MySQL5.7的安装和测试

# 安装配置

|  |
| --- |
| #mysql安装  wget http://cdn.mysql.com/Downloads/MySQL-5.7/mysql-boost-5.7.13.tar.gz -P /usr/local/src  cd /usr/local/src  tar -zxvf mysql-boost-5.7.13.tar.gz  cmake . -DCMAKE\_INSTALL\_PREFIX=/opt/mysql -DMYSQL\_DATADIR=/data/mysql -DSYSCONFDIR=/opt/mysql -DDEFAULT\_CHARSET=utf8 -DDEFAULT\_COLLATION=utf8\_general\_ci -DEXTRA\_CHARSETS=all -DENABLED\_LOCAL\_INFILE=1 -DDOWNLOAD\_BOOST=1 -DWITH\_BOOST=./boost  make &&make install  cat > /opt/mysql/my.cnf <<EOF  [mysqld]  #GENERAL  basedir =/opt/mysql  datadir=/data/mysql/data  socket=/data/mysql/data/mysql.sock  pid\_file=/data/mysql/data/mysql.pid  user=mysql  port=3306  #INNODB  innodb\_buffer\_pool\_size=6G  innodb\_buffer\_pool\_instances=4  innodb\_log\_buffer\_size=16M  innodb\_log\_file\_size =256M  innodb\_log\_files\_in\_group=8  innodb\_file\_per\_table=1  innodb\_data\_file\_path = ibdata1:512M:autoextend  innodb\_undo\_tablespaces = 3  innodb\_undo\_log\_truncate = 1  #MyISAM  key\_buffer\_size=32M  #单独键缓冲：my\_cache. key\_buffer\_size=32M  #运行时设置：SET GLOBAL my\_cache. key\_buffer\_size=32M;  #创建表后分配缓存：cache index table\_name in my\_cache  #预先加载表索引到给定的缓存：LOAD INDEX INTO table\_name  #LOGGING  log\_error = /data/mysql/log/mysql-alert.log  slow\_query\_log = 1  slow\_query\_log\_file = /data/mysql/log/slow.log  long\_query\_time =0.5  #Replication配置  server-id=1  log-bin=/data/mysql/log/bin-log  relay\_log=/data/mysql/log/relay-log  replicate\_wild\_ignore\_table=mysql.%  replicate\_wild\_ignore\_table=performance\_schema.%  replicate\_wild\_ignore\_table=information\_schema.%  replicate\_wild\_ignore\_table=test.%  gtid\_mode = ON  enforce-gtid-consistency=true  log-slave-updates=true  #OTHER  tmp\_table\_size=64M  max\_heap\_table\_size=64M  query\_cache\_type=0  query\_cache\_size=0  max\_connections=500  thread\_cache\_size=150  table\_open\_cache = 5000  table\_definition\_cache=5000  open\_files\_limit=65535  skip-name-resolve  lower\_case\_table\_names=1  max\_allowed\_packet=16M  explicit\_defaults\_for\_timestamp=true  character-set-server=utf8mb4  collation-server=utf8mb4\_general\_ci  sql\_mode=NO\_ENGINE\_SUBSTITUTION,STRICT\_TRANS\_TABLES  [client]  socket=/data/mysql/data/mysql.sock  port=3306  user=root  password=123  [mysql]  default-character-set=utf8mb4  [mysqldump]  quick  max\_allowed\_packet = 16M  user=root  password=123    EOF  /bin/cp /opt/mysql/my.cnf /etc/my.cnf  #只创建root@localhost用户，密码为空  /opt/mysql/bin/mysqld --initialize-insecure --user=mysql --basedir=/opt/mysql --datadir=/data/mysql/data  #设置mysql开机自动启动  cat >> /etc/profile <<EOF  export PATH=\$PATH:/opt/mysql/bin  EOF  source /etc/profile  cp /opt/mysql/support-files/mysql.server /etc/rc.d/init.d/mysqld  chmod 755 /etc/init.d/mysqld #增加执行权限  chkconfig mysqld on #加入开机启动  service mysqld start #启动mysql  mysql -uroot -e "GRANT ALL ON \*.\* to root@'%' IDENTIFIED BY '123' WITH GRANT OPTION"  mysql -uroot -e "set password for 'root'@'localhost'=PASSWORD('123')" |

# mysql基准测试

## sysbench测试

安装：yum install sysbench -y

yum install perl-DBD-MySQL –y

|  |
| --- |
| **cpu性能测试**  #测试打印20000个素数的时间，时间越小，性能越好40s.  sysbench --test=cpu --cpu-max-prime=20000 run |
| **内存测试**  sysbench --test=memory --num-threads=16 --memory-block-size=8192 --memory-total-size=10G run |
| **IO测试**  #!/bin/bash  set -u  set -x  set -e  for size in 2G 8G;do  for mode in seqrd seqrw rndrd rndwr rndrw;do  for blksize in 4096 16384;do  sysbench --test=fileio --file-num=64 --file-total-size=$size prepare  for threads in 1 4 8 16 32;do  echo "====== testing $blksize in $threads threads"  echo PARAMS $size $mode $threads $blksize > sysbench-size-$size-mode-$mode-threads-$threads-blksz-$blksize  for i in 1 2 3;do  sysbench --test=fileio --file-total-size=$size --file-test-mode=$mode --max-time=180 --max-requests=100000000\  --num-threads=$threads --init-rng=on --file-num=64 --file-extra-flags=direct --file-fsync-freq=0\  --file-block-size=$blksize run | tee -a sysbench-size-$size-mode-$mode-threads-$threads-blksz-$blksize 2>&1  done  done  sysbench --test=fileio --file-total-size=$size cleanup  done  done  done |
| **OLTP测试**  **--准备**  sysbench --test=oltp --mysql-table-engine=innodb --mysql-user=root --mysql-password=123 --db-driver=mysql --mysql-db=test --oltp-table-size=10000000 --oltp-table-name=t1 --oltp-nontrx-mode=insert --mysql-socket=/data/mysql/data/mysql.sock prepare  **--执行**  sysbench --test=oltp --mysql-table-engine=innodb --mysql-user=root --mysql-password=123 --db-driver=mysql --mysql-db=test --oltp-table-size=10000000 --oltp-table-name=t1 --num-threads=16 --mysql-socket=/data/mysql/data/mysql.sock run  **--清理**  drop table t1 |

## mysql-tpcc

git clone <https://github.com/Percona-Lab/tpcc-mysql>

|  |
| --- |
| #1 Build binaries  cd scr  #( you should have mysql\_config available in $PATH)  make  #2 Load data  ln -s /usr/local/src/mysql-5.7.13/libmysql/libmysqlclient.so.20 /usr/lib64/libmysqlclient.so.20  create database ：mysqladmin create tpcc1000  create tables： mysql tpcc1000 < create\_table.sql  create indexes and FK ( this step can be done after loading data)：  mysql tpcc1000 < add\_fkey\_idx.sql  # ./tpcc\_load --help 查看所有选项  populate data：./tpcc\_load -h127.0.0.1 -d tpcc1000 -u root -p "" -w 1000  每列含义：|hostname:port| |dbname| |user| |password| |WAREHOUSES|  load data in parallel check load.sh script  举例：./tpcc\_load -h127.0.0.1 -d tpcc100 -u root -p "123" -w 100  #3start benchmark  #./tpcc\_start ref. tpcc\_start --help查看所有选项  ./tpcc\_start -h127.0.0.1 -P3306 -dtpcc1000 -uroot -w1000 -c32 -r10 -l10800  每列含义：|hostname| |port| |dbname| |user| |WAREHOUSES| |CONNECTIONS| |WARMUP TIME| |BENCHMARK TIME|  举例：./tpcc\_start -h127.0.0.1 -P3306 -dtpcc100 -uroot -p123 -w100 -c32 -r300 -l3600 |