

PostgreSQL 性能测试

1. 下载 TPC-H

http://www.tpc.org/tpc_documents_current_versions/download_programs/tools-download-request.asp?BM=TPC-H&version=2.17.1&mode=CURRENT-ONLY

打开网站，填写信息后，可以下载 TPC-H Tools

2. 安装 TCP-H

解压缩下载的 zip 包，进入目录 tpch_2_17_0

```
$unzip TPCH_Tools.zip
$cd tpch_2_17_0/
```

3. 设置 makefile

进入目录 dbgen，拷贝 makefile.suite，设置参数 CC，DATABASE，MACHINE 和 WORKLOAD。

```
$cd dbgen
$cp makefile.suite Makefile
$vi Makefile

#####
## CHANGE NAME OF ANSI COMPILER HERE
#####

CC      = gcc

# Current values for DATABASE are: INFORMIX, DB2, TDAT (Teradata)
#
#                                SQLSERVER, SYBASE, ORACLE, VECTORWISE
# Current values for MACHINE are: ATT, DOS, HP, IBM, ICL, MVS,
#
#                                SGI, SUN, U2200, VMS, LINUX, WIN32
# Current values for WORKLOAD are: TPCH

DATABASE=ORACLE

MACHINE = LINUX

WORKLOAD = TPCH
```

修改完后 make。

4. 生成测试数据

执行命令：`./dbgen -s 100 -f`

-s 表示 scale（单位为 GB），生成 8 张表的数据。

```

$./dbgen -s 100 -f
$ll -h *.tbl
-rw-r--r-- 1 digoad users 12G Nov 19 13:35 customer.tbl
-rw-r--r-- 1 digoad users 6.6G Nov 19 13:39 lineitem.tbl
-rw-r--r-- 1 digoad users 2.2K Nov 19 12:48 nation.tbl
-rw-r--r-- 1 digoad users 1.5G Nov 19 13:39 orders.tbl
-rw-r--r-- 1 digoad users 1.2G Nov 19 12:48 partsupp.tbl
-rw-r--r-- 1 digoad users 233M Nov 19 12:48 part.tbl
-rw-r--r-- 1 digoad users 389 Nov 19 12:48 region.tbl
-rw-r--r-- 1 digoad users 686M Nov 19 13:30 supplier.tbl

```

5. 修改数据格式

生成的数据末尾多了一个'|'，不符合 postgresql 数据导入格式，故需要修改数据格式。生成新的数据“.csv”个数。

```
for i in `ls *.tbl`; do sed 's/|$//' $i > ${i/tbl/csv}; done
```

```

$for i in `ls *.tbl`; do sed 's/|$//' $i > ${i/tbl/csv}; done

$ll -rth *.csv
-rw-r--r-- 1 digoad users 12G Dec 1 13:32 customer.csv
-rw-r--r-- 1 digoad users 6.5G Dec 1 13:33 lineitem.csv
-rw-r--r-- 1 digoad users 2.2K Dec 1 13:33 nation.csv
-rw-r--r-- 1 digoad users 1.5G Dec 1 13:34 orders.csv
-rw-r--r-- 1 digoad users 1.2G Dec 1 13:34 partsupp.csv
-rw-r--r-- 1 digoad users 231M Dec 1 13:34 part.csv
-rw-r--r-- 1 digoad users 384 Dec 1 13:34 region.csv
-rw-r--r-- 1 digoad users 681M Dec 1 13:34 supplier.csv

```

6. 软链接

把包含 csv 文件的目录，软链接到/tmp/dss-data。 tpch-pg 脚本中一会要用到这个目录。

```
ln -s /home/beaver/Documents/tpch_2_17_0/dbgen /tmp/dss-data
```

7. 下载 pg_tpch

解压缩下载的 zip 包，进入目录 dss。

```
wget https://github.com/tvondra/pg_tpch/archive/master.zip
unzip master.zip
cd pg_tpch-master/
```

修改 tpch.sh , 去掉第一行“#!/bin/bash”; 将变量 DSS_TIMEOUT 的值设大 (3600)。

进入目录 dss , 将所有 copy 语句修改成 (共 8 句)

```
\copy part from '/tmp/dss-data/part.csv' delimiter '|';
```

将 pg_tpch 的文件都拷贝到 tpch_2_17_0/dbgen 所在的目录 :

```
cp -r pg_tpch-master/* tpch/tpch_2_17_0/dbgen
```

8. 生成 queries

进入目录 tpch_2_17_0/dbgen/dss , 创建目录 queries。

```
cd tpch/tpch_2_17_0/dbgen/dss
mkdir queries
```

然后 , 将文件 dbgen/ dists.dss 拷贝到目录 dss。

```
cp ../dists.dss ./
```

最后 , 生成测试 SQL (在目录 dbgen 下执行) :

```
for q in `seq 1 22`
do
DSS_QUERY=dss/templates ./qgen $q >> dss/queries/$q.sql
sed 's/^select/explain select/' dss/queries/$q.sql > dss/queries/$q.explain.sql
cat dss/queries/$q.sql >> dss/queries/$q.explain.sql;
done
```

9. 创建数据库和用户 (已创建可跳过此步)

```
create role beaver login;
create database tpch owner beaver;
```

注 : 配置 pg_hba.conf , 将 md5 改为 trust。

10. 测试

在 dbgen 目录下执行

```
./tpch.sh ./results tpch beaver
```

使用 beaver 用户连接到 tpch 数据库 , 结果输出到 ./results 目录 , 自动创建表 ,

加载数据。

11. 结果

运行结果全部在目录 results 下，bench.log 记录了每个 query 的执行时间，results/results 和 results/explain 记录了每个 query 的查询结果和查询计划。

参考网址：https://github.com/digoal/pg_tpch