TPCC 日常daily基准测试自动化代码说明文档

# 目录介绍：

performance\_tpcc\_everyday

│

├── Config文件夹：配置文件

│

├── Doc 文件夹：TPCC的数据模板

│

├── Html文件夹：每日的测试数据和html文件

│

├── Lib文件夹： perfermance的类

│

├── Log文件夹： log日志

│

├── Packages文件夹： rpm安装包

│

├── Result 文件夹：每日TPCC的日志和执行计划等

│

├── Tool 文件夹： TPCC的运行程序

│

└── Performance\_tpcc\_main.py: 自动化主函数入口

# 2.Config文件夹

## 2.1 data\_config 文件：

存放数据库配置参数：

logging\_collector = on

log\_directory = 'log'

shared\_buffers = 120GB

fsync = off

full\_page\_writes = off

track\_activities = off

track\_counts = off

update\_process\_title = off

max\_connections = 1200

synchronous\_commit = off

effective\_cache\_size = 400GB

enable\_shortcut = on

listen\_addresses = '\*'

## 2.2 sys\_config.ini文件：

存放自动化配置参数：

[perfoermance]

perfoermance\_tpcc\_path = /opt/performance/performance\_tpcc\_everyday

[sshConnection]

ssh\_ip = 10.15.40.37

ssh\_port = 22

ssh\_name = root

ssh\_passwd = esg123.com

[database]

data\_path = /data/qianbasetp

license = False

#version\_date = 20230904

version\_date = 20230916

package\_url = http://10.14.40.24/qianbasetp/daily/{}/15/x86\_64/

package\_name = QianBaseTP-15.0-1.el7.x86\_64.{}.rpm

package\_md5\_name = QianBaseTP-15.0-1.el7.x86\_64.{}.rpm.md5

mail\_list=xin.zhao@esgyn.cn

[tpcc]

#unlog/log

tableCreates = "tableCreatesUnlog"

conn=jdbc:postgresql://10.15.40.37:20158/tpcc

user=qianbase

password=qianbase

warehouses=10

loadWorkers=100

terminals=100

runMins=5

tpcc\_database = tpcc

[tpcc\_run]

#threads = ("1" "50" "100" "120" "150" "180" "200" "300" "500" "1000")

threads = ("1" "20" "50")

run\_number = 5

result\_directory = /opt/performance/performance\_tpcc\_everyday/result/{}/unlog\_max\_performance\_daily

result\_name\_flag = unlog\_max\_performance\_daily

data\_tpcc\_source = /data/qianbasetp\_back

run\_preheat = false

2.3 Modules to be installed需要安装的模块文件：

存放需要提前安装的模块：

pip3 install -U pip setuptools -i https://pypi.tuna.tsinghua.edu.cn/simple

pip3 install paramiko -i https://pypi.tuna.tsinghua.edu.cn/simple

pip3 install wget -i https://pypi.tuna.tsinghua.edu.cn/simple

pip3 install openpyxl -i https://pypi.tuna.tsinghua.edu.cn/simple

pip3 install pandas -i https://pypi.tuna.tsinghua.edu.cn/simple

pip3 install schedule -i https://pypi.tuna.tsinghua.edu.cn/simple

pip3 install xlrd==1.2.0 -i https://pypi.tuna.tsinghua.edu.cn/simple

2.4 sysctl\_config 用于存放操作系统参数的文件：

#TPCC

kernel.shmmax = 500000000000

kernel.shmmni = 32768

kernel.shmall = 40000000000

kernel.sem = 1000 32768000 1000 32768

kernel.sysrq = 1

kernel.core\_uses\_pid = 1

kernel.msgmnb = 1048576

kernel.msgmax = 1048576

kernel.msgmni = 32768

net.ipv4.tcp\_syncookies = 1

net.ipv4.conf.default.accept\_source\_route = 0

net.ipv4.tcp\_max\_syn\_backlog = 32768

net.ipv4.tcp\_syn\_retries = 3

net.ipv4.conf.all.arp\_filter = 1

net.ipv4.ip\_local\_port\_range = 1025 65535

net.ipv4.ip\_local\_reserved\_ports = 5432,40000-40127,41000-41127,50000-50127,51000-51127

net.ipv6.conf.all.disable\_ipv6 = 1

net.ipv6.conf.default.disable\_ipv6 = 1

net.core.netdev\_max\_backlog = 80000

net.core.rmem\_default = 2097152

net.core.rmem\_max = 16777216

net.core.wmem\_max = 16777216

vm.overcommit\_memory = 2

vm.overcommit\_ratio = 95

vm.swappiness = 0

vm.zone\_reclaim\_mode = 0

vm.dirty\_expire\_centisecs = 200

vm.dirty\_writeback\_centisecs = 100

vm.dirty\_background\_ratio = 5

vm.dirty\_ratio = 10

net.core.somaxconn = 65535

net.ipv4.ipfrag\_low\_thresh=31457280

net.ipv4.ipfrag\_high\_thresh=33554430

net.ipv4.ipfrag\_max\_dist=96

#fix Unable to start qbfdist on port 4007

net.ipv4.tcp\_syncookies = 1

net.ipv4.tcp\_tw\_reuse = 1

#net.ipv4.tcp\_tw\_recycle = 1

net.ipv4.tcp\_fin\_timeout = 60

# 3. doc文件夹：

存放了tpcc的excel模板：

《performance\_tpcc\_all\_moban.xlsx》、《performance\_tpcc\_moban.xlsx》

# 4. html文件夹：

文本

描述已自动生成

## 4.1 生成tpcc所有测试结果的excel：

《performance\_tpcc\_all.xlsx》

## 4.2 每日测试结果：

生成按日期命名的文件夹和excel

# 5. lib文件夹：

## 5.1 dataBaseOption.py：

**DataBaseOption 类**：该类用于执行数据库操作。

**属性:**

ssh：SSH 连接对象列表，用于与远程主机进行通信。

**方法:**

database\_qboption(option)

描述：执行数据库操作。

参数：

option：要执行的数据库操作选项。

返回值：布尔值，表示操作执行的结果。

## 5.2 downLoad.py：

**DownLoad 类：**该类用于下载和传输软件包

**属性:**

package\_dir：软件包目录的路径。

package\_url：软件包的基本 URL。

package\_name：软件包的名称模板。

package\_md5\_name：软件包 MD5 校验文件的名称模板。

**方法:**

download\_daily\_package()

描述：下载每日软件包并进行 MD5 校验。

参数：无。

返回值：布尔值，表示软件包的校验结果。

verify\_package\_md5(package\_path, package\_md5\_path)

描述：校验软件包的 MD5 值。

参数：

package\_path：软件包文件的路径。

package\_md5\_path：软件包 MD5 校验文件的路径。

返回值：布尔值，表示校验结果。

create\_remote\_dir(ssh, path)

描述：在远程主机上创建目录。

参数：

ssh：SSH 连接对象。

path：要创建的远程目录路径。

返回值：布尔值，表示目录创建结果。

scp\_package(ssh, package\_path)

描述：将软件包传输到远程主机。

参数：

ssh：SSH 连接对象。

package\_path：要传输的软件包文件的路径。

返回值：布尔值，表示传输结果。

scp\_packages(ssh\_list)

描述：在给定的 SSH 连接列表中传输软件包。

参数：ssh\_list：包含 SSH 连接对象的列表。

返回值：布尔值，表示传输结果。

## 5.3 installPackage.py:

**InstallPackage 类:** 该类用于安装数据库和执行相关配置操作。

**属性:**

ssh\_all：SSH 连接对象列表，用于与远程主机进行通信。

master\_path：主数据库路径。

primary\_path：从数据库路径。

config\_dir：配置文件存储目录。

ssh\_segment：除主数据库外的其他数据库的 SSH 连接对象列表。

**方法:**

check\_datebase()

描述：检查并卸载已安装的数据库软件。

返回值：布尔值，表示操作执行的结果。

install\_package()

描述：安装数据库软件包。

返回值：布尔值，表示操作执行的结果。

cleanup\_datebase()

描述：清理数据库。

返回值：布尔值，表示操作执行的结果。

描述：写入 .bashrc 配置文件。

返回值：无。

install\_database()

描述：安装数据库并进行初始化操作。

返回值：布尔值，表示操作执行的结果。

## 5.4 Logger.py:

**Logger 类:** 实现了一个日志记录器（Logger）类，用于记录程序运行过程中的日志信息。

**属性:**

out：日志输出文件路径。

logger：日志记录器对象。

**方法:**

\_\_init\_\_()

描述：初始化 Logger 类。

参数：无。

返回值：无。

info(message='', s=1, \*\*kwargs)

描述：记录一条信息级别的日志。

参数：

message：要记录的日志消息。

s：指定是否对消息进行格式化，默认为 1，表示需要格式化。

\*\*kwargs：其他可选参数。

返回值：无。

warning(string='')

描述：记录一条警告级别的日志。

参数：

string：要记录的警告消息。

返回值：无。

error(string='')

描述：记录一条错误级别的日志。

参数：

string：要记录的错误消息。

返回值：无。

get\_file\_handler()

描述：返回一个将日志记录到文件中的处理器（FileHandler）。

参数：无。

返回值：日志文件处理器对象。

get\_logger(logger\_name)

描述：返回一个新的带有文件处理器的日志记录器实例。

参数：

logger\_name：日志记录器的名称。

返回值：新的日志记录器对象。

## 5.5 tpInfo.py

**tpInfo 类**: 定义了一个名为 tpInfo 的类，用于读取和存储系统配置信息。

**属性:**

    ssh\_ip=None

    ssh\_port=None

    ssh\_name=None

    ssh\_passwd = None

    data\_path=None

    tableCreates = None

    version\_date = None

    package\_url = None

    package\_name = None

    package\_md5\_name = None

**方法:**

\_\_init\_\_()

描述：初始化 tpInfo 类，读取系统配置文件并将配置信息存储到对应的属性中。

参数：无。

返回值：无。

## 5.6 tpQconfig.py:

**tpQconfig 类**: 定义了一个名为 tpQconfig 的类，用于执行一系列与数据库配置相关的操作。

**属性:**

ssh：SSH连接对象，用于与远程服务器进行通信。

path：节点的路径。

data\_config\_path：数据配置文件的路径。

**方法:**

\_\_init\_\_(self, ssh\_all, master\_path)

描述：初始化 tpQconfig 类，设置 SSH连接对象和主节点路径。

参数：

ssh\_all：SSH连接对象。

返回值：无。

getConfig(self, path\_config)

描述：读取配置文件内容。

参数：

path\_config：配置文件的路径。

返回值：配置文件内容的字符串。

create\_database(self)

描述：创建数据库 qbadmin。

参数：无。

返回值：布尔值，表示数据库创建是否成功。

database\_qconfig(self)

描述：执行数据库配置操作。

参数：无。

返回值：布尔值，表示数据库配置是否成功。

append\_qb\_hba\_conf(self)

描述：向 qb\_hba.conf 文件中追加配置。

参数：无。

返回值：无。

Replace\_licese(self):

描述：判断是否需要替换licence。

参数：无。

返回值：布尔值，表示替换成功。

database\_config(self)

描述：执行数据库配置操作的整体流程。

参数：无。

返回值：布尔值，表示数据库配置是否成功。

5.6 resultProcess.py:

ResultProcess类: 用于处理TPCC测试结果

\_\_init\_\_(self):

初始化ResultProcess类的实例。设置了日期、TPCC日志路径、TPCC Excel路径以及TPCC模板文件的路径。

read\_tpcds\_log\_list(self):

读取TPCC日志列表文件，并返回其内容。

tpcds\_result(self, log\_file):

根据给定的日志文件，提取TPCC测试结果中的数据加载、数据生成、查询耗时和分析耗时等信息，并返回这些信息。

performance\_tpcds\_excel(self, log\_files):

根据给定的TPCC日志文件列表，生成一个Excel文件，其中包含了各项测试指标的数据，并计算了平均值。

save\_as\_mht(self, excel\_file):

将给定的Excel文件转换为HTML格式，并保存为.mht文件。

performance\_tpcds\_excel\_all(self, start\_test\_time, excel\_file):

将单个测试结果的Excel文件合并到总的测试结果Excel文件中，包括每次测试的总时间、加载时间、查询时间和平均值等信息。

## 5.7 sendMail.py

**SendMail 类**: 用于发送邮件，包含了发送邮件的各种功能。

**属性:**

smtpsrvr：SMTP服务器地址

smtpport：SMTP服务器端口

username：SMTP用户名

password：SMTP密码

sender：发件人邮箱地址

**方法:**

read\_excel\_file:

用于读取Excel文件。它首先通过os.path.join函数获取Excel文件的路径，然后使用pd.read\_excel函数读取Excel文件的内容，并将其转换为HTML格式。如果读取的行数小于等于6行，将整个表格内容转换为HTML并返回；否则，只返回最后五行的HTML格式。

read\_data\_config :

用于读取"data\_config"文件的内容。它通过os.path.join函数获取文件路径，然后使用open函数打开文件并逐行读取文件内容。最后，将读取的内容以HTML格式返回。

make\_html\_content :

用于构建邮件的HTML内容。它使用了f-string的格式化字符串功能，将各个部分的内容拼接成完整的HTML字符串，并返回该字符串。

send :

用于发送邮件。它接受两个参数：excel\_file和excel\_file1，表示要作为附件发送的Excel文件路径。首先调用make\_html\_content方法生成HTML内容，然后使用smtplib库连接SMTP服务器，登录账户，创建邮件消息对象，并设置发件人、收件人、主题和正文内容。接着，读取附件文件，并将其添加到邮件消息对象中。最后，发送邮件并返回发送结果。

send\_error\_mail :

用于发送错误邮件。它接受一个参数filename，表示要作为附件发送的日志文件路径。首先构建邮件的HTML内容，然后使用smtplib库连接SMTP服务器，登录账户，创建邮件消息对象，并设置发件人、收件人、主题和正文内容。接着，读取附件文件，并将其添加到邮件消息对象中。最后，发送邮件并返回发送结果。

## 5.8 sshClient.py：

**sshClient 类**：这个类用于创建SSH连接。

**方法：**

Conn：

这个方法用于建立SSH连接。它接受四个参数：ip、username、password和port，表示要连接的SSH服务器的IP地址、用户名、密码和端口号。在方法内部，首先创建一个SSHClient对象，然后设置密钥策略为自动添加。接着，使用connect方法连接SSH服务器，并传入IP地址、端口号、用户名和密码。如果连接成功，返回SSHClient对象；如果连接失败，打印错误信息并返回False。

Close：

这个方法用于关闭SSH连接。它检查self.ssh属性是否存在，如果存在，则调用close方法关闭SSH连接。

## 5.9 toHtml.py

Class to\_html类：这个类用于将excel转化为html，给邮箱提供html格式的表格

# 6.log文件夹

用于存放5.4章节的Logger 类的日志

# 7. packages文件夹

用于存放QianbaseTP的rpm安装包

# 8. result文件夹

用于存放tpcc的运行日志和每次循环的执行日志等：

图形用户界面, 文本

描述已自动生成

# 9. tool文件夹：

用于存放tpcc运行程序，相关代码可以查看：

文本

描述已自动生成

10. perfermance\_main.py文件：

**Test 类**：性能测试的主要类，用于执行各个测试任务。

**方法：**

Init：

这个方法用于初始化Test类的实例。它接受一个参数ssh\_all，表示SSH连接对象的列表。在方法内部，将SSH连接对象列表、TPCC测试相关的路径、日期、数量、密码和段数存储为Test类实例的属性。

chown\_tpcc：

这个方法用于设置tpcds文件夹和文件的权限。首先构建两个命令字符串，一个用于修改文件夹和文件的权限，一个用于更改文件夹和文件的所有者。然后执行这两个命令。如果执行成功，返回True；否则返回False。

run\_tpcc：

这个方法用于运行TPCC测试。首先调用chown\_tpcds方法设置权限。然后构建运行TPCC的命令，并使用SSH连接对象执行该命令。在命令执行期间，实时读取命令输出并记录日志。当命令执行完成后，关闭标准输出和标准错误流，并记录日志。如果执行成功，返回True；否则返回False。

test\_main：

这个函数是性能测试的主函数。在函数内部，首先获取SSH连接对象，并将它们存储在ssh\_all列表中。然后进行一系列的测试任务，包括下载安装包、安装数据库、配置数据库参数、运行TPCC测试、处理结果和发送邮件。最后，返回生成的Excel文件路径。如果发生任何错误，记录错误信息并返回None。

定时任务设置：

在if \_\_name\_\_ == '\_\_main\_\_':条件下，设置了一个定时任务，每天的22:00执行test\_main函数。

循环执行任务和发送邮件：

在while True:循环中，通过schedule.run\_pending()来运行定时任务。然后调用test\_main函数执行测试任务，并根据返回的Excel文件路径发送邮件。如果执行成功，记录成功信息；否则记录错误信息并发送错误邮件。