

# Mysql5.7 Failover Test

今回はSysbenchのベンチマークテストをご紹介、テスト対象はMysql5.7とMysql5.8です

## 概要

本文章はSysbenchでMysqlの性能測定をご紹介します。

## 1 Sysbench環境をインストールする

## 2 Sysbenchでデータを準備する

## 3 Failoverテストを実行する

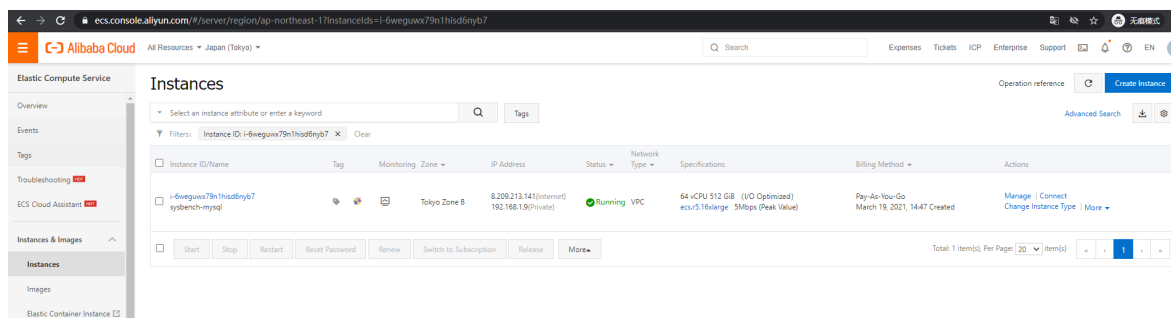
## 1 Sysbench環境をインストールする

Sysbenchとはよくデータベース、ファイルシステムやCPU、メモリなどシステムのベンチマークを行うソフトウェアです。

### 1-1 Sysbenchインストール

#### 1) ECSインスタンスを作成する

ECS:  
Specifications: 64 vCPU 512 GiB ecs.r5.16xlarge  
OS: CentOS 7.7 64-bit



#### 2) Sysbenchをインストールする

①以下のコマンドでSysbenchインストールが行えます。

```
# yum install gcc gcc-c++ autoconf automake make libtool bzip2 mysql-devel git mysql
```

```
[root@iz6weguwx79n1hisd6nyb7Z ~]# yum install gcc gcc-c++ autoconf automake make libtool bzip2 mysql-devel git mysql
Loaded plugins: fastestmirror
Determining fastest mirrors
```

base

| 3.6 kB 00:00:00

epel

| 4.7 kB 00:00:00

extras

| 2.9 kB 00:00:00

updates

| 2.9 kB 00:00:00

(1/7): epel/x86\_64/group\_gz

| 96 kB 00:00:00

.....

Installed:

autoconf.noarch 0:2.69-11.e17 automake.noarch 0:1.13.4-3.e17  
bzip2.x86\_64 0:2.5.1-14.e17 gcc.x86\_64 0:4.8.5-44.e17 gcc-c++.x86\_64  
0:4.8.5-44.e17 git.x86\_64 0:1.8.3.1-23.e17\_8 libtool.x86\_64 0:2.4.2-  
22.e17\_3 mariadb.x86\_64 1:5.5.68-1.e17  
mariadb-devel.x86\_64 1:5.5.68-1.e17

Dependency Installed:

cpp.x86\_64 0:4.8.5-44.e17 glibc-devel.x86\_64 0:2.17-  
323.e17\_9 glibc-headers.x86\_64 0:2.17-323.e17\_9 kernel-headers.x86\_64  
0:3.10.0-1160.15.2.e17 keyutils-libs-devel.x86\_64 0:1.5.8-3.e17 krb5-  
devel.x86\_64 0:1.15.1-50.e17  
libcom\_err-devel.x86\_64 0:1.42.9-19.e17 libkadm5.x86\_64 0:1.15.1-50.e17  
libmpc.x86\_64 0:1.0.1-3.e17 libselinux-devel.x86\_64  
0:2.5-15.e17 libsepol-devel.x86\_64 0:2.5-10.e17 libstdc++-  
devel.x86\_64 0:4.8.5-44.e17  
libverto-devel.x86\_64 0:0.2.5-4.e17 mpfr.x86\_64 0:3.1.1-4.e17  
openssl-devel.x86\_64 1:1.0.2k-21.e17\_9 pcre-devel.x86\_64 0:8.32-  
17.e17 perl-Data-Dumper.x86\_64 0:2.145-3.e17 perl-  
Error.noarch 1:0.17020-2.e17  
perl-Git.noarch 0:1.8.3.1-23.e17\_8 perl-TermReadKey.x86\_64 0:2.30-  
20.e17 perl-Test-Harness.noarch 0:3.28-3.e17 perl-Thread-Queue.noarch  
0:3.02-2.e17 zlib-devel.x86\_64 0:1.2.7-19.e17\_9

Dependency Updated:

e2fsprogs.x86\_64 0:1.42.9-19.e17 e2fsprogs-libs.x86\_64 0:1.42.9-19.e17  
glibc.x86\_64 0:2.17-323.e17\_9 glibc-common.x86\_64 0:2.17-323.e17\_9  
krb5-libs.x86\_64 0:1.15.1-50.e17 libcom\_err.x86\_64 0:1.42.9-19.e17  
libgcc.x86\_64 0:4.8.5-44.e17  
libgomp.x86\_64 0:4.8.5-44.e17 libselinux.x86\_64 0:2.5-15.e17  
libselinux-python.x86\_64 0:2.5-15.e17 libselinux-utils.x86\_64 0:2.5-15.e17  
libss.x86\_64 0:1.42.9-19.e17 libstdc++.x86\_64 0:4.8.5-44.e17  
mariadb-libs.x86\_64 1:5.5.68-1.e17  
nscd.x86\_64 0:2.17-323.e17\_9 openssl.x86\_64 1:1.0.2k-21.e17\_9  
openssl-libs.x86\_64 1:1.0.2k-21.e17\_9 zlib.x86\_64 0:1.2.7-19.e17\_9  
Complete!

```
| 3.6 kB 00:00:00
| 4.7 kB 00:00:00
| 2.9 kB 00:00:00
| 2.9 kB 00:00:00
| 96 kB 00:00:00
| 1.0 MB 00:00:00
| 225 kB 00:00:00
| 6.9 MB 00:00:00
| 5.7 MB 00:00:00
| 6.1 MB 00:00:00
| 153 kB 00:00:00
```

30/6  
31/6  
32/6  
33/6  
34/6  
35/6  
36/6  
37/6  
38/6  
39/6  
40/6  
41/6  
42/6  
43/6  
44/6  
45/6  
46/6  
47/6  
48/6  
49/6  
50/6  
51/6  
52/6  
53/6  
54/6  
55/6  
56/6  
57/6  
58/6  
59/6  
60/6  
61/6  
62/6  
63/6  
64/6  
65/6  
66/6  
67/6  
68/6

②下記のリンクからSysbenchをダウンロードする

```
# git clone https://github.com/akopytov/sysbench.git
```

```
[root@iz6weguwx79n1hisd6nyb7Z ~]# git clone
https://github.com/akopytov/sysbench.git
Cloning into 'sysbench'...
remote: Enumerating objects: 62, done.
remote: Counting objects: 100% (62/62), done.
remote: Compressing objects: 100% (33/33), done.
remote: Total 10220 (delta 28), reused 44 (delta 23), pack-reused 10158
Receiving objects: 100% (10220/10220), 4.23 MiB | 1.38 MiB/s, done.
Resolving deltas: 100% (7326/7326), done.
```

### ③SysBench 1.0.18バージョンにチェックアウトする

```
# cd sysbench
# git checkout 1.0.18
```

```
[root@iZ6weguwx79n1hisd6nyb7Z ~]# cd sysbench
[root@iZ6weguwx79n1hisd6nyb7Z sysbench]# git checkout 1.0.18
Note: checking out '1.0.18'.
```

You are in 'detached HEAD' state. You can look around, make experimental

changes and commit them, and you can discard any commits you make in this state without impacting any branches by performing another checkout.

If you want to create a new branch to retain commits you create, you may do so (now or later) by using `-b` with the checkout command again. Example:

```
git checkout -b new_branch_name
```

HEAD is now at ab7d582... Release 1.0.18.

```
[root@iz6weguwx79nlhisd6nyb7Z ~]# cd sysbench
[root@iz6weguwx79nlhisd6nyb7Z sysbench]# git checkout 1.0.18
Note: checking out '1.0.18'.
```

You are in 'detached HEAD' state. You can look around, make experimental changes and commit them, and you can discard any commits you make in this state without impacting any branches by performing another checkout.

If you want to create a new branch to retain commits you create, you may do so (now or later) by using `-b` with the checkout command again. Example:

```
git checkout -b new_branch_name
```

HEAD is now at ab7d582... Release 1.0.18.

#### ④autogen.shを実行します

```
# ./autogen.sh
# ./configure --prefix=/usr --mandir=/usr/share/man
```

```
[root@iz6weguwx79nlhisd6nyb7Z sysbench]# ./autogen.sh
./autogen.sh: running `libtoolize --copy --force'
libtoolize: putting auxiliary files in AC_CONFIG_AUX_DIR, `config'.
libtoolize: copying file `config/ltmain.sh'
libtoolize: putting macros in AC_CONFIG_MACRO_DIR, `m4'.
libtoolize: copying file `m4/libtool.m4'
libtoolize: copying file `m4/ltoptions.m4'
libtoolize: copying file `m4/ltugar.m4'
libtoolize: copying file `m4/ltversion.m4'
libtoolize: copying file `m4/lt~obsolete.m4'
./autogen.sh: running `aclocal -I m4'
./autogen.sh: running `autoheader'
./autogen.sh: running `automake -c --foreign --add-missing'
configure.ac:59: installing 'config/ar-lib'
configure.ac:45: installing 'config/compile'
configure.ac:27: installing 'config/config.guess'
configure.ac:27: installing 'config/config.sub'
configure.ac:32: installing 'config/install-sh'
configure.ac:32: installing 'config/missing'
src/Makefile.am: installing 'config/depcomp'
parallel-tests: installing 'config/test-driver'
./autogen.sh: running `autoconf'
Libtoolized with: libtoolize (GNU libtool) 2.4.2
Automade with: automake (GNU automake) 1.13.4
Configured with: autoconf (GNU Autoconf) 2.69
[root@iz6weguwx79nlhisd6nyb7Z sysbench]# ./configure --prefix=/usr --
mandir=/usr/share/man
checking build system type... x86_64-unknown-linux-gnu
checking host system type... x86_64-unknown-linux-gnu
```

```
checking target system type... x86_64-unknown-linux-gnu
checking for a BSD-compatible install... /usr/bin/install -c
checking whether build environment is sane... yes
checking for a thread-safe mkdir -p... /usr/bin/mkdir -p
.....
config.status: executing libtool commands
=====
sysbench version   : 1.0.18-ab7d582
CC                 : gcc -std=gnu99
CFLAGS             : -O2 -funroll-loops -ggdb3 -march=core2 -Wall -Wextra -
Wpointer-arith -Wbad-function-cast -Wstrict-prototypes -Wnested-externs -Wno-
format-zero-length -Wundef -Wstrict-prototypes -Wmissing-prototypes -Wmissing-
declarations -Wredundant-decls -Wcast-align -Wvla -pthread
CPPFLAGS           : -D_GNU_SOURCE -I$(top_srcdir)/src -
I$(abs_top_builddir)/third_party/luajit/inc -
I$(abs_top_builddir)/third_party/concurrency_kit/include
LDFLAGS            : -L/usr/lib
LIBS               : -lm
EXTRA_LDFLAGS      :

prefix             : /usr
bindir             : ${prefix}/bin
libexecdir         : ${prefix}/libexec
mandir             : /usr/share/man
datadir            : ${prefix}/share

MySQL support      : yes
Drizzle support    : no
AttachSQL support  : no
Oracle support     : no
PostgreSQL support : no

LuaJIT             : bundled
LUAJIT_CFLAGS      : -I$(abs_top_builddir)/third_party/luajit/inc
LUAJIT_LIBS        : $(abs_top_builddir)/third_party/luajit/lib/libluajit-5.1.a
-lld
LUAJIT_LDFLAGS     : -rdynamic

Concurrency Kit    : bundled
CK_CFLAGS          : -I$(abs_top_builddir)/third_party/concurrency_kit/include
CK_LIBS            : $(abs_top_builddir)/third_party/concurrency_kit/lib/libck.a
configure flags    :
=====
```

```
[root@iz6weguwx79n1hisd6nyb7Z sysbench]# ./autogen.sh
./autogen.sh: running 'libtoolize --copy --force'
libtoolize: putting auxiliary files in AC_CONFIG_AUX_DIR, 'config'.
libtoolize: copying file 'config/ltmain.sh'
libtoolize: putting macros in AC_CONFIG_MACRO_DIR, 'm4'.
libtoolize: copying file 'm4/libtool.m4'
libtoolize: copying file 'm4/ltoptions.m4'
libtoolize: copying file 'm4/ltsugar.m4'
libtoolize: copying file 'm4/ltversion.m4'
libtoolize: copying file 'm4/lt-obsolete.m4'
./autogen.sh: running 'aclocal -I m4'
./autogen.sh: running 'autoheader'
./autogen.sh: running 'automake -c --foreign --add-missing'
configure.ac:59: installing 'config/ar-lib'
configure.ac:45: installing 'config/compile'
configure.ac:27: installing 'config/config.guess'
configure.ac:27: installing 'config/config.sub'
configure.ac:32: installing 'config/install-sh'
configure.ac:32: installing 'config/missing'
src/Makefile.am: installing 'config/depcomp'
parallel-tests: installing 'config/test-driver'
./autogen.sh: running 'autoconf'
Libtoolized with: libtoolize (GNU libtool) 2.4.2
Automake with: automake (GNU automake) 1.13.4
Configured with: autoconf (GNU Autoconf) 2.69
[root@iz6weguwx79n1hisd6nyb7Z sysbench]# ./configure --prefix=/usr --mandir=/usr/share/man
checking build system type... x86_64-unknown-linux-gnu
checking host system type... x86_64-unknown-linux-gnu
checking target system type... x86_64-unknown-linux-gnu
checking for a BSD-compatible install... /usr/bin/install -c
checking whether build environment is sane... yes
checking for a thread-safe mkdir -p... /usr/bin/mkdir -p
checking for gawk... gawk
checking whether make sets $(MAKE)... yes
checking whether make supports nested variables... yes
checking for style of include used by make... GNU
checking for gcc... gcc
checking whether the C compiler works... yes
checking for C compiler default output file name... a.out
checking for suffix of executables...
checking whether we are cross compiling... no
checking for suffix of object files... o
checking whether we are using the GNU C compiler... yes
checking whether gcc accepts -g... yes
checking for gcc option to accept ISO C89... none needed
checking dependency style of gcc... gcc3
checking how to run the C preprocessor... gcc -E
checking for grep that handles long lines and -e... /usr/bin/grep
checking for egrep... /usr/bin/grep -E
checking for ANSI C header files... yes
checking for sys/types.h... yes
checking for sys/stat.h... yes
checking for stdlib.h... yes
```

```
config.status: creating doc/Makefile
config.status: creating third_party/luajit/Makefile
config.status: creating third_party/concurrency_kit/Makefile
config.status: creating src/Makefile
config.status: creating src/drivers/Makefile
config.status: creating src/drivers/mysql/Makefile
config.status: creating src/drivers/drizzle/Makefile
config.status: creating src/drivers/oracle/Makefile
config.status: creating src/drivers/pqsql/Makefile
config.status: creating src/drivers/redis/Makefile
config.status: creating src/test/Makefile
config.status: creating src/test/cpu/Makefile
config.status: creating src/test/files/Makefile
config.status: creating src/test/memory/Makefile
config.status: creating src/test/network/Makefile
config.status: creating src/test/mutex/Makefile
config.status: creating src/ua/internal/Makefile
config.status: creating tests/Makefile
config.status: creating tests/include/config.sh
config.status: creating snap/snapsrcraft.yaml
config.status: creating config/config.h
config.status: executing depfiles commands
config.status: executing libtool commands

sysbench version : 1.0.10-ab7d82
CC               : gcc -std=c99
CFLAGS           : -O2 -funroll-loops -gdb3 -march=core2 -Wall -Wextra -Wpointer-arith -Wbad-function-cast -Wstrict-prototypes -Wnested-externs -Wno-format-zero-length -Wundef -Wstrict-prototypes -Wmissing-prototypes -Wmissing-declarations -Wredundant-decls -Wcast-a
top_wala         : pthread
CPPFLAGS         : -D_GNU_SOURCE -Isys_top_srcdir/src -Isys_top_builddir/third_party/luajit/inc -Isys_top_builddir/third_party/concurrency_kit/include
LDFLAGS          : -L/usr/lib
LIBS             : -lm
EXTRA_LDFLAGS    :

prefix           : /usr
bindir            : ${prefix}/bin
libexecdir        : ${prefix}/libexec
mandir           : /usr/share/man
datadir          : ${prefix}/share

MySQL support    : yes
Drizzle support  : no
AtaSQL support   : no
Oracle support   : no
PostgreSQL support : no

LUAJIT          : bundled
LUAJIT_CFLAGS   : -Isys_top_builddir/third_party/luajit/inc
LUAJIT_LIBS     : sys_top_builddir/third_party/luajit/lib/libluajit-5.1.a -ldl
LUAJIT_LDFLAGS  : -rdynamic

Concurrency Kit  : bundled
CK_CFLAGS       : -Isys_top_builddir/third_party/concurrency_kit/include
CK_LIBS         : sys_top_builddir/third_party/concurrency_kit/lib/libck.a
Configure flags  :

[root@iz6weguwx79n1hisd6nyb7Z sysbench]#
```

## ⑤コンパイル

```
# make
# make install
```

```
[root@iz6weguwx79n1hisd6nyb7Z sysbench]# make
Making all in doc
make[1]: Entering directory `/root/sysbench/doc'
Making all in xs1
```

```
make[2]: Entering directory `/root/sysbench/doc/xs1'
make[2]: Nothing to be done for `all'.
make[2]: Leaving directory `/root/sysbench/doc/xs1'
make[2]: Entering directory `/root/sysbench/doc'
touch manual.html
make[2]: Leaving directory `/root/sysbench/doc'
make[1]: Leaving directory `/root/sysbench/doc'
Making all in third_party/luajit
make[1]: Entering directory `/root/sysbench/third_party/luajit'
make -C ./luajit clean
make[2]: Entering directory `/root/sysbench/third_party/luajit/luajit'
make -C src clean
省略
完了しました
```

```

root@2b6m9ao70h1hld0n972 sybench1# make
Making all in doc
make[1]: Entering directory '/root/sybench/doc'
Making all in xsl
make[2]: Entering directory '/root/sybench/doc/xsl'
make[2]: Nothing to be done for 'all'.
make[2]: Leaving directory '/root/sybench/doc/xsl'
make[1]: Entering directory '/root/sybench/doc'
touch manual.html
make[1]: Leaving directory '/root/sybench/doc'
make[1]: Leaving directory '/root/sybench/doc'
Making all in third_party/luajit
make[1]: Entering directory '/root/sybench/third_party/luajit'
make -C ./luajit clean
make[2]: Entering directory '/root/sybench/third_party/luajit/luajit'
make -C src clean
make[2]: Entering directory '/root/sybench/third_party/luajit/luajit/src'
make[2]: Leaving directory '/root/sybench/third_party/luajit/luajit/src'
make[1]: Leaving directory '/root/sybench/third_party/luajit/luajit'
rm -rf tmp
mkdir tmp
tar -x -C -cf - luajit | tar -x -f - -C tmp/
chmod -R uwx tmp
make -C tmp/luajit
make[2]: Entering directory '/root/sybench/third_party/luajit \
INSTALL_INC=/root/sybench/third_party/luajit/inc \
install
make[2]: Entering directory '/root/sybench/third_party/luajit/tmp/luajit'
make -C src
make[2]: Entering directory '/root/sybench/third_party/luajit/tmp/luajit/src'
HOSTCC host/minilua.o
HOSTCC host/minilua
DYNASM host/build_arch.h
HOSTCC host/build.o
HOSTCC host/build.o.o
HOSTCC host/build_peeb.o
HOSTCC host/build_lib.o
HOSTCC host/build_tbl.o
HOSTLINK host/build
BUILDMM lj.o.o
ASM lj.o.o
CC lj.o.o
BUILDMM lj_ffdef.h
CC lj_err.o
CC lj_char.o
BUILDMM lj_bcode.h
CC lj_bc.o
CC lj_obj.o
CC lj_buf.o
CC lj_str.o
CC lj_tab.o
CC lj_func.o
CC lj_udata.o

```

[illegible]

```

make[3]: Entering directory '/root/sysbench/src/tests'
make[4]: Entering directory '/root/sysbench/src/tests'
make[4]: Nothing to be done for 'install-data-am'.
make[4]: Leaving directory '/root/sysbench/src/tests'
make[3]: Leaving directory '/root/sysbench/src/tests'
Making install in lua
make[2]: Entering directory '/root/sysbench/src/lua'
Making install in internal
make[3]: Entering directory '/root/sysbench/src/lua/internal'
make install-am
make[4]: Entering directory '/root/sysbench/src/lua/internal'
make[5]: Entering directory '/root/sysbench/src/lua/internal'
make[5]: Nothing to be done for 'install-data-am'.
make[5]: Leaving directory '/root/sysbench/src/lua/internal'
make[4]: Leaving directory '/root/sysbench/src/lua/internal'
make[3]: Leaving directory '/root/sysbench/src/lua'
make[2]: Leaving directory '/root/sysbench/src/lua'
make[4]: Entering directory '/root/sysbench/src/lua'
make[4]: Nothing to be done for 'install-exec-am'.
/usr/bin/mkdir -p '/usr/share/sysbench'
/usr/bin/install -c -m 644 oltp_comm.lua '/usr/share/sysbench'
/usr/bin/install -c bulk_insert.lua oltp_delete.lua oltp_insert.lua oltp_read_only.lua oltp_read_write.lua oltp_select.lua oltp_update_index.lua oltp_update_non_index.lua oltp_write_only.lua select_random_points.lua select_random_ranges.lua '/usr/share/sysbench'
make[4]: Leaving directory '/root/sysbench/src/lua'
make[2]: Leaving directory '/root/sysbench/src/lua'
Making install in
make[2]: Entering directory '/root/sysbench/src'
make[3]: Entering directory '/root/sysbench/src'
/usr/bin/mkdir -p '/usr/bin'
/libtool --libtool --mode=install /usr/bin/install -c sysbench '/usr/bin'
libtool: install: /usr/bin/install -c sysbench /usr/bin/sysbench
make[3]: Nothing to be done for 'install-data-am'.
make[3]: Leaving directory '/root/sysbench/src'
make[2]: Leaving directory '/root/sysbench/src'
make[1]: Leaving directory '/root/sysbench/src'
Making install in tests
make[1]: Entering directory '/root/sysbench/tests'
make[2]: Entering directory '/root/sysbench/tests'
make[2]: Nothing to be done for 'install-exec-am'.
make -IINSTALL_TO_DIR="/usr/share/sysbench/tests" install_test_files
make[2]: Leaving directory '/root/sysbench/tests'
make[3]: Leaving directory '/root/sysbench/tests'
/usr/bin/mkdir -p '/usr/share/sysbench/tests'
/usr/bin/install -c test_run.sh '/usr/share/sysbench/tests'
make[2]: Leaving directory '/root/sysbench/tests'
make[1]: Leaving directory '/root/sysbench'
make[1]: Entering directory '/root/sysbench'
make[2]: Entering directory '/root/sysbench'
make[2]: Nothing to be done for 'install-data-am'.
make[2]: Leaving directory '/root/sysbench'
make[1]: Leaving directory '/root/sysbench'
[root@iz20wguwx79nlhisd6nyb7Z sysbench]#

```

⑥SysBench clientの設定、ffffffは 32 coresが使われている。

```
sudo sh -c 'for x in /sys/class/net/eth0/queues/rx-*; do echo
ffffff>$x/rps_cpus; done'
```

```
sudo sh -c "echo 32768 > /proc/sys/net/core/rps_sock_flow_entries"
sudo sh -c "echo 4096 > /sys/class/net/eth0/queues/rx-0/rps_flow_cnt"
sudo sh -c "echo 4096 > /sys/class/net/eth0/queues/rx-1/rps_flow_cnt"
```

```

[root@iz20wguwx79nlhisd6nyb7Z sysbench]# sudo sh -c 'for x in /sys/class/net/eth0/queues/rx-*; do echo fffffff>$x/rps_cpus; done'
[root@iz20wguwx79nlhisd6nyb7Z sysbench]# sudo sh -c "echo 32768 > /proc/sys/net/core/rps_sock_flow_entries"
[root@iz20wguwx79nlhisd6nyb7Z sysbench]# sudo sh -c "echo 4096 > /sys/class/net/eth0/queues/rx-0/rps_flow_cnt"
[root@iz20wguwx79nlhisd6nyb7Z sysbench]# sudo sh -c "echo 4096 > /sys/class/net/eth0/queues/rx-1/rps_flow_cnt"
[root@iz20wguwx79nlhisd6nyb7Z sysbench]#

```

⑦インストール完了

```

[root@iz20wguwx79nlhisd6nyb7Z sysbench]# sysbench --version
sysbench 1.0.18-ab7d582
[root@iz20wguwx79nlhisd6nyb7Z sysbench]#

```

操作ガイドは下記のユーザーガイドもご参照ください

<https://www.alibabacloud.com/help/doc-detail/146103.htm?spm=a2c63.l28256.b99.186.188d3784k2PgLLH>

iZ6wguwx79n1hisd6nyb7ZのImageを取って、ECSを作成する

## 3) Polardbインスタンスを作成する

PolarDB: Mysql5.7 8Core32GB

① Polardbインスタンスを作成する



Alibaba Cloud

Cart Tickets Bills English + Oiw\_admin@ab...

Confirm Order

Parameters

Apsara PolarDB (Pay-as-you-go)

Configurations	Payment Option	Quantity	Discount	Price
Region: Japan (Tokyo) Create Type: Create Primary Cluster Primary Availability Zone: Zone B Network Type: VPC VPC: PolarDBBenchmark VSwitch: PolarDBvswB Compatibility: MySQL 5.7 Edition: Cluster (2-16 Nodes) (Recommended) Node Specification: 8 Cores 32 GB (dedicated) Nodes: 2 Time Zone: UTC +08:00 (default) Table Name Case Sensitivity: Not Case-sensitive (Default) Release Cluster: Retain Last Automatic Backup (Automatic Backup before Release) (Default) Cluster Name: 8Core32GB	Pay-As-You-Go	1	\$0.000/hour	Configuration Fee: \$2.060/hour

Terms of Service ☒ I have read and agree to Apsara PolarDB (Pay-as-you-go) Agreement of Service

Payable \$2.060/hour [Activate Now](#)

## ② PolarDBのホワイトリストにECSプライベートIPを追加する

Alibaba Cloud

Q Search Expenses Tickets ICP Er

Cluster / Whitelists

← 8Core32GB | pc-0iw244o83t3g94791 Running

[Log On to Database](#) [Create GDN](#) [Migrate Data to Current Cluster](#)

Overview  
Settings and Management  
Whitelists  
Security Management  
Accounts  
Databases  
Backup and Restore

IP List

[Add IP Whitelist](#)

- default

192.168.1.235

## ③ PolarDBのアカウントを追加する

Alibaba Cloud

Q Search Expenses Tickets ICP Er

Cluster / Accounts

← 8Core32GB | pc-0iw244o83t3g94791 Running

[Log On to Database](#) [Create GDN](#) [Migrate Data to Current Cluster](#)

Overview  
Settings and Management  
Whitelists  
Security Management  
Accounts  
Databases  
Backup and Restore  
Parameters

[Create Account](#) [Customize Permissions](#) [Fuzzy Match \(Current Page\)](#)

Account Name	Status	Database Name	Description	Type	Actions
sttest	Active	-	🔗	Privileged Account	<a href="#">Reset Permissions</a> <a href="#">Change Password</a> <a href="#">Delete</a>

Items per Page: 30

## ④ テストDBを追加する

Alibaba Cloud

Q Search Expenses Tickets ICP Er

Cluster / Databases

← 8Core32GB | pc-0iw244o83t3g94791 Running

[Log On to Database](#) [Create GDN](#) [Migrate Data to Current Cluster](#)

Overview  
Settings and Management  
Whitelists  
Security Management  
Accounts  
Databases  
Backup and Restore  
Parameters

[Create Database](#) [Fuzzy Match \(Current Page\)](#)

Database Name	Status	Character Set	Authorized Account	Description
sttest	Running	utf8mb4	-	🔗

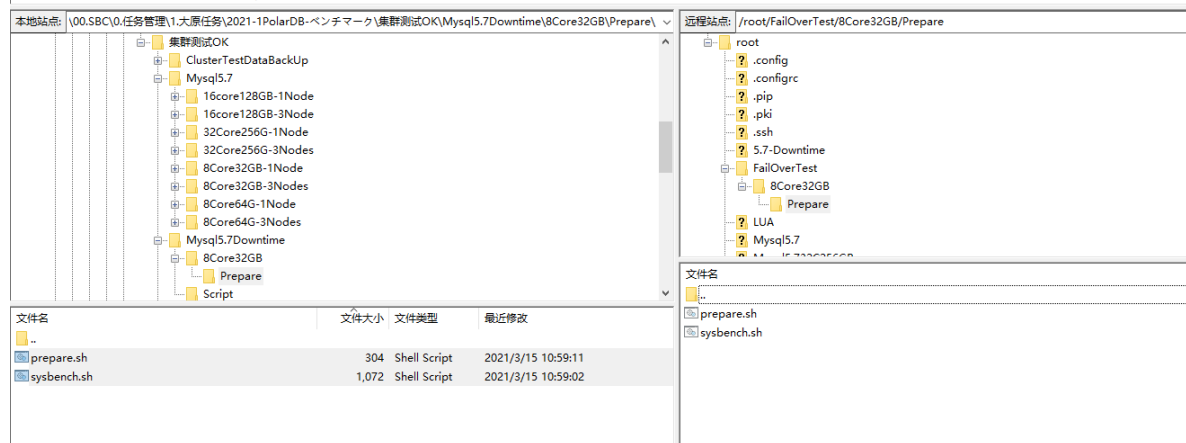
Items per Page: 30

# 2 Sysbenchでデータを準備する

## 2-1 Polardb-Mysql5.7に100GBデータを初期化する

## ① ScriptをECSにアップロードする

状態: Listing directory /root/FailOverTest/8Core32GB/Prepare  
状態: 列出 /root/FailOverTest/8Core32GB/Prepare の目录成功



スクリプトファイル (sysbench.shとprepare.sh) を用意する  
sysbench.sh

```
#!/bin/sh
LUA=/usr/share/sysbench/oltp_read_write.lua
SIZE=5000000
DB=mysql
#prepare data using primary host
HOST=pc-0iwjkn9do507eiak.mysql.polardb.japan.rds.aliyuncs.com
PORT=3306
USER=sbtest
PASSWORD=Test1234
DBNAME=sbtest
usage()
{
    echo "Usage: ./sysbench.sh <prepare|run|cleanup> <num of threads>"
    exit "${1}"
}
#chack argumets
if [ "${1}" = "" -o $# -gt 3 ]; then
    usage 1
elif [ "${2}" = "" ]; then
    THREADS=1
else
    THREADS=${2}
fi
echo "Running command: sysbench ${LUA} --db-driver=${DB} --mysql-host=${HOST} --mysql-port=${PORT} --mysql-user=${USER} --mysql-password=${PASSWORD} --mysql-db=${DBNAME} --table-size=${SIZE} --tables=500 --events=0 --time=60 --db-ps-mode=disable --percentile=95 --report-interval=1 --threads=${THREADS} ${1}"
sysbench ${LUA} --db-driver=${DB} --mysql-host=${HOST} --mysql-port=${PORT} --mysql-user=${USER} --mysql-password=${PASSWORD} --mysql-db=${DBNAME} --table-size=${SIZE} --tables=20 --events=0 --time=120 --db-ps-mode=disable --percentile=95 --report-interval=1 --threads=${THREADS} ${1}
```

prepare.sh

```
#!/bin/sh
mkdir -p logs
thread=100
echo "prepare data using default settings, ref sysbench SIZE" >>
logs/sysbench_read_write_0_prepare.log
./sysbench.sh prepare ${thread} >> logs/sysbench_read_write_0_prepare.log
echo "data had been successfully initialized." >>
logs/sysbench_read_write_0_prepare.log
```

②  
Sysbenchでおよそ100GBデータをDBにPushする

```
# chmod a+x *.sh
# nohup sh prepare.sh 2>&1&
```

```
-rwxr-xr-x 1 root root 926557 Mar 14 09:34 result_event_2000_concurrent_10.log
-rwxr-xr-x 1 root root 229 Mar 14 09:23 run-10t.sh
-rwxr-xr-x 1 root root 5529 Mar 13 23:07 SetupMySQLFailOverTest.sql
[root@izwmpa8j6wq3o8flbms2 ~]# cd Prepare/
[root@izwmpa8j6wq3o8flbms2 Prepare]# ll
total 0
-rwxr-xr-x 1 root root 304 Mar 15 10:59 prepare.sh
-rwxr-xr-x 1 root root 1072 Mar 15 10:59 sysbench.sh
[root@izwmpa8j6wq3o8flbms2 Prepare]# chmod a+x *.sh
[root@izwmpa8j6wq3o8flbms2 Prepare]# ll
total 0
-rwxr-xr-x 1 root root 304 Mar 15 10:59 prepare.sh
-rwxr-xr-x 1 root root 1072 Mar 15 10:59 sysbench.sh
[root@izwmpa8j6wq3o8flbms2 Prepare]# nohup sh prepare.sh 2>&1&
[1] 4441
[root@izwmpa8j6wq3o8flbms2 Prepare]# nohup: ignoring input and appending output to 'nohup.out'
[root@izwmpa8j6wq3o8flbms2 Prepare]# ll
total 12
drwxr-xr-x 2 root root 4096 Mar 15 11:01 logs
-rwxr-xr-x 1 root root 0 Mar 15 11:01 nohup.out
-rwxr-xr-x 1 root root 304 Mar 15 10:59 prepare.sh
-rwxr-xr-x 1 root root 1072 Mar 15 10:59 sysbench.sh
[root@izwmpa8j6wq3o8flbms2 Prepare]# cd logs/
[root@izwmpa8j6wq3o8flbms2 logs]# ll
total 0
-rwxr-xr-x 1 root root 4096 Mar 15 11:01 sysbench_read_write_0_prepare.log
[root@izwmpa8j6wq3o8flbms2 logs]# tail -f sysbench_read_write_0_prepare.log
Inserting 5000000 records into 'sbttest37'
Inserting 5000000 records into 'sbttest9'
Inserting 5000000 records into 'sbttest15'
Inserting 5000000 records into 'sbttest17'
Inserting 5000000 records into 'sbttest16'
Inserting 5000000 records into 'sbttest11'
Inserting 5000000 records into 'sbttest24'
Inserting 5000000 records into 'sbttest13'
Inserting 5000000 records into 'sbttest23'
Inserting 5000000 records into 'sbttest25'
Inserting 5000000 records into 'sbttest3'
Inserting 5000000 records into 'sbttest14'
Inserting 5000000 records into 'sbttest31'
Inserting 5000000 records into 'sbttest10'
Inserting 5000000 records into 'sbttest19'
Inserting 5000000 records into 'sbttest43'
Inserting 5000000 records into 'sbttest8'
Inserting 5000000 records into 'sbttest40'
Inserting 5000000 records into 'sbttest65'
Inserting 5000000 records into 'sbttest19'
Inserting 5000000 records into 'sbttest38'
Inserting 5000000 records into 'sbttest84'
Inserting 5000000 records into 'sbttest12'
Inserting 5000000 records into 'sbttest49'
Inserting 5000000 records into 'sbttest36'
Inserting 5000000 records into 'sbttest47'
Inserting 5000000 records into 'sbttest59'
Inserting 5000000 records into 'sbttest52'
```

## 2-2 PolarDB-MysqlにPROCEDUREを作成する

①DMSでPolarDB-MysqlDBに接続し、下記のPROCEDUREを実行する

DMSでSetupMySQLFailOverTest.sqlファイルのSQLを実行する

```
DROP PROCEDURE IF EXISTS `PROC_TEST_CREATE_FO_REQUIRED_DATA`$$
```

ation\_schema@pc-0iw244o83t3g94791.mysql.polardb.japan.rds.....:3306 【8Core32GB】

Flexible | account

SQLConsole

Execute(F8) Format(F10) Execute Plan(F9) Saved SQL SQL Diagnostics Settings Task Orchestration Data Visualization Cross-database Query

```
1 DROP PROCEDURE IF EXISTS `PROC_TEST_CREATE_FO_REQUIRED_DATA`;
```

Execution History Execution ... x

--- A total of (1) statements executed ---

[statement 1]:  
DROP PROCEDURE IF EXISTS `PROC\_TEST\_CREATE\_FO\_REQUIRED\_DATA`  
[Success], Time consumed: 153 (ms)  
Number of rows affected: 0

execution ended, successfully executed (1)!

```
DELIMITER $$
```

```
CREATE PROCEDURE `PROC_TEST_CREATE_FO_REQUIRED_DATA` (in_data_row_count int)  
BEGIN
```

```
    DECLARE nRetRowCount int default 0;
```

```
    DECLARE nLoopCounter int default 0;
```

```
    SELECT count(table_name) INTO nRetRowCount FROM information_schema.tables  
    WHERE table_name IN ('nancy_fo_select','nancy_fo_update','nancy_fo_insert');
```

```
    IF nRetRowCount != 3 THEN
```

```
        -- create tables needed
```

```
        CREATE TABLE IF NOT EXISTS `nancy_fo_select` ( `id` bigint(20) unsigned  
        NOT NULL AUTO_INCREMENT COMMENT 'primary key', `gmt_create` datetime NOT NULL  
        DEFAULT CURRENT_TIMESTAMP COMMENT 'create time', `gmt_modified` datetime NOT  
        NULL DEFAULT CURRENT_TIMESTAMP COMMENT 'modify datetime', `aid` bigint(20)  
        unsigned NOT NULL COMMENT 'A ID', `abalance` bigint(20) unsigned NOT NULL  
        COMMENT 'A BALANCE', PRIMARY KEY (`id`), KEY `idx_a_id` (`aid`)) ENGINE=InnoDB  
        DEFAULT CHARSET=utf8mb4 ;
```

```
        CREATE TABLE IF NOT EXISTS `nancy_fo_update` ( `id` bigint(20) unsigned  
        NOT NULL AUTO_INCREMENT COMMENT 'primary key', `gmt_create` datetime NOT NULL  
        DEFAULT CURRENT_TIMESTAMP COMMENT 'create time', `gmt_modified` datetime NOT  
        NULL DEFAULT CURRENT_TIMESTAMP COMMENT 'modify datetime', `tid` bigint(20)  
        unsigned NOT NULL COMMENT 'T ID', `tbalance` bigint(20) unsigned NOT NULL  
        COMMENT 'T BALANCE', PRIMARY KEY (`id`), KEY `idx_t_id` (`tid`)) ENGINE=InnoDB  
        DEFAULT CHARSET=utf8mb4 ;
```

```
        CREATE TABLE IF NOT EXISTS `nancy_fo_insert` ( `id` bigint(20) unsigned  
        NOT NULL AUTO_INCREMENT COMMENT 'primary key', `gmt_create` datetime NOT NULL  
        DEFAULT CURRENT_TIMESTAMP COMMENT 'create time', `gmt_modified` datetime NOT  
        NULL DEFAULT CURRENT_TIMESTAMP COMMENT 'modify datetime', `tid` bigint(20)  
        unsigned NOT NULL COMMENT 'T ID', `bid` bigint(20) unsigned NOT NULL COMMENT 'B  
        ID', `aid` bigint(20) unsigned NOT NULL COMMENT 'A ID', `delta` bigint(20)  
        unsigned NOT NULL COMMENT 'DELTA', `mtime` datetime NOT NULL COMMENT 'time  
        stamp', PRIMARY KEY (`id`)) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 ;
```

```
        -- initialize data needed
```

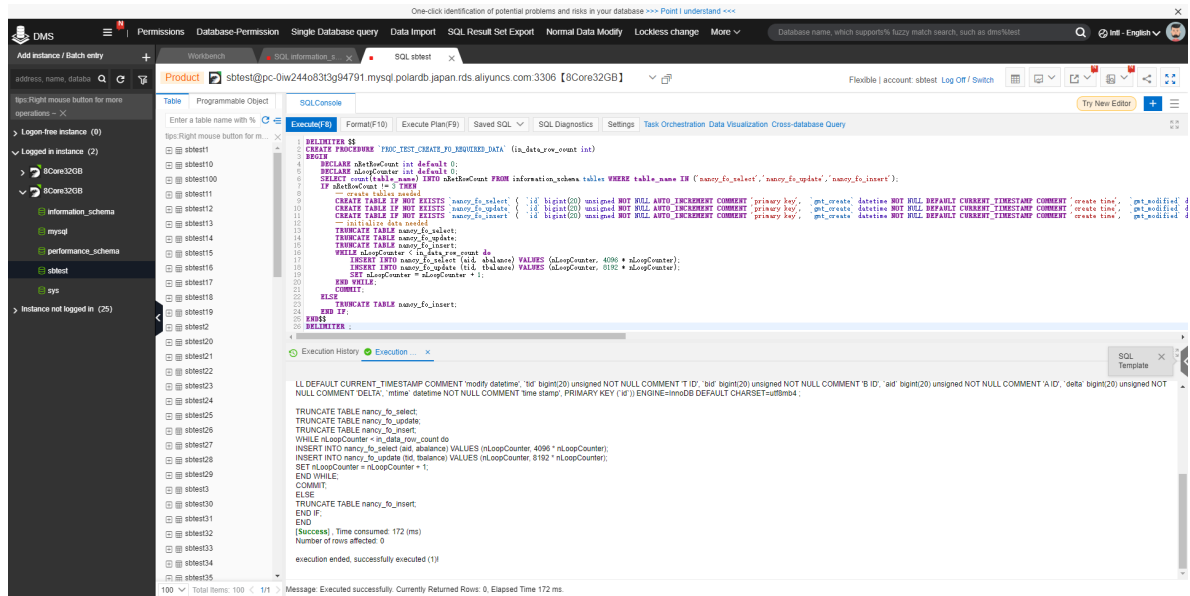
```
        TRUNCATE TABLE nancy_fo_select;
```

```

TRUNCATE TABLE nancy_fo_update;
TRUNCATE TABLE nancy_fo_insert;
WHILE nLoopCounter < in_data_row_count do
    INSERT INTO nancy_fo_select (aid, abalance) VALUES (nLoopCounter,
4096 * nLoopCounter);
    INSERT INTO nancy_fo_update (tid, tbalance) VALUES (nLoopCounter,
8192 * nLoopCounter);
    SET nLoopCounter = nLoopCounter + 1;
END WHILE;
COMMIT;

ELSE
    TRUNCATE TABLE nancy_fo_insert;
END IF;
END$$
DELIMITER ;

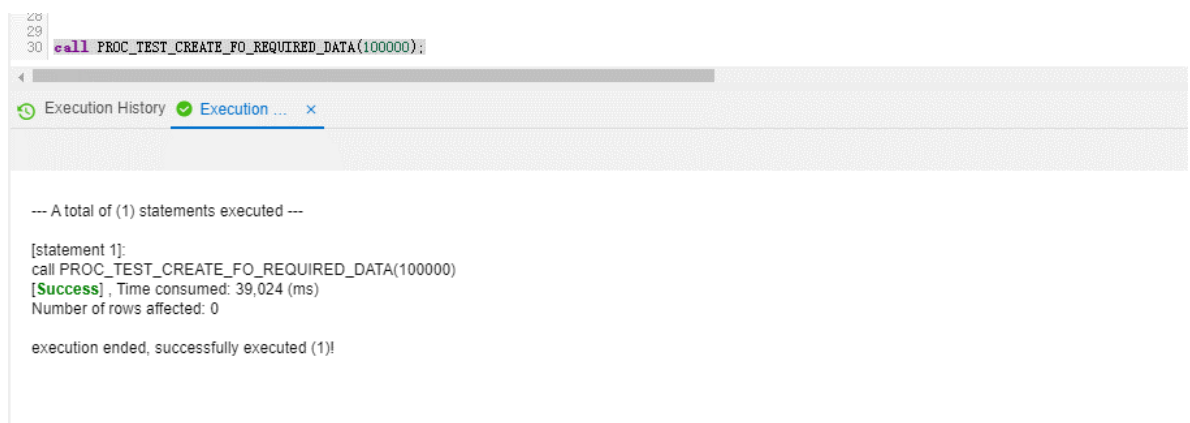
```



```

-- create table struct and init 100000 data
call PROC_TEST_CREATE_FO_REQUIRED_DATA(100000);

```



```

-- create function
DELIMITER $$
DROP FUNCTION IF EXISTS `FUNC_DO_STH_AND_RET_CURTS`$$

DELIMITER $$

```

```

31
32
33 DELIMITER $$
34 DROP FUNCTION IF EXISTS `FUNC_DO_STH_AND_RET_CURTS` $$
35
36 DELIMITER $$

```

Execution History ✓ Execution ... x

--- A total of (1) statements executed ---

[statement 1]:  
 DROP FUNCTION IF EXISTS `FUNC\_DO\_STH\_AND\_RET\_CURTS`  
 [Success], Time consumed: 151 (ms)  
 Number of rows affected: 0

execution ended, successfully executed (1)!

```

CREATE FUNCTION `FUNC_DO_STH_AND_RET_CURTS`(t_id int, b_id int, a_id int, delta_
int) RETURNS INT DETERMINISTIC
BEGIN
    DECLARE curTimeTs INT DEFAULT 0;
    DECLARE aBalanceValue INT DEFAULT 0;
    -- test logic
    SELECT abalance INTO aBalanceValue FROM nancy_fo_select WHERE aid = a_id
LIMIT 1;
    UPDATE nancy_fo_update SET tbalance = tbalance + delta_ + aBalanceValue
WHERE tid = t_id;
    INSERT INTO nancy_fo_insert (tid, bid, aid, delta, mtime) VALUES (t_id,
b_id, a_id, delta_, CURRENT_TIMESTAMP);
    SET curTimeTs = UNIX_TIMESTAMP();
    RETURN curTimeTs;
END$$
DELIMITER ;

```

```

35
36 DELIMITER $$
37
38
39 CREATE FUNCTION `FUNC_DO_STH_AND_RET_CURTS`(t_id int, b_id int, a_id int, delta_int) RETURNS INT DETERMINISTIC
40 BEGIN
41     DECLARE curTimeTs INT DEFAULT 0;
42     DECLARE aBalanceValue INT DEFAULT 0;
43     -- test logic
44     SELECT abalance INTO aBalanceValue FROM nancy_fo_select WHERE aid = a_id LIMIT 1;
45     UPDATE nancy_fo_update SET tbalance = tbalance + delta_ + aBalanceValue WHERE tid = t_id;
46     INSERT INTO nancy_fo_insert (tid, bid, aid, delta, mtime) VALUES (t_id, b_id, a_id, delta_, CURRENT_TIMESTAMP);
47     SET curTimeTs = UNIX_TIMESTAMP();
48     RETURN curTimeTs;
49 END$$
50 DELIMITER ;

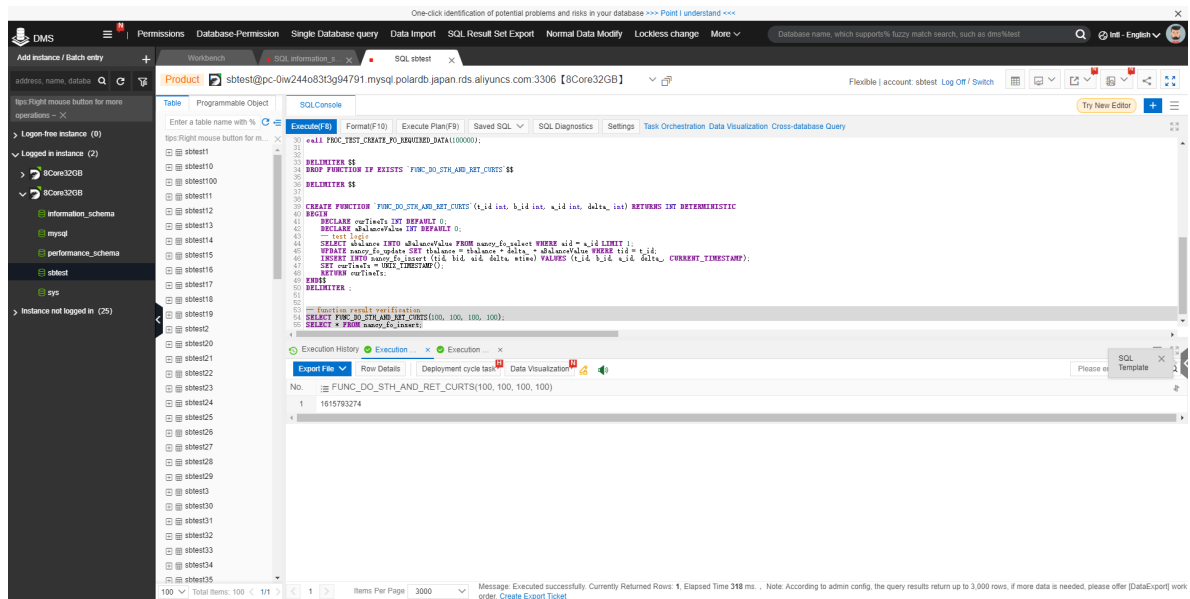
```

Execution History ✓ Execution ... x

--- A total of (1) statements executed ---

[statement 1]:  
 CREATE FUNCTION `FUNC\_DO\_STH\_AND\_RET\_CURTS`(t\_id int, b\_id int, a\_id int, delta\_int) RETURNS INT DETERMINISTIC  
 BEGIN  
 DECLARE curTimeTs INT DEFAULT 0;  
 DECLARE aBalanceValue INT DEFAULT 0;  
  
 SELECT abalance INTO aBalanceValue FROM nancy\_fo\_select WHERE aid = a\_id LIMIT 1;  
 UPDATE nancy\_fo\_update SET tbalance = tbalance + delta\_ + aBalanceValue WHERE tid = t\_id;  
 INSERT INTO nancy\_fo\_insert (tid, bid, aid, delta, mtime) VALUES (t\_id, b\_id, a\_id, delta\_, CURRENT\_TIMESTAMP);  
 SET curTimeTs = UNIX\_TIMESTAMP();  
 RETURN curTimeTs;  
 END  
 [Success], Time consumed: 191 (ms)  
 Number of rows affected: 0  
 execution ended, successfully executed (1)!

```
-- function result verification
SELECT FUNC_DO_STH_AND_RET_CURTS(100, 100, 100, 100);
SELECT * FROM nancy_fo_insert;
```



## 3 Failoverテストを実行する

### 3-1 ECSでScriptを実行開始する

①スクリプトファイル (fot.shとrun-fot.sh) を用意する  
fot.sh

```
#!/bin/sh
HOST=pc-0iw244o83t3g94791.rwlb.japan.rds.aliyuncs.com
PORT=3306
USER=sbtest
PASSWORD=Test1234
DBNAME=sbtest

## default timeout
TIMEOUT_SECONDS=2

for x in {1..10}
do
    timeout ${TIMEOUT_SECONDS} mysql -h ${HOST} -P ${PORT} -D ${DBNAME} -u ${USER} \
    -p${PASSWORD} -e "SELECT FUNC_DO_STH_AND_RET_CURTS($x,$x,$x,$x);"
    retVal=$?
    if [ $retVal -ne 0 ]; then
        echo "TIMEOUT OCCURRED! EVENT: "$1
    fi
done
```

run-fot.sh

```
#!/bin/sh

TOTAL_EVENT_COUNT=$1
CONCURRENT_PROCESS=$2

echo "Start FailOver testing right now."

#seq 1000 | xargs -t -P10 -n1 ./fot.sh
seq ${TOTAL_EVENT_COUNT} | xargs -P${CONCURRENT_PROCESS} -n1 ./fot.sh

echo "ALL DONE"
```

②Scriptを実行開始する

```
# nohup sh run-fot.sh 20000 10 &
```

```
-rwxr-xr-x 1 root root 407 Mar 15 13:37 fot.sh
-rw----- 1 root root 143344 Mar 15 15:29 nohup.out
drwxr-xr-x 3 root root 4096 Mar 15 11:01 Prepare
-rwxr-xr-x 1 root root 223 Mar 15 13:37 run-fot.sh
-rwxr-xr-x 1 root root 3537 Mar 15 13:37 SetUpMySQLFailOverTest.sql
[root@iZ6wega8j6xeq3boflbonsZ 8Core32GB]# tail nohup.out
FUNC_DO_STH_AND_RET_CURTS(3,3,3,3)
1615793385
FUNC_DO_STH_AND_RET_CURTS(10,10,10,10)
1615793385
FUNC_DO_STH_AND_RET_CURTS(6,6,6,6)
1615793385
FUNC_DO_STH_AND_RET_CURTS(2,2,2,2)
1615793385
FUNC_DO_STH_AND_RET_CURTS(2,2,2,2)
1615793385
[ root@iZ6wega8j6xeq3boflbonsZ 8Core32GB]# █
```

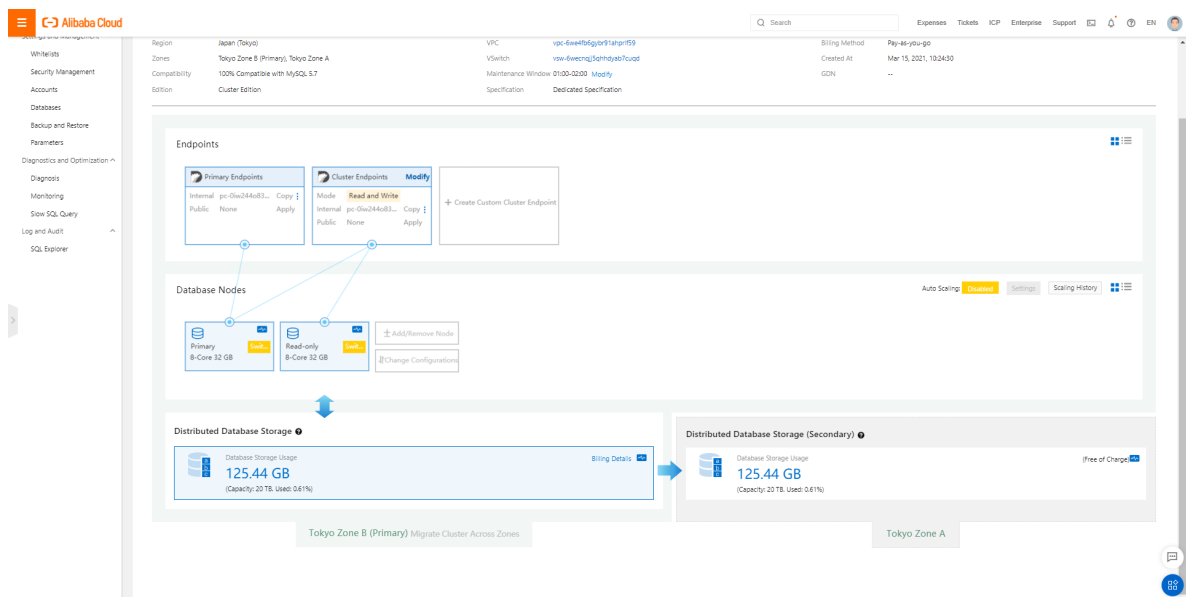
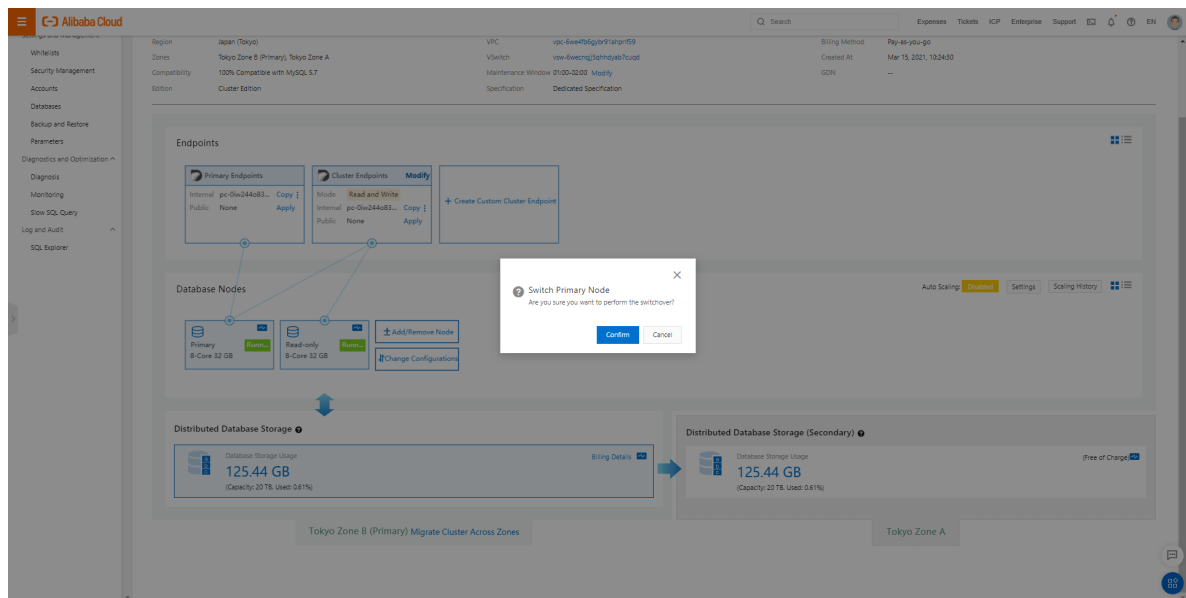
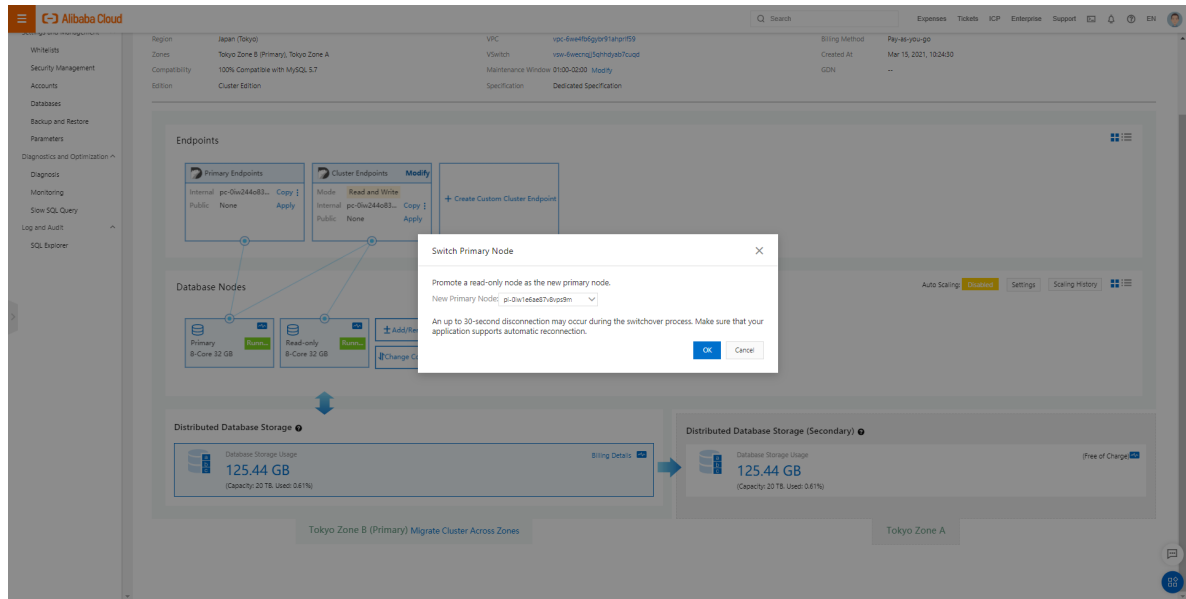


```
1615793450
FUNC_DO_STH_AND_RET_CURTS(9,9,9,9)
1615793450
FUNC_DO_STH_AND_RET_CURTS(4,4,4,4)
1615793450
FUNC_DO_STH_AND_RET_CURTS(7,7,7,7)
1615793450
FUNC_DO_STH_AND_RET_CURTS(10,10,10,10)
1615793450
FUNC_DO_STH_AND_RET_CURTS(1,1,1,1)
1615793450
FUNC_DO_STH_AND_RET_CURTS(1,1,1,1)
1615793450
FUNC_DO_STH_AND_RET_CURTS(1,1,1,1)
1615793450
FUNC_DO_STH_AND_RET_CURTS(1,1,1,1)
1615793450
FUNC_DO_STH_AND_RET_CURTS(5,5,5,5)
1615793450
FUNC_DO_STH_AND_RET_CURTS(10,10,10,10)
1615793450
FUNC_DO_STH_AND_RET_CURTS(9,9,9,9)
1615793450
FUNC_DO_STH_AND_RET_CURTS(8,8,8,8)
1615793450
FUNC_DO_STH_AND_RET_CURTS(2,2,2,2)
1615793450
FUNC_DO_STH_AND_RET_CURTS(2,2,2,2)
1615793450
FUNC_DO_STH_AND_RET_CURTS(2,2,2,2)
1615793450
FUNC_DO_STH_AND_RET_CURTS(1,1,1,1)
1615793450
FUNC_DO_STH_AND_RET_CURTS(2,2,2,2)
1615793450
FUNC_DO_STH_AND_RET_CURTS(1,1,1,1)
1615793450
FUNC_DO_STH_AND_RET_CURTS(3,3,3,3)
1615793450
FUNC_DO_STH_AND_RET_CURTS(6,6,6,6)
1615793450
FUNC_DO_STH_AND_RET_CURTS(8,8,8,8)
1615793450
FUNC_DO_STH_AND_RET_CURTS(10,10,10,10)
1615793450
FUNC_DO_STH_AND_RET_CURTS(3,3,3,3)
1615793450
FUNC_DO_STH_AND_RET_CURTS(3,3,3,3)
1615793450
FUNC_DO_STH_AND_RET_CURTS(2,2,2,2)
1615793450
FUNC_DO_STH_AND_RET_CURTS(3,3,3,3)
1615793450
FUNC_DO_STH_AND_RET_CURTS(1,1,1,1)
1615793450
FUNC_DO_STH_AND_RET_CURTS(2,2,2,2)
1615793450
█
```

## 3-2 PolarDB-MysqlコンソールでNodeを切り替える



### ③PolarDB-MysqlコンソールでNodeを切り替え開始する



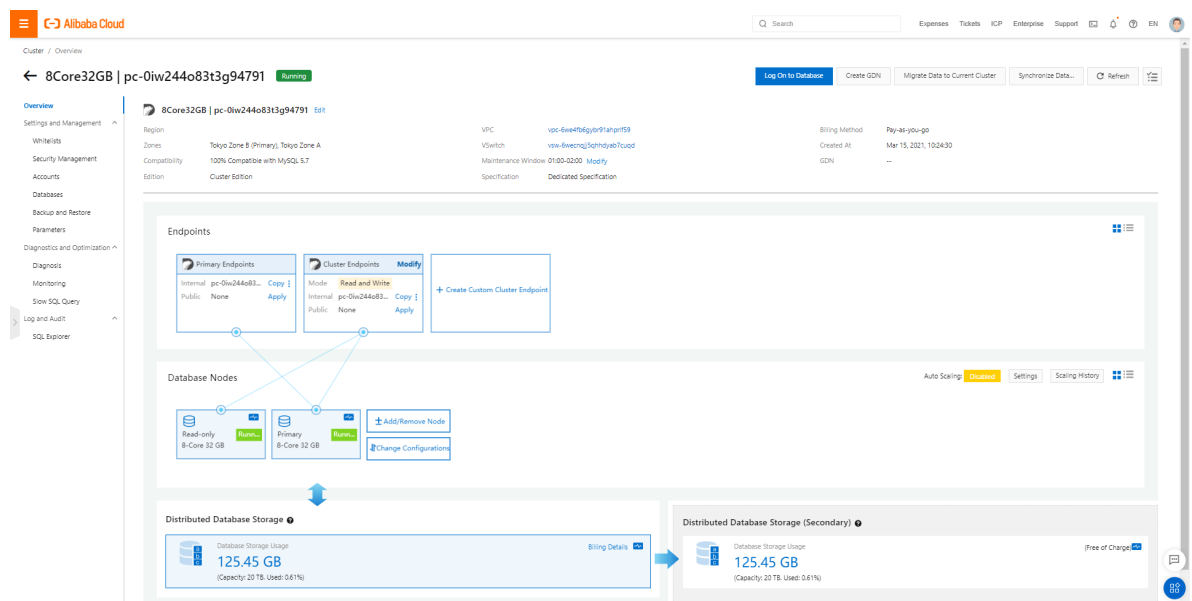
#### ④Nodeを切り替え中で、Script実行状態を確認する

```
1615793512
FUNC_DO_STH_AND_RET_CURTS(7,7,7,7)
1615793512
FUNC_DO_STH_AND_RET_CURTS(6,6,6,6)
1615793512
FUNC_DO_STH_AND_RET_CURTS(6,6,6,6)
1615793512
FUNC_DO_STH_AND_RET_CURTS(6,6,6,6)
1615793512
FUNC_DO_STH_AND_RET_CURTS(10,10,10,10)
1615793512
FUNC_DO_STH_AND_RET_CURTS(8,8,8,8)
1615793512
FUNC_DO_STH_AND_RET_CURTS(7,7,7,7)
1615793512
FUNC_DO_STH_AND_RET_CURTS(7,7,7,7)
1615793512
FUNC_DO_STH_AND_RET_CURTS(9,9,9,9)
1615793512
FUNC_DO_STH_AND_RET_CURTS(1,1,1,1)
1615793512
FUNC_DO_STH_AND_RET_CURTS(6,6,6,6)
1615793512
FUNC_DO_STH_AND_RET_CURTS(9,9,9,9)
1615793512
FUNC_DO_STH_AND_RET_CURTS(10,10,10,10)
1615793512
TIMEOUT OCCURRED! EVENT: 10517
TIMEOUT OCCURRED! EVENT: 10515
TIMEOUT OCCURRED! EVENT: 10521
TIMEOUT OCCURRED! EVENT: 10513
TIMEOUT OCCURRED! EVENT: 10520
TIMEOUT OCCURRED! EVENT: 10516
TIMEOUT OCCURRED! EVENT: 10518
TIMEOUT OCCURRED! EVENT: 10514
TIMEOUT OCCURRED! EVENT: 10522
TIMEOUT OCCURRED! EVENT: 10523
TIMEOUT OCCURRED! EVENT: 10521
TIMEOUT OCCURRED! EVENT: 10510
TIMEOUT OCCURRED! EVENT: 10520
TIMEOUT OCCURRED! EVENT: 10513
TIMEOUT OCCURRED! EVENT: 10517
TIMEOUT OCCURRED! EVENT: 10518
TIMEOUT OCCURRED! EVENT: 10516
TIMEOUT OCCURRED! EVENT: 10522
TIMEOUT OCCURRED! EVENT: 10524
TIMEOUT OCCURRED! EVENT: 10523
TIMEOUT OCCURRED! EVENT: 10517
TIMEOUT OCCURRED! EVENT: 10520
TIMEOUT OCCURRED! EVENT: 10521
TIMEOUT OCCURRED! EVENT: 10515
TIMEOUT OCCURRED! EVENT: 10525
TIMEOUT OCCURRED! EVENT: 10518
TIMEOUT OCCURRED! EVENT: 10522
TIMEOUT OCCURRED! EVENT: 10526
TIMEOUT OCCURRED! EVENT: 10524
TIMEOUT OCCURRED! EVENT: 10523
```

```
1615793558
FUNC_DO_STH_AND_RET_CURTS(3,3,3,3)
1615793558
FUNC_DO_STH_AND_RET_CURTS(1,1,1,1)
1615793558
FUNC_DO_STH_AND_RET_CURTS(2,2,2,2)
1615793558
FUNC_DO_STH_AND_RET_CURTS(8,8,8,8)
1615793558
FUNC_DO_STH_AND_RET_CURTS(5,5,5,5)
1615793558
FUNC_DO_STH_AND_RET_CURTS(9,9,9,9)
1615793558
FUNC_DO_STH_AND_RET_CURTS(6,6,6,6)
1615793558
FUNC_DO_STH_AND_RET_CURTS(10,10,10,10)
1615793558
FUNC_DO_STH_AND_RET_CURTS(4,4,4,4)
1615793558
FUNC_DO_STH_AND_RET_CURTS(2,2,2,2)
1615793558
FUNC_DO_STH_AND_RET_CURTS(8,8,8,8)
1615793558
FUNC_DO_STH_AND_RET_CURTS(5,5,5,5)
1615793558
FUNC_DO_STH_AND_RET_CURTS(3,3,3,3)
1615793558
FUNC_DO_STH_AND_RET_CURTS(9,9,9,9)
1615793558
FUNC_DO_STH_AND_RET_CURTS(6,6,6,6)
1615793558
FUNC_DO_STH_AND_RET_CURTS(7,7,7,7)
1615793558
FUNC_DO_STH_AND_RET_CURTS(10,10,10,10)
1615793558
FUNC_DO_STH_AND_RET_CURTS(1,1,1,1)
1615793558
FUNC_DO_STH_AND_RET_CURTS(8,9,9,9)
1615793558
FUNC_DO_STH_AND_RET_CURTS(3,3,3,3)
1615793558
FUNC_DO_STH_AND_RET_CURTS(4,4,4,4)
1615793558
FUNC_DO_STH_AND_RET_CURTS(5,5,5,5)
1615793558
FUNC_DO_STH_AND_RET_CURTS(6,6,6,6)
1615793558
FUNC_DO_STH_AND_RET_CURTS(7,7,7,7)
1615793558
FUNC_DO_STH_AND_RET_CURTS(10,10,10,10)
1615793558
FUNC_DO_STH_AND_RET_CURTS(8,8,8,8)
1615793558
FUNC_DO_STH_AND_RET_CURTS(2,2,2,2)
1615793558
FUNC_DO_STH_AND_RET_CURTS(1,1,1,1)
1615793558
```

## 3-3 FailOverを確認する

#### ⑤Nodeを切り替え完了、Script実行を確認する



```

FUNC_DO_STH_AND_RET_CURTS(6,6,6,6)
1615793569
FUNC_DO_STH_AND_RET_CURTS(6,6,6,6)
1615793569
FUNC_DO_STH_AND_RET_CURTS(8,8,8,8)
1615793569
FUNC_DO_STH_AND_RET_CURTS(8,8,8,8)
1615793569
FUNC_DO_STH_AND_RET_CURTS(9,9,9,9)
1615793569
FUNC_DO_STH_AND_RET_CURTS(5,5,5,5)
1615793569
FUNC_DO_STH_AND_RET_CURTS(9,9,9,9)
1615793569
FUNC_DO_STH_AND_RET_CURTS(9,9,9,9)
1615793569
FUNC_DO_STH_AND_RET_CURTS(7,7,7,7)
1615793569
FUNC_DO_STH_AND_RET_CURTS(9,9,9,9)
1615793569
FUNC_DO_STH_AND_RET_CURTS(10,10,10,10)
1615793569
FUNC_DO_STH_AND_RET_CURTS(9,9,9,9)
1615793569
FUNC_DO_STH_AND_RET_CURTS(7,7,7,7)
1615793569
FUNC_DO_STH_AND_RET_CURTS(6,6,6,6)
1615793569
FUNC_DO_STH_AND_RET_CURTS(10,10,10,10)
1615793569
FUNC_DO_STH_AND_RET_CURTS(10,10,10,10)
1615793569
FUNC_DO_STH_AND_RET_CURTS(10,10,10,10)
1615793569
FUNC_DO_STH_AND_RET_CURTS(8,8,8,8)
1615793569
FUNC_DO_STH_AND_RET_CURTS(7,7,7,7)
1615793569
FUNC_DO_STH_AND_RET_CURTS(8,8,8,8)
1615793569
FUNC_DO_STH_AND_RET_CURTS(10,10,10,10)
1615793569
FUNC_DO_STH_AND_RET_CURTS(8,8,8,8)
1615793569
FUNC_DO_STH_AND_RET_CURTS(9,9,9,9)
1615793569
FUNC_DO_STH_AND_RET_CURTS(9,9,9,9)
1615793569
FUNC_DO_STH_AND_RET_CURTS(9,9,9,9)
1615793569
FUNC_DO_STH_AND_RET_CURTS(10,10,10,10)
1615793569
FUNC_DO_STH_AND_RET_CURTS(10,10,10,10)
1615793569
FUNC_DO_STH_AND_RET_CURTS(10,10,10,10)
1615793569
ALL DONE

```

⑥FailOver計算する

```
# mv nohup.out Mysql5.7_8Core32GB_result_event_2000_concurrent_10.log
```

```

[1]+  Done                  nohup sh run-fot.sh 20000 10
[root@iz6wega8j6xeq3boflbonsZ 8Core32GB]# ll
total 9080
-rwxr-xr-x 1 root root    407 Mar 15 13:37 fot.sh
-rw----- 1 root root 9278657 Mar 15 15:34 nohup.out
drwxr-xr-x 3 root root   4096 Mar 15 11:01 Prepare
-rwxr-xr-x 1 root root    223 Mar 15 13:37 run-fot.sh
-rwxr-xr-x 1 root root   3537 Mar 15 13:37 SetupMySQLFailOverTest.sql
[root@iz6wega8j6xeq3boflbonsZ 8Core32GB]# mv nohup.out Mysql5.7_8Core32GB_result_event_2000_concurrent_10.log
[root@iz6wega8j6xeq3boflbonsZ 8Core32GB]# ll
total 9080
-rwxr-xr-x 1 root root    407 Mar 15 13:37 fot.sh
-rw----- 1 root root 9278657 Mar 15 15:34 Mysql5.7_8Core32GB_result_event_2000_concurrent_10.log
drwxr-xr-x 3 root root   4096 Mar 15 11:01 Prepare
-rwxr-xr-x 1 root root    223 Mar 15 13:37 run-fot.sh
-rwxr-xr-x 1 root root   3537 Mar 15 13:37 SetupMySQLFailOverTest.sql
[root@iz6wega8j6xeq3boflbonsZ 8Core32GB]#

```

```

[root@iz6wega8j6xeq3boflbonsZ 8Core32GB]# grep -C 4 TIMEOUT
Mysql5.7_8Core32GB_result_event_2000_concurrent_10.log | head -n 6
FUNC_DO_STH_AND_RET_CURTS(9,9,9,9)
1615793512
FUNC_DO_STH_AND_RET_CURTS(10,10,10,10)
1615793512
TIMEOUT OCCURRED! EVENT: 10517
TIMEOUT OCCURRED! EVENT: 10515

```

```

[root@iz6wega8j6xeq3boflbonsZ 8Core32GB]# grep -C 4 TIMEOUT
Mysql5.7_8Core32GB_result_event_2000_concurrent_10.log | tail -n 6
TIMEOUT OCCURRED! EVENT: 10524
TIMEOUT OCCURRED! EVENT: 10523
FUNC_DO_STH_AND_RET_CURTS(1,1,1,1)
1615793530
FUNC_DO_STH_AND_RET_CURTS(3,3,3,3)
1615793530

```

```

# bc
# 1615793530-1615793512
18

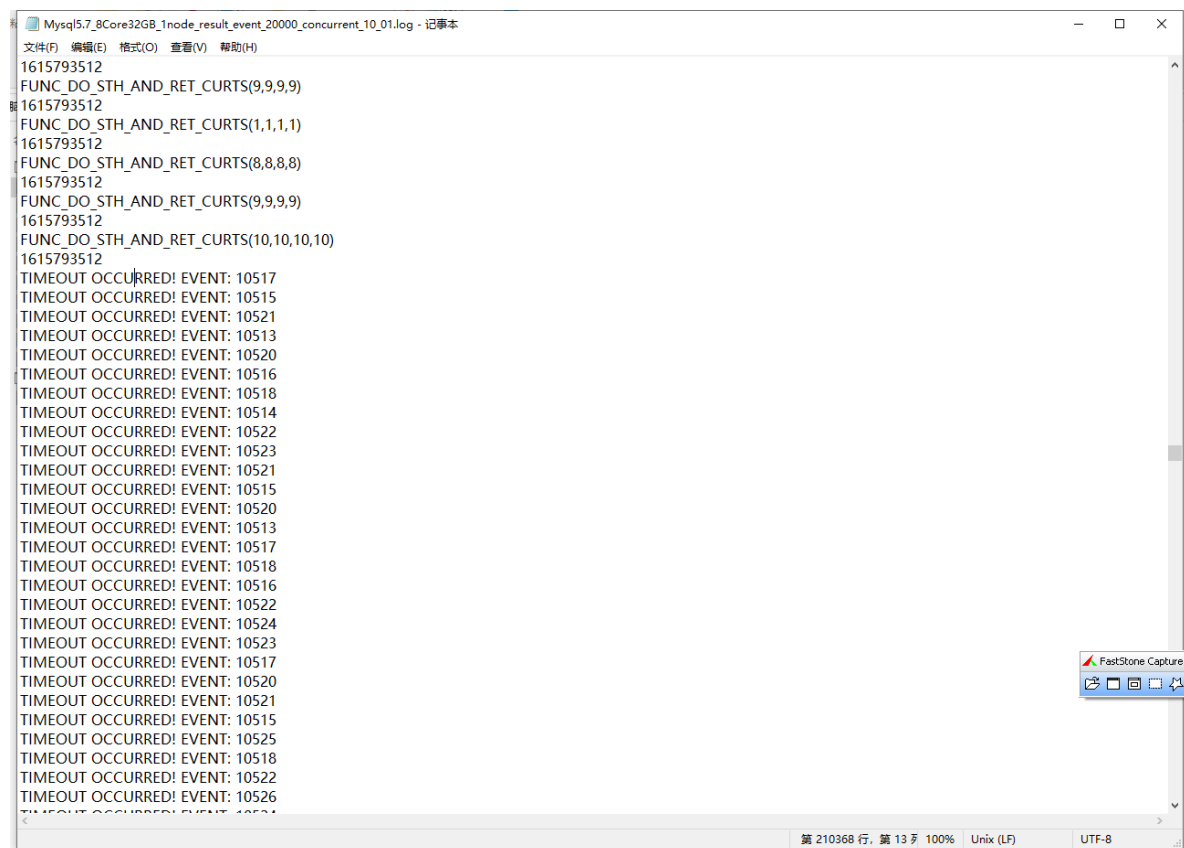
```

```

[root@iZ6wega8j6xeq3boflbonsZ 8Core32GB]# ll
total 9080
-rwxr-xr-x 1 root root 407 Mar 15 13:37 fot.sh
-rw-r--r-- 1 root root 9278657 Mar 15 15:34 Mysql5.7_8Core32GB_result_event_2000_concurrent_10.log
drwxr-xr-x 3 root root 4096 Mar 15 11:01 Prepare
-rwxr-xr-x 1 root root 223 Mar 15 13:37 run-fot.sh
-rwxr-xr-x 1 root root 3537 Mar 15 13:37 SetupMySQLFailoverTest.sql
[root@iZ6wega8j6xeq3boflbonsZ 8Core32GB]# grep -C 4 TIMEOUT Mysql5.7_8Core32GB_result_event_2000_concurrent_10.log | head -n 6
FUNC_DO_STH_AND_RET_CURTS(9,9,9,9)
1615793512
FUNC_DO_STH_AND_RET_CURTS(10,10,10,10)
1615793512
TIMEOUT OCCURRED! EVENT: 10517
TIMEOUT OCCURRED! EVENT: 10515
[root@iZ6wega8j6xeq3boflbonsZ 8Core32GB]# grep -C 4 TIMEOUT Mysql5.7_8Core32GB_result_event_2000_concurrent_10.log | tail -n 6
TIMEOUT OCCURRED! EVENT: 10524
TIMEOUT OCCURRED! EVENT: 10523
FUNC_DO_STH_AND_RET_CURTS(1,1,1,1)
1615793530
FUNC_DO_STH_AND_RET_CURTS(3,3,3,3)
1615793530
[root@iZ6wega8j6xeq3boflbonsZ 8Core32GB]# bc
bc 1.06.95
Copyright 1991-1994, 1997, 1998, 2000, 2004, 2006 Free Software Foundation, Inc.
This is free software with ABSOLUTELY NO WARRANTY.
For details type `warranty'.
1615793530-1615793512
18

```

Failover タイムは18秒である、同じ手順で5回Scriptを実行し平均値を計算する



```

Mysql5.7_8Core32GB_1node_result_event_20000_concurrent_10_01.log - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
1615793512
FUNC_DO_STH_AND_RET_CURTS(9,9,9,9)
1615793512
FUNC_DO_STH_AND_RET_CURTS(1,1,1,1)
1615793512
FUNC_DO_STH_AND_RET_CURTS(8,8,8,8)
1615793512
FUNC_DO_STH_AND_RET_CURTS(9,9,9,9)
1615793512
FUNC_DO_STH_AND_RET_CURTS(10,10,10,10)
1615793512
TIMEOUT OCCURRED! EVENT: 10517
TIMEOUT OCCURRED! EVENT: 10515
TIMEOUT OCCURRED! EVENT: 10521
TIMEOUT OCCURRED! EVENT: 10513
TIMEOUT OCCURRED! EVENT: 10520
TIMEOUT OCCURRED! EVENT: 10516
TIMEOUT OCCURRED! EVENT: 10518
TIMEOUT OCCURRED! EVENT: 10514
TIMEOUT OCCURRED! EVENT: 10522
TIMEOUT OCCURRED! EVENT: 10523
TIMEOUT OCCURRED! EVENT: 10521
TIMEOUT OCCURRED! EVENT: 10515
TIMEOUT OCCURRED! EVENT: 10520
TIMEOUT OCCURRED! EVENT: 10513
TIMEOUT OCCURRED! EVENT: 10517
TIMEOUT OCCURRED! EVENT: 10518
TIMEOUT OCCURRED! EVENT: 10516
TIMEOUT OCCURRED! EVENT: 10522
TIMEOUT OCCURRED! EVENT: 10524
TIMEOUT OCCURRED! EVENT: 10523
TIMEOUT OCCURRED! EVENT: 10517
TIMEOUT OCCURRED! EVENT: 10520
TIMEOUT OCCURRED! EVENT: 10521
TIMEOUT OCCURRED! EVENT: 10515
TIMEOUT OCCURRED! EVENT: 10525
TIMEOUT OCCURRED! EVENT: 10518
TIMEOUT OCCURRED! EVENT: 10522
TIMEOUT OCCURRED! EVENT: 10526
TIMEOUT OCCURRED! EVENT: 10524

```

以上です