

openGauss训练营

openGauss 实践总结

彭 冲 云和恩墨(PG技术顾问)



openGauss



Gauss松鼠会

01

安装常见问题

02

编译及插件问题

03

客户端连接问题

04

用户自定义函数

05

数据库兼容性

06

日期时间类型

07

权限配置

08

备份恢复

09

SSL验证模式

10

客户端工具

安装常见问题： python3版本



openGauss



Gauss松鼠会

□ Python3.6

□ Python3.7

```
/opt/python3/bin/python3  
>>> import ssl  
>>> import _ssl
```

上面两个import不报错则正常

□ Python3.8

openGauss安装脚本依赖python3，推荐使用3.7。

安装常见问题： root不能SSH互信



openGauss



Gauss松鼠会

□ 预安装

```
./gs_preinstall -U omm -G dbgrp \  
-X /opt/software/openGauss/clusterconfig.xml \  
-L
```

可以在各个节点root用户先本地预安装通过，预安装完成后再使用omm互信来进行安装。

安装常见问题： MTU设置问题



openGauss



Gauss松鼠会

ssh连接提示connection closed by XX

现象表现为： ping和telnet端口都正常，但使用sftp、 scp、 ssh命令连接对端机器时会卡住，最后超时后提示Connection closed by XX。

使用-v选项打印调试信息

```
debug1: SSH2_MSG_KEXINIT sent
debug1: SSH2_MSG_KEXINIT received
debug1: kex: server->client aes128-ctr hmac-md5-etm@openssh.com none
debug1: kex: client->server aes128-ctr hmac-md5-etm@openssh.com none
debug1: sending SSH2_MSG_KEX_ECDH_INIT
debug1: expecting SSH2_MSG_KEX_ECDH_REPLY
Connection closed by [REDACTED]
```

原因：测试环境Redhat7.6下修改了网卡MTU值，改为默认值1500后恢复正常。

`ifconfig eth0 mtu 8192`

□ xml数据类型

需要使用--with-libxml选项编译openGauss-server，同时需要安装依赖包libxml2-devel。

□ gsql使用历史命令

需要使用--with-readline选项编译openGauss-server，同时需要安装依赖包readline-devel。

□ make错误代码127

需要安装依赖包redhat-lsb-core和patch。

```
make -C smgr all
make[4]: Entering directory `/opt/openGauss-server/src/gausskernel/storage/sm
make[4]: Nothing to be done for `all'.
make[4]: Leaving directory `/opt/openGauss-server/src/gausskernel/storage/smg
make -C mot all
make[4]: Entering directory `/opt/openGauss-server/src/gausskernel/storage/mo
make -C core/src all
make[5]: Entering directory `/opt/openGauss-server/src/gausskernel/storage/mo
make[5]: *** [mot_target] Error 127
make[5]: Leaving directory `/opt/openGauss-server/src/gausskernel/storage/mot
make[4]: *** [core/src-recursive] Error 2
make[4]: Leaving directory `/opt/openGauss-server/src/gausskernel/storage/mot
make[3]: *** [mot-recursive] Error 2
make[3]: Leaving directory `/opt/openGauss-server/src/gausskernel/storage'
make[2]: *** [storage-recursive] Error 2
make[2]: Leaving directory `/opt/openGauss-server/src/gausskernel'
make[1]: *** [all-gausskernel-recursive] Error 2
make[1]: Leaving directory `/opt/openGauss-server/src'
make: *** [all-src-recursive] Error 2
```

外部表插件：mysql_fdw支持



openGauss



Gauss松鼠会

需要使用`--enable_mysql_fdw`选项编译openGauss-server，同时需要安装依赖包`mariadb-connector-c-devel`。

`mariadb-connector-c-devel-3.0.10-1.el7.x86_64.rpm`

```
# rpm -ql mariadb-connector-c-devel-3.0.10-1.el7 |grep mariadb_com.h
/usr/include/mysql/mariadb_com.h
```

```
# rpm -ivh MariaDB-common-5.5.68-1.el7.centos.x86_64.rpm
# rpm -ivh crypto-policies-20170816-1.git2618a6c.el7.noarch.rpm
# rpm -ivh openssl11-libs-1.1.0i-1.el7.x86_64.rpm
# rpm -ivh mariadb-connector-c-3.0.10-1.el7.x86_64.rpm
# rpm -ivh mariadb-connector-c-devel-3.0.10-1.el7.x86_64.rpm
```

```
mysql ((GaussDB Kernel V500R001C20 build ) compiled at 2021-09-08 07:40:49 commit 0 last mr )
Non-SSL connection (SSL connection is recommended when requiring high-security)
Type "help" for help.

postgres=> create extension mysql_fdw with schema public;
CREATE EXTENSION
postgres=> select mysql_fdw_version();
mysql_fdw_version
-----
20505
(1 row)
```

外部表插件： oracle_fdw支持



openGauss



Gauss松鼠会

需要使用--enable_oracle_fdw选项编译openGauss-server，同时需要安装oracle客户端依赖包，并配置动态库环境变量。

```
# yum install oracle-instantclient11.2-basic-11.2.0.4.0-1.x86_64.rpm
# yum install oracle-instantclient11.2-devel-11.2.0.4.0-1.x86_64.rpm
```

```
# vi /etc/ld.so.conf
include ld.so.conf.d/*.conf
/usr/lib/oracle/11.2/client64/lib

# ldconfig
```

```
postgres=> create extension oracle_fdw with schema public;
CREATE EXTENSION
postgres=> select oracle_diag();
               oracle_diag
-----
oracle_fdw 2.2.0, PostgreSQL 9.2.4, Oracle client 11.2.0.4.0
(1 row)
```


postgis插件编译



openGauss



Gauss松鼠会

下载postgis及相关依赖包

```
cd /opt/openGauss-server/third_party/dependency/postgis
wget https://opengauss.obs.cn-south-1.myhuaweicloud.com/dependency/postgis-xc-master-2020-09-17.tar.gz
tar zxvf postgis-xc-master-2020-09-17.tar.gz
mv postgis-xc-master postgis-xc
```

先编译安装postgis依赖包

```
cd postgis-xc/proj-4.9.2
chmod +x ./configure
./configure --prefix=$GAUSSHOME/pggis_tools/proj
make && make install
ldd $GAUSSHOME/pggis_tools/proj/lib/libproj.so
```

```
cd ../json-c-json-c-0.12.1-20160607
chmod +x ./configure
./configure --prefix=$GAUSSHOME/pggis_tools/json
make && make install
```

```
cd ../geos-3.6.2
chmod +x ./configure
./configure --prefix=$GAUSSHOME/pggis_tools/geos
make && make install
ldd $GAUSSHOME/pggis_tools/geos/lib/libgeos.so
```

```
cd ../libxml2-2.7.1
chmod +x ./configure
./configure --prefix=$GAUSSHOME/pggis_tools/libxml2
make && make install
ldd /opt/og/pggis_tools/libxml2/lib/libxml2.so
```

应用patch包，否则编译postgis会报错

```
cd $CODE_BASE
cd /opt/openGauss-server/
patch -p1 < third_party/dependency/postgis/postgis.patch
```

```
rm -f /usr/local/lib/liblwgeom-2.4.so.0
```

```
rm -f /usr/local/lib/liblwgeom.so
```

```
visudo -f /etc/sudoers
```

```
omm ALL=(ALL) NOPASSWD:ALL
```

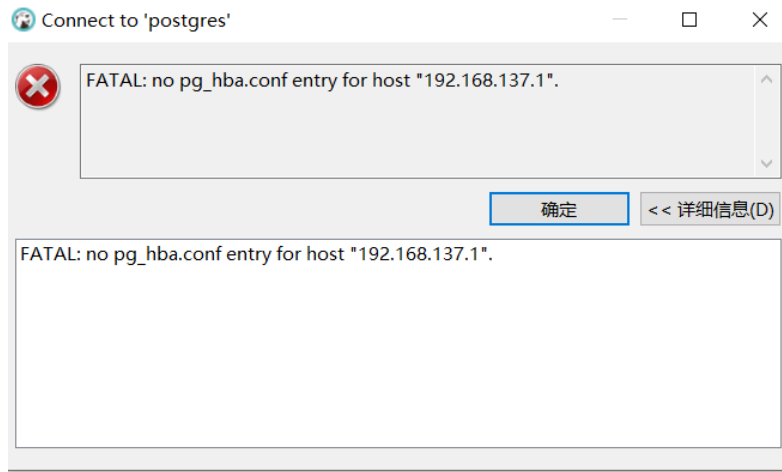
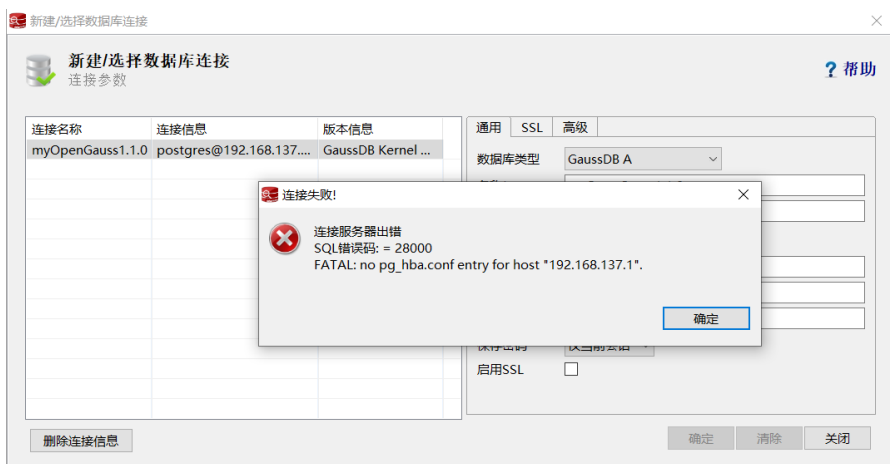
编译postgis

```
cd /opt/openGauss-server/third_party/dependency/postgis/postgis-xc/postgis-2.4.2

./configure --prefix=$toolhome/pggis2.4.2 \
--with-pgconfig=$GAUSSHOME/bin/pg_config \
--with-projdir=$toolhome/proj \
--with-geosconfig=$toolhome/geos/bin/geos-config \
--with-jsondir=$toolhome/json \
--with-xml2config=$toolhome/libxml2/bin/xml2-config \
--with-raster \
--with-gdalconfig=$toolhome/gdal/bin/gdal-config \
--with-topology \
--without-address-standardizer \
CFLAGS="-fPIC -O2 -fpermissive -DPGXC -pthread -D_THREAD_SAFE -D__STDC_FORMAT_MACROS -DMEMORY_CONTEXT_CHECKING -W -I$CODE_

make && make install
```

客户端IP地址未加入数据库白名单



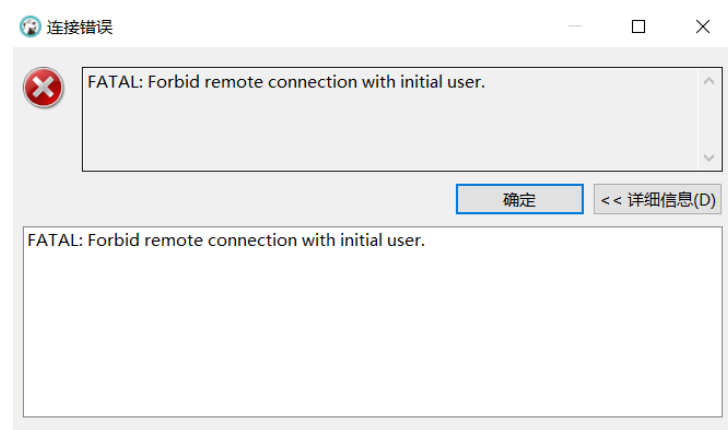
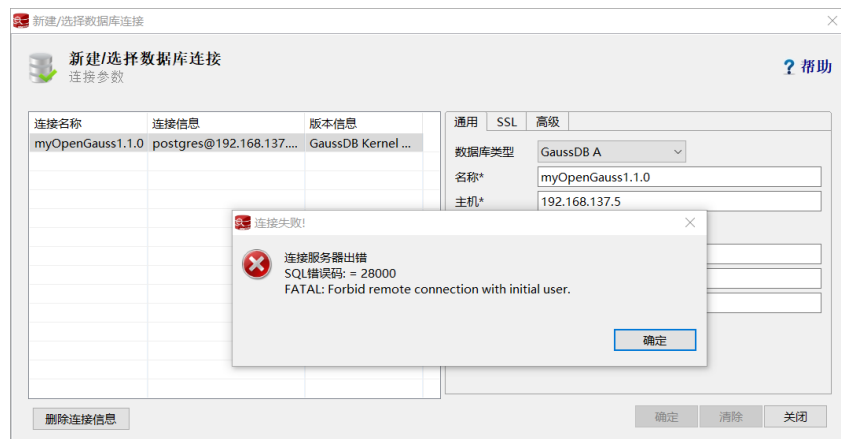
● pg_hba.conf文件记录格式

```
local.....database·user·auth-method·[auth-options]↵  
host.....database·user·address·auth-method·[auth-options]↵  
hostssl.....database·user·address·auth-method·[auth-options]↵  
hostnossl.....database·user·address·auth-method·[auth-options]↵  
host.....database·user·IP-address·IP-mask·auth-method·[auth-options]↵  
hostssl.....database·user·IP-address·IP-mask·auth-method·[auth-options]↵  
hostnossl.....database·user·IP-address·IP-mask·auth-method·[auth-options]↵
```

● pg_hba.conf文件参考配置

- 服务端本地数据库用户免密登录
local all all trust
- 拒绝管理用户从网络登录
host all postgres 0.0.0.0/0 reject
- 普通用户密码验证登陆
host all all 0.0.0.0/0 md5
- 流复制用户
host replication replica 172.18.0.10/32 trust
host replication replica 172.18.0.11/32 trust

初始化用户远程登录问题



类似操作系统设置禁止root远程登录，openGauss数据库限制初始化时设置的用户(默认为omm用户)，不允许远程访问数据库。数据库远程登录需要新建数据库用户，同时配置认证文件pg_hba.conf或者通过gs_guc工具进行设置。

数据库混用sha256与md5加密方式导致客户端连接认证失败

openGauss数据库默认安装后，创建的用户密码采用sha256的加密方式。修改加密方式为md5 (password_encryption_type=0)后需要重新修改已创建的数据库用户密码，然后再设置HBA文件认证方法为md5。

解决方法：重新修改用户密码。

JDBC驱动与java版本问题

JDBC字段返回大写jar包：

<https://www.modb.pro/download/117810>

← → ↺

docs.mogdb.io/zh/mogdb/v2.0.1/websphere-configures-mogdb(postgresql)-data-source-reference#websphere配置mogdbpostgresql数据源参

MogDB
Enterprise Ready Database

搜索文档

v2.0.1

关于MogDB

快速入门

容器化安装

单节点安装

访问数据库

使用命令行访问MogDB

使用图形工具访问MogDB

使用中间件访问MogDB

WebLogic配置
MogDB(PostgreSQL)数据源参考

WebSphere配置
MogDB(PostgreSQL)数据源参考

安装指南

下载对应JDBC驱动包

当前版本42.2.23。

这是驱动程序的当前版本。除非您有不寻常的需求（运行旧的应用程序或JVM），否则这就是您应该使用的驱动程序。

如果您使用的是Java8或更高版本，那么应该使用JDBC4.2版本。
如果您使用的是Java7，那么应该使用JDBC4.1版本。
如果您使用的是Java6，那么应该使用JDBC4.0版本。

各JDK对应下载JDBC版本链接如下：

JDK版本	JDBC驱动链接地址
1.6	https://jdbc.postgresql.org/download/postgresql-42.2.23.jre6.jar
1.7	https://jdbc.postgresql.org/download/postgresql-42.2.23.jre7.jar
1.8	https://jdbc.postgresql.org/download/postgresql-42.2.23.jar

Version	JDBC 4.0	JDBC 4.1	JDBC 4.2	Source
42.2.23	42.2.23 JDBC 4	42.2.23 JDBC 41	42.2.23 JDBC 42	42.2.23 JDBC Source
42.2.22	42.2.22 JDBC 4	42.2.22 JDBC 41	42.2.22 JDBC 42	42.2.22 JDBC Source
42.2.21	42.2.21 JDBC 4	42.2.21 JDBC 41	42.2.21 JDBC 42	42.2.21 JDBC Source

■ 列表分区

```
CREATE TABLE tab_list (  
    country varchar(20),  
    ...  
) PARTITION BY LIST (country)
```

■ 范围分区

```
CREATE TABLE pt1  
(  
    id    INTEGER    ,  
    name  varchar(20) ,  
    score DECIMAL(5,2)  
)  
PARTITION BY RANGE(score)  
(  
    PARTITION P1 VALUES LESS THAN(60) tablespace tbs1,  
    PARTITION P2 VALUES LESS THAN(85) tablespace tbs2,  
    PARTITION P3 VALUES LESS THAN(MAXVALUE) tablespace tbs2  
);
```

■ 哈希分区

```
CREATE TABLE tab_hash (  
    part_no varchar(20),  
    ...  
) PARTITION BY HASH (part_no)
```

■ 自动扩展间隔分区

```
CREATE TABLE tab_interval (  
    create_date date,  
    ...  
) PARTITION BY RANGE (create_date)  
INTERVAL ( '1 month' )
```

■ 分区表操作

- select * from ... partition (p1);
- alter table ... add partition p2
- alter table ... drop partition p3;
- alter table ... merge partitions p4,p5 into partition p4_5;
- alter table ... split partition p6 into (partition p61,partition p62);
- alter table ... exchange partition (p7) ...;
- alter table ... truncate partition p8;

用户自定义函数

兼容PostgreSQL风格

```
create or replace function months_between()  
returns number  
as $function$  
begin  
    return 0;  
end;  
$function$ language plpgsql;
```

编程语言

```
gsql ((GaussDB Kernel V500R001C20 build ) compiled  
Non-SSL connection (SSL connection is recommended v  
Type "help" for help.  
  
postgres=# select lanname from pg_language;  
lanname  
-----  
internal  
c  
sql  
java  
plpgsql  
(5 rows)
```

兼容Oracle风格

```
create or replace function months_between()  
return number  
as  
begin  
    return 1;  
end;  
/
```

调用方式

```
select fun();  
select (fun()).*  
select * from fun();
```

用户自定义函数

```
[omm@opengauss1 ~]$ gsql -Uomm postgres -r
gsql ((GaussDB Kernel V500R001C20 build ) compiled at 2021-09-08 0
7:40:49 commit 0 last mr )
Non-SSL connection (SSL connection is recommended when requiring h
igh-security)
Type "help" for help.

postgres=# drop schema public;
DROP SCHEMA
postgres=# create or replace function postgres.months_between()
postgres-# returns number
postgres-# as $function$
postgres$$ begin
postgres$$     return 1;
postgres$$ end;
postgres$$ $function$ language plpgsql;
CREATE FUNCTION
postgres=#
postgres=# show search_path;
search_path
-----
"$user",public
(1 row)

postgres=#
```

```
[omm@opengauss1 ~]$ gsql -Uomm postgres -r
gsql ((GaussDB Kernel V500R001C20 build ) compiled at 2021-09-08 0
7:40:49 commit 0 last mr )
Non-SSL connection (SSL connection is recommended when requiring h
igh-security)
Type "help" for help.

postgres=# create or replace function months_between()
postgres-# returns number
postgres-# as $function$
postgres$$ begin
postgres$$     return 1;
postgres$$ end;
postgres$$ $function$ language plpgsql;
ERROR:  schema "public" does not exist
postgres=#
postgres=# create or replace function pg_catalog.months_between()
postgres-# returns number
postgres-# as $function$
postgres$$ begin
postgres$$     return 1;
postgres$$ end;
postgres$$ $function$ language plpgsql;
ERROR:  the search_path is empty while the porc belongs to pg_cata
log
CONTEXT:  compilation of PL/pgSQL function "months_between" near l
ine 1
```

用户必须要有一个有效的search_path


```
create or replace function schema.months_between()  
returns number  
as $function$  
declare  
    v_param1 schemaname.tablename.column%type;  
begin  
    return 0;  
end;  
$function$ language plpgsql;
```

函数里引用其它schema下的对象需要明确指定schema

数据库兼容性



openGauss



Gauss松鼠会

● 初始化指定

```
gs_initdb --nodename=og_6432 \  
--pgdata=/home/omm/data6432 \  
--encoding=UTF-8 \  
-- locale=en_US.UTF-8 \  
--dbcompatibility='A' \  
--username=omm \  
--pwpasswd=Abcd@2021
```

● 创建DB时指定

```
CREATE DATABASE db_gbk  
WITH ENCODING 'GBK'  
LC_COLLATE='C'  
LC_CTYPE='C'  
DBCOMPATIBILITY='PG'  
TEMPLATE=template0;
```

```
postgres=# select a,encode(a::bytea,'hex') from t_char ;  
 a      | encode  
-----+-----  
 0      | 30202020202020202020  
 a      | 61202020202020202020  
 中     | e4b8ad20202020202020  
(3 rows)
```

```
db_gbk=# select a,encode(a::bytea,'hex') from t_char ;  
 a      | encode  
-----+-----  
 0      | 30202020202020202020  
 a      | 61202020202020202020  
 中     | d6d02020202020202020  
(3 rows)
```

```
create table t_char(a char(10));  
create table t_varchar(a varchar(10));  
insert into t_char values('0'),('a'),('中');  
insert into t_varchar values('0'),('a'),('中');
```

```
postgres=# select a,encode(a::bytea,'hex') from t_varchar;  
 a      | encode  
-----+-----  
 0      | 30  
 a      | 61  
 中     | e4b8ad  
(3 rows)
```

```
db_gbk=# select a,encode(a::bytea,'hex') from t_varchar;  
 a      | encode  
-----+-----  
 0      | 30  
 a      | 61  
 中     | d6d0  
(3 rows)
```

```
db_a=# create table t_char(a char(1));  
CREATE TABLE  
db_a=# insert into t_char values('中');  
ERROR:  invalid byte sequence for encoding "UTF8": 0xe5 0xe2 0x80
```

```
db_pg=# create table t_char(a char(1));  
CREATE TABLE  
db_pg=# insert into t_char values('中');  
INSERT 0 1
```

日期时间类型

```
db_pg=# create table tab(
db_pg(# a date,
db_pg(# b pg_catalog.date);
CREATE TABLE
db_pg=# \d tab
      Table "public.tab"
  Column | Type   | Modifiers
-----+-----+-----
 a       | date   |
 b       | date   |
```

```
db_oracle=# create table tab(
db_oracle(# a date,
db_oracle(# b pg_catalog.date);
CREATE TABLE
db_oracle=# \d tab
      Table "public.tab"
  Column | Type                                | Modifiers
-----+-----+-----
 a       | timestamp(0) without time zone |
 b       | date                             |
```

推荐使用pg原生的date类型

- Statement级别或者事务级别的影响
- 客户端时区设置的影响
- Daylight Saving Time (DST)

考虑到日期和时间函数不同方面的影响（例如客户端时区设置和DST更改的影响）之后，可以使用如下函数来模拟Oracle的SYSDATE 函数。它提供了一个语句级的时间戳，并不受客户端设置的影响：

```
CREATE OR REPLACE FUNCTION <<Your schema>>.sysdate()  
RETURNS TIMESTAMP WITHOUT TIME ZONE  
AS  
$BODY$  
    SELECT clock_timestamp() AT TIME ZONE '<<DB Timezone>>';  
$BODY$  
LANGUAGE sql;
```

□ 权限体系

- ◆ 实例权限pg_hba.conf
- ◆ 数据库级权限(CTc)
- ◆ 模式级权限(UC)
- ◆ 对象级权限
- ◆ 行级访问控制
- ◆ 成员角色及继承权限
- ◆ 默认权限

□ 系统权限和对象权限

- ◆ 系统权限在创建用户或角色时指定，包括login、inherit、sysadmin、createdb、createrole等
- ◆ 对象权限是指在表、表列、序列、函数等数据库对象上执行特殊动作的权限，其权限类型有select、insert、update、delete、references、create、connect、temporary、execute和usage等

□ 访问控制列表(ACL)

- ◆ pg_database.datacl
- ◆ pg_namespace.nspacl
- ◆ pg_class.relacl
- ◆ pg_attribute.attacl
- ◆ pg_proc.proacl
- ◆ pg_type.typacl
- ◆ pg_language.lanacl
- ◆ pg_tablespace.spcacl

□ 三权分立

- ◆ 实系统管理员
- ◆ 安全管理员
- ◆ 审计管理员

□ 私有用户

- ◆ INDEPENDENT

初始用户的权限不受三权分立设置影响

- ❑ `gs_dump` 可以选择一个数据库或部分表进行备份
- ❑ `gs_dumpall` 备份整个cluster
- ❑ `copy \copy` 导入导出表数据

dump的四种格式

- `-Ft` tar格式的转储输出格式，不支持压缩。
- `-Fd` 以目录的格式创建备份
- `-Fc` 备份为二进制格式, 压缩存储, 并且可被 `gs_restore` 用于精细还原
- `-Fp` 备份为文本, 大库不推荐
- `copy` 命令用于表与文件之间的相互拷贝;
- `copy` 命令在数据库服务端操作文件，需要在数据库中配置读写权限;
- `\copy` 命令在客户端执行导入客户端的数据文件，无需额外申请权限;
- 支持三种格式：文本格式(默认)、CSV格式(可跨平台)和二进制格式;

□ 客户端验证服务器证书的两种模式

- 客户端连接参数SSLMODE设置为verify-ca 仅校验数据库证书真伪。
- 客户端连接参数SSLMODE设置为verify-full校验数据库证书真伪及通用名CN匹配数据库连接的hostname。

```
gsql "sslmode=verify-ca" -p6432 -h 192.168.137.5 -Upostgres
gsql: root certificate file "/home/omm/.postgresql/root.crt" does not exist
Either provide the file or change sslmode to disable server certificate verification.
```

```
gsql "sslmode=verify-full" -p6432 -h opengauss1 -Upostgres
gsql: server common name "192.168.137.5" does not match host name "opengauss1"
```

□ 服务端验证客户端证书的两种模式

- 数据库认证文件pg_hba.conf配置认证选项clientcert=verify-ca仅验证客户端证书真伪，认证方法可选。
- 数据库认证文件pg_hba.conf配置认证方法cert，免密验证客户端证书真伪及CN匹配数据库连接用户名或映射匹配。

```
hostssl all all 0.0.0.0/0 md5
```

```
hostssl all all 0.0.0.0/0 md5 clientcert=verify-ca
```

```
hostssl all all 0.0.0.0/0 cert
```

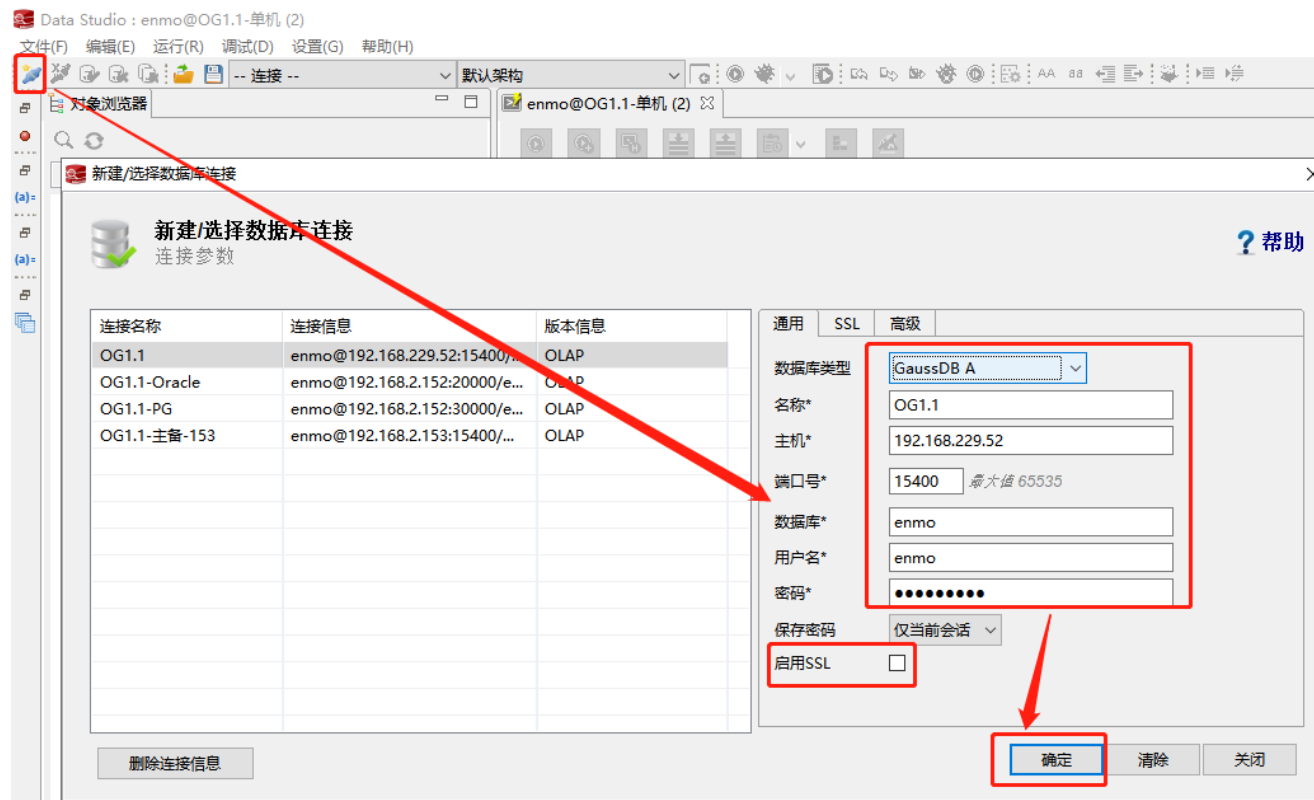
Data Studio连接OpenGuass数据库

- 数据库端设置访问权限 (pg_hba.conf)

\$ gs_guc reload -D /opt/OpenGuass/data/dn -h "host all all 192.168.229.1/32 sha256"

- 客户端连接设置

以非SSL方式连接数据库：



以SSL方式连接数据库：要选中启用SSL复选框，并且选择合适的SSL认证模式以及提供的相应的证书文件。

The image displays two side-by-side screenshots of the DataStudio connection configuration window. The left screenshot shows the '通用' (General) tab with various database connection parameters. The '启用SSL' (Enable SSL) checkbox is checked and highlighted with a red box. A red arrow points from this checkbox to the right screenshot. The right screenshot shows the 'SSL' tab, which is also highlighted with a red box. This tab contains fields for '客户端SSL证书' (Client SSL Certificate), '客户端SSL密钥' (Client SSL Key), '根证书' (Root Certificate), and 'SSL 密码' (SSL Password), all of which have file selection buttons. Below these fields, the 'SSL模式' (SSL Mode) dropdown menu is open, showing options: 'require', 'require' (highlighted), 'verify-ca', and 'verify-full'. The 'require' option is selected. At the bottom of both windows are buttons for '确定' (OK), '清除' (Clear), and '关闭' (Close).

通用 SSL 高级

数据库类型 GaussDB A

名称* OG1.1

主机* 192.168.229.52

端口号* 15400 最大值 65535

数据库* enmo

用户名* enmo

密码*

保存密码 仅当前会话

启用SSL ☒

确定 清除 关闭

通用 SSL 高级

客户端SSL证书 C:\Knowledge_20210108\PG数...

客户端SSL密钥 C:\Knowledge_20210108\PG数...

根证书 C:\Knowledge_20210108\PG数...

SSL 密码

SSL模式 require

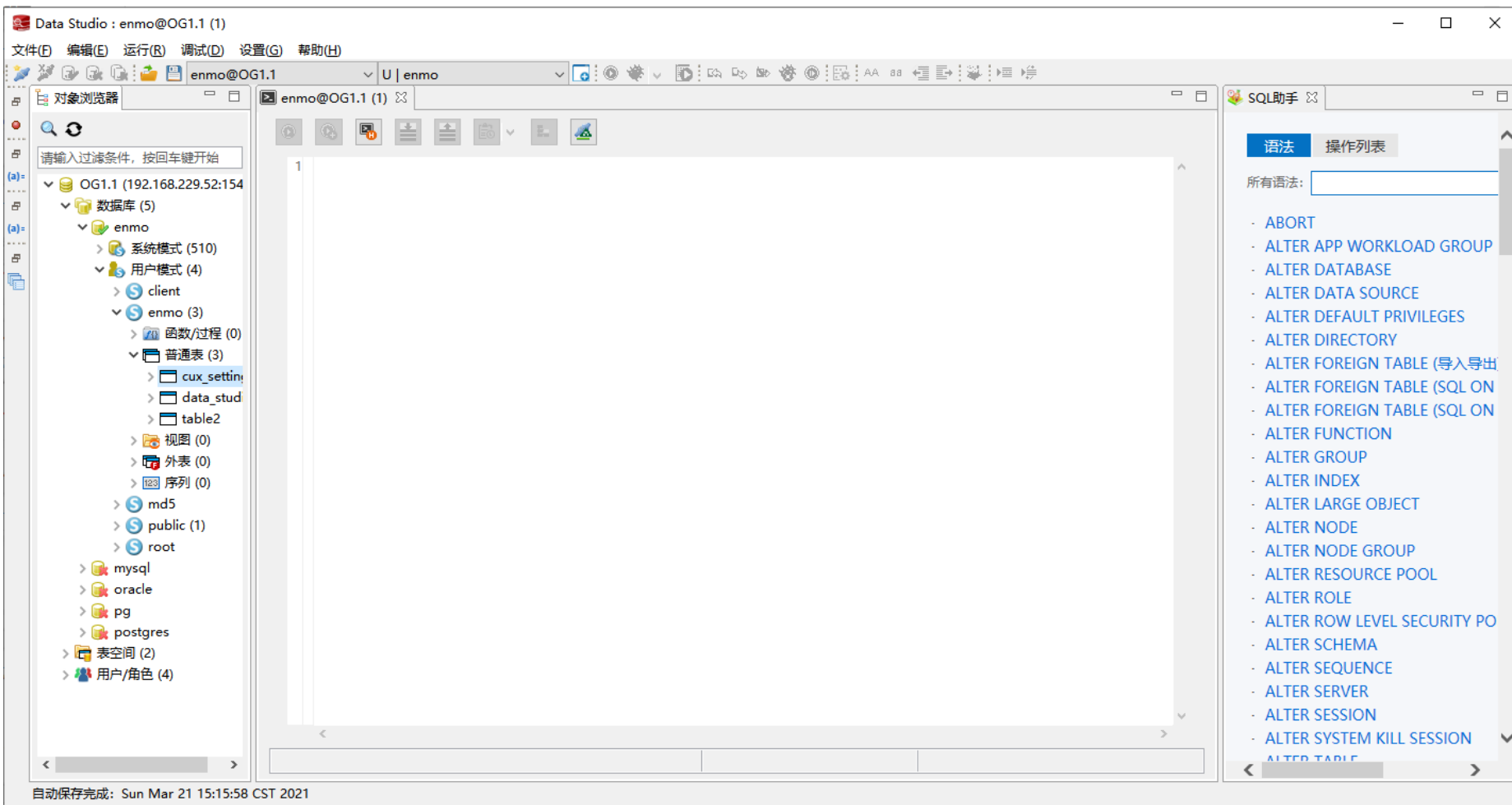
require

verify-ca

verify-full

确定 清除 关闭

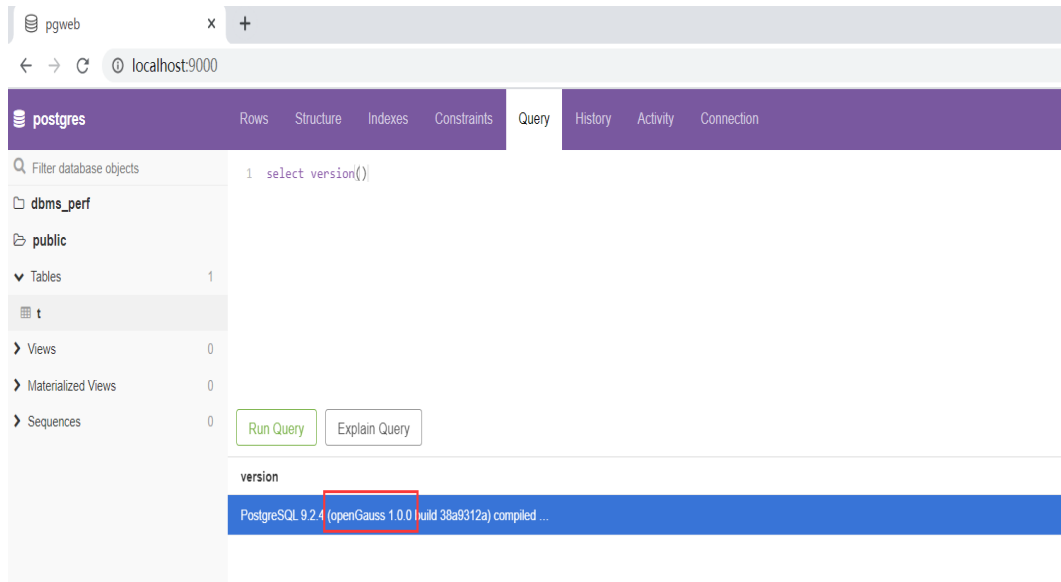
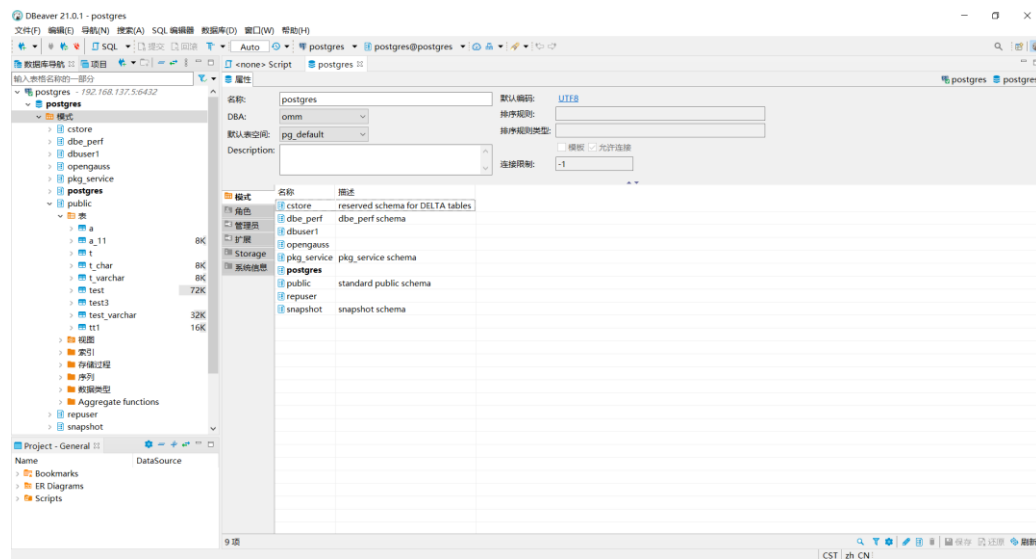
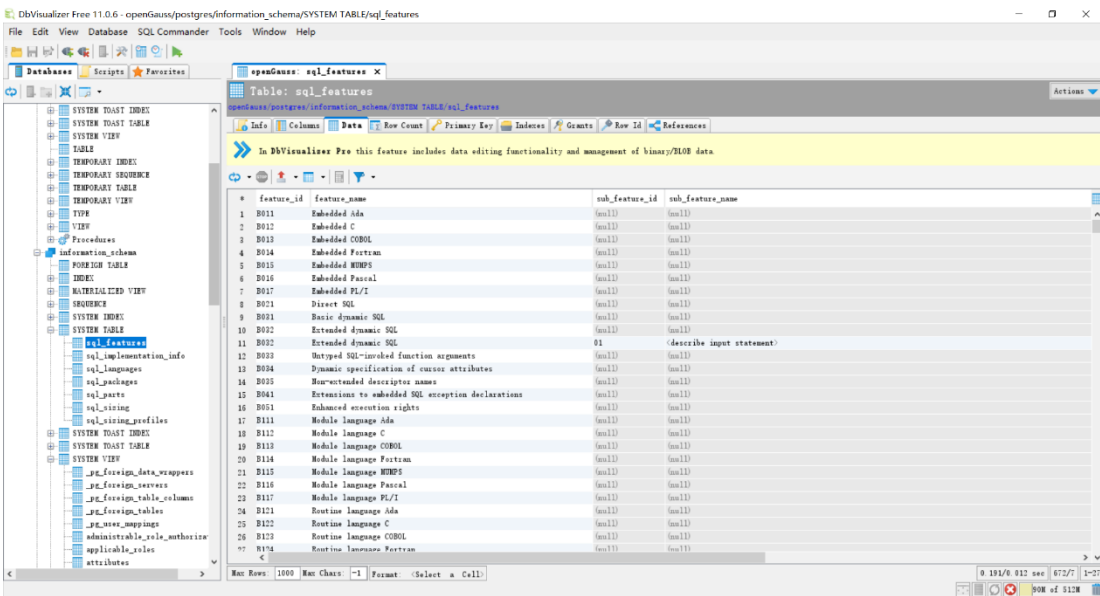
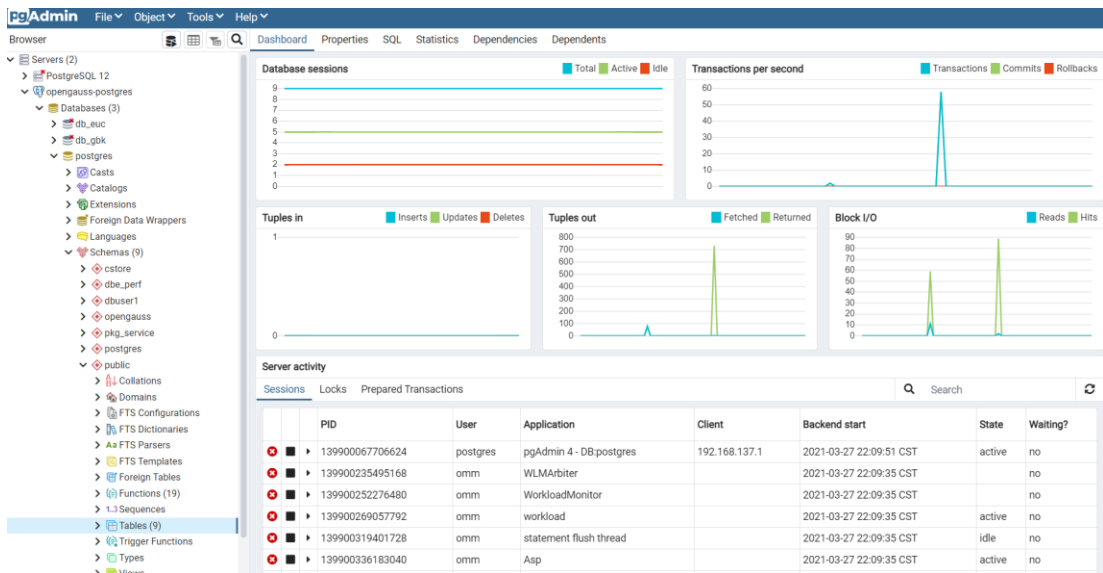
Data Studio工具操作界面



其它客户端工具



Gauss松鼠会



- A. [PostgreSQL/openGauss数据库易犯的十大错误](#)
- B. [Oracle数据表搬迁到openGauss](#)
- C. [openGauss与PostgreSQL分区策略语法测试](#)
- D. [openGauss与PostgreSQL对比测试SSL之自签名CA证书双向认证测试](#)
- E. [openGauss与PostgreSQL对比测试SSL之自签名CA证书单向认证测试](#)
- F. [openGauss与PostgreSQL对比测试SSL之自签名私有证书测试](#)
- G. [openGauss备份工具gs_probackup细项使用](#)
- H. [opengauss 1.1.0支持从库基准备份](#)
- I. [openGauss编译支持postgis](#)
- J. [opengauss1.0.1支持oracle_fdw和mysql_fdw](#)
- K. [opengauss1.0.1部分新特性测试](#)
- L. [opengauss1.0.1支持docker主备搭建](#)
- M. [PG客户端工具测试连接openGauss](#)
- N. [openGauss常用的客户端连接工具](#)
- O. [抢鲜体验2：openGauss从源码到主备](#)
- P. [抢鲜体验：openGauss安装编译过程要点及问题解决](#)

感谢聆听

彭冲 (微信号: skypkmoon)
手机: 13642168529
邮箱: chong.peng@enmotech.com



Gauss松鼠会