy](#)（或类似交换机的mirror功能）将线上实际用户请求导向测试环境，进行仿真模拟测试。

b、利用 [http\\_load](#) 或 [siege](#) 工具模拟真实的用户请求URL进行压力测试，这方面我不是太专业，可以请教企业内部的压力测试同事。

### 3、压测结果如何解读

【老叶建议】压测结果除了tps/TpmC指标外，还应该关注压测期间的系统负载数据，尤其是 **iops**、**iowait**、**svctm**、**%util**、**每秒I/O字节数**（I/O吞吐）、**事务响应时间**（tpcc-mysql/sysbench 打印的测试记录中均有）。另外，如果I/O设备能提供设备级 **IOPS**、**读写延时** 数据的话，也应该一并关注。

### 4、如何加快tpcc\_load加载数据的效率

【老叶建议】tpcc\_load其实是可以并行加载的，一方面是可以区分 **ITEMS**、**WAREHOUSE**、**CUSTOMER**、**ORDERS** 四个维度的数据并行加载。另外，比如最终想加载1000个 warehouse的话，也可以分开成1000个并发并行加载的。看下 tpcc\_load 工具的参数就知道了：

参考：<http://imysql.cn/tag/%E5%8E%8B%E6%B5%8B>  
<http://blog.csdn.net/mablevi/article/details/5881491>

注：

```
[root@localhost ~]# mysqlslap --help
mysqlslap Ver 1.0 Distrib 5.7.11-4, for Linux (x86_64)
Copyright (c) 2009-2016 Percona LLC and/or its affiliates
Copyright (c) 2005, 2016, Oracle and/or its affiliates. All rights reserved.
```

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

Run a query multiple times against the server.

Usage: mysqlslap [OPTIONS]

Default options are read from the following files in the given order:

/etc/my.cnf /etc/mysql/my.cnf /usr/local/Percona-Server-5.7.11-4-Linux.x86\_64.ssl101/etc/my.cnf ~/.my.cnf

The following groups are read: mysqlslap client

The following options may be given as the first argument:

```
--print-defaults      Print the program argument list and exit.
--no-defaults         Don't read default options from any option file,
                        except for login file.
--defaults-file=#      Only read default options from the given file #.
--defaults-extra-file=# Read this file after the global files are read.
--defaults-group-suffix=#
                        Also read groups with concat(group, suffix)
--login-path=#        Read this path from the login file.
-?, --help            Display this help and exit.
-a, --auto-generate-sql
                        Generate SQL where not supplied by file or command line.
--auto-generate-sql-add-autoincrement
                        Add an AUTO_INCREMENT column to auto-generated tables.
--auto-generate-sql-execute-number=#
                        Set this number to generate a set number of queries to
                        run.
--auto-generate-sql-guid-primary
                        Add GUID based primary keys to auto-generated tables.
--auto-generate-sql-load-type=name
                        Specify test load type: mixed, update, write, key, or
                        read; default is mixed.
--auto-generate-sql-secondary-indexes=#
```

Number of secondary indexes to add to auto-generated tables.

--auto-generate-sql-unique-query-number=#  
Number of unique queries to generate for automatic tests.

--auto-generate-sql-unique-write-number=#  
Number of unique queries to generate for auto-generate-sql-write-number.

--auto-generate-sql-write-number=#  
Number of row inserts to perform for each thread (default is 100).

--commit=# Commit records every X number of statements.

-C, --compress Use compression in server/client protocol.

-c, --concurrency=name  
Number of clients to simulate for query to run.

--create=name File or string to use create tables.

--create-schema=name  
Schema to run tests in.

--csv[=name] Generate CSV output to named file or to stdout if no file is named.

-#, --debug[=#] This is a non-debug version. Catch this and exit.

--debug-check This is a non-debug version. Catch this and exit.

-T, --debug-info This is a non-debug version. Catch this and exit.

--default-auth=name Default authentication client-side plugin to use.

-F, --delimiter=name  
Delimiter to use in SQL statements supplied in file or command line.

--detach=# Detach (close and reopen) connections after X number of requests.

--enable-clear-text-plugin  
Enable/disable the clear text authentication plugin.

-e, --engine=name Storage engine to use for creating the table.

-h, --host=name Connect to host.

-i, --iterations=# Number of times to run the tests.

--no-drop Do not drop the schema after the test.

-x, --number-char-cols=name  
Number of VARCHAR columns to create in table if specifying --auto-generate-sql.

-y, --number-int-cols=name  
Number of INT columns to create in table if specifying --auto-generate-sql.

--number-of-queries=#  
Limit each client to this number of queries (this is not exact).

--only-print Do not connect to the databases, but instead print out what would have been done.

-p, --password[=name]  
Password to use when connecting to server. If password is not given it's asked from the tty.

--plugin-dir=name Directory for client-side plugins.

-P, --port=# Port number to use for connection.

--post-query=name Query to run or file containing query to execute after tests have completed.

--post-system=name system() string to execute after tests have completed.

--pre-query=name Query to run or file containing query to execute before running tests.

--pre-system=name system() string to execute before running tests.

--protocol=name The protocol to use for connection (tcp, socket, pipe, memory).

```

-q, --query=name      Query to run or file containing query to run.
--secure-auth         Refuse client connecting to server if it uses old
                      (pre-4.1.1) protocol. Deprecated. Always TRUE
-s, --silent          Run program in silent mode - no output.
-S, --socket=name     The socket file to use for connection.
--sql-mode=name       Specify sql-mode to run mysqlslap tool.
--ssl-mode=name       SSL connection mode.
--ssl                 Deprecated. Use --ssl-mode instead.
                      (Defaults to on; use --skip-ssl to disable.)
--ssl-verify-server-cert
                      Deprecated. Use --ssl-mode=VERIFY_IDENTITY instead.
--ssl-ca=name         CA file in PEM format.
--ssl-capath=name     CA directory.
--ssl-cert=name       X509 cert in PEM format.
--ssl-cipher=name     SSL cipher to use.
--ssl-key=name        X509 key in PEM format.
--ssl-crl=name        Certificate revocation list.
--ssl-crlpath=name    Certificate revocation list path.
--tls-version=name    TLS version to use, permitted values are: TLSv1, TLSv1.1,
                      TLSv1.2
-u, --user=name       User for login if not current user.
-v, --verbose         More verbose output; you can use this multiple times to
                      get even more verbose output.
-V, --version         Output version information and exit.

```

[root@localhost ~]#