



数据库启动

- 1.pg_ctl start -D \$PGDATA/数据库名称
- 2.\$PGHOME/network/startpg.sh

数据库路径

数据库的文件路径存放于PG数据服务器:
\$HOME/pgdata/数据库名

数据库关闭

- 1.pg_ctl -D \$PGDATA/数据库名称 stop -m smart
- 2.pg_ctl -D \$PGDATA/数据库名称 stop -m fast
- 3.pg_ctl -D \$PGDATA/数据库名称 stop -m immediate



psql使用

- 1.psql -p 端口号
- 2.psql -p 端口号 -d 数据库名
- 3.psql -p 端口号 -d 数据库名 -U 用户名
- 4.在psql中切换用户 \c - 用户名

命令3.1

以postgres用户进入缺省数据库 postgres。

命令3.2

以postgres用户进入指定数据库。

命令3.3

以指定用户进入指定数据库。

端口号

查看数据库文件: postgresql.conf 参数port。

文件目录

Pgdata目录下不建议存放任何非pg数据库的其他目录。

3.immediate

立即退出, 相当于 abort。

关闭方案

由于GeoEast采用连接池技术, 因此推荐使用1。

1.smart

不允许新的连接, 等待会话结束。

2.fast

不允许新的连接, 终止正在进行的回话。

状态查询

- 1.数据查询数据库名称
Select datname from pg_database;
- 2.查询数据库连接状态
Select datname,username,client_addr,wait_event,state,query from pg_stat_activity
- 3.查询数据库参数
Select name ,setting from pg_settings;
- 4.查询锁等待
SELECT pid, wait_event_type, wait_event FROM pg_stat_activity WHERE pid <> pg_backend_pid() and wait_event is NOT NULL;

数据库活动状态

查询数据库的名称、连接状态、实时参数、锁等待、监听端口等

查看监听端口

netstat -lntp |grep 端口号

运维关注

PostgreSQL对磁盘空间比较敏感, 运维关注重点。

空间查询

- 1.查询数据库大小
select datname,pg_size_pretty(pg_database_size(datname)) as size from pg_database;
- 2.查看表空间大小
Select spcname,pg_size_pretty(pg_tablespace_size(spcname))as size from pg_tablespace where spcname not like('virtual%')order by 2;
- 3.查看项目中表的空间占用
Select relname,pg_size_pretty(pg_relation_size(relid)) as size from pg_stat_user_tables where schemaname=lower('项目名')order by 2;

数据库空间查询

查询数据库、表空间、项目表空间、磁盘空间大小。

查看数据库磁盘

df -h

状态查询

1. cpu与iowait
sar -u 1 3
2. 查看平均负载
sar -q 3 3
3. 查看内存使用
sar -r 3 3
4. 查看交换区
sar -W 3 3
5. 查看IO
sar -b 3 3
6. 查看网络接口
sar -n DEV 1 1
7. 查看tcp连接统计
sar -n TCP 1 1

服务器状态检查

关注服务器CPU、iowait、平均负载、内存、网络等。

参数调优

关注服务器CPU、iowait、平均负载、内存、网络等。

系统参数

- =关闭numa
- =关闭透明大页
- =设置noatime
- =vm.swappiness = 0
- =vm.owvercommit_memory = 2
- =vm.dirty_background_ratio = 5
- =vm.dirty_ratio = 10

参数设置

Shared_buffers: 内存的max (内存的10%, 8G)
Work_mem:4M
Fsync=off
effective_cache_size= Shared_buffers
Commit_siblings &commit_delay (20, 100)
archive_mode =off
Autovacuum = on;
Autovacuum_naptime
Full_page_wirtes=off

PG参数优化

大型处理中心根据作业数建议Shared_buffers设置为10G或20G。
vacuum如果有能力可以设置为off, 手动的进行回收, 时间间隔缺省是1分钟, 太频繁了可以调节成三十分钟。

系统参数优化

高并发下, 务必关闭numa, 负责会引起 Remote Acces瓶颈。解释系统不用考虑。查看服务器是否启动numa可以使用lscpu或numctl - hardware。
高并发环境vm.owvercommit_memory建议设置为2, 其他环境设置为0。如果系统内存很大, vm.*_radio可适当调小。

状态关注

负载关注iowait和idle
内存关注kbcommit

数据库备份

数据库的冷热备份和参数设置以及热备操作实例。

冷备份

数据库停库以后, tar -cvf 备份名.tar 数据库路径。

热备实例

```
pg_basebackup -Ft -Pv -Xf -z -Z5 -p 5533 -D ndp04 -U admin1 -h 130.1.10.99
pg_basebackup: initiating base backup, waiting for checkpoint to complete
pg_basebackup: checkpoint completed
transaction log start point: 0/2E000028 on timeline 1
817760/817760 kB (100%), 67/67 tablespaces
transaction log end point: 0/2E0141C0
pg_basebackup: base backup completed
```

热备份

pg_basebackup -Ft -Pv -Xf -z -Z5 -p 端口号 -D 数据库名称 -U admin1 -h IP地址

热备结果

```
[postgres@ndn10-svr ndp04]$ ls
102045.tar.gz 126237.tar.gz 36381.tar.gz
102046.tar.gz 126238.tar.gz 36382.tar.gz
105501.tar.gz 24578.tar.gz 39837.tar.gz
```

数据库恢复

通过备份库, 将数据恢复到最近备份包。

停库

- 1.将原数据库移走
cd \$PGDATA
- 2.新建同名库目录
mkdir ndp
- 3.修改目录权限
chmod 0700 ndp

配置与重启

- 1.配置recovery.conf
cp /recovery.conf.sample recovery.conf
vi recovery.conf
restore_command = 'cp /data/postgre/pgdata/backup/%f %p'
- 2.启动数据库
pg_ctl start -D ndp
- 3.恢复完成
恢复文件后缀名变为.done。

解压

将备份文件base.tar.gz解压至新目录
cd ndp
tar -xvf /data/postgre/pgdata/backup/base backup/base.tar.gz -C

多表空间

多个表空间, 参照tablespace_map文件, 解压其他表空间到指定目录
tar -xvf /data/postgre/pgdata/backup/base backup/232379.tar.gz -C /data/postgre/pgdata/ndp02/ndp02/virtualproject_1_data

