

PostgreSQL - Dumping Data, Stopping Queries and Maintenance

Windows

```
cd "\\Program Files\\PostgreSQL\\9.3\\bin"
pg_dumpall -Upostgres > data.bu
```

Linux

```
su - postgres
pg_dumpall > data.bu
zip data.zip data.bu
exit
```

The above will save the data in the folder /var/lib/pgsql

PostgreSQL - Restoring Data

Since you are restoring the whole database you will need to delete the database completely. So please remember to do a back up before you do so. Also, **you must stop tomcat**.

Windows

```
cd "\\Program Files\\PostgreSQL\\9.3\\bin"
psql -Upostgres
drop database da;
\q
unzip data.zip
psql -Upostgres < data.bu
```

Linux

```
su - postgres
psql
drop database da;
\q
psql < data.bu
exit
```

Important Maintenance Statements

```
-- Show Running Queries -- Look for active ones

SELECT pid, age(query_start, clock_timestamp()), username, state ,query
FROM pg_stat_activity

ORDER BY query_start desc;

-- kill running query
SELECT pg_cancel_backend(procpid);

-- kill idle query
SELECT pg_terminate_backend(procpid);

-- vacuum command
VACUUM (VERBOSE, ANALYZE);

-- all database users
select * from pg_stat_activity where current_query not like '<%';

-- all databases and their sizes
select * from pg_user;

-- all tables and their size, with/without indexes
select datname, pg_size_pretty(pg_database_size(datname))
from pg_database
order by pg_database_size(datname) desc;

-- cache hit rates (should not be less than 0.99)
SELECT sum(heap_blks_read) as heap_read, sum(heap_blks_hit) as heap_hit,
(sum(heap_blks_hit) - sum(heap_blks_read)) / sum(heap_blks_hit) as ratio
FROM pg_statio_user_tables;

-- table index usage rates (should not be less than 0.99)
SELECT relname, 100 * idx_scan / (seq_scan + idx_scan)
percent_of_times_index_used, n_live_tup rows_in_table
FROM pg_stat_user_tables
ORDER BY n_live_tup DESC;

-- how many indexes are in cache
SELECT sum(idx_blks_read) as idx_read, sum(idx_blks_hit) as idx_hit,
(sum(idx_blks_hit) - sum(idx_blks_read)) / sum(idx_blks_hit) as ratio
FROM pg_statio_user_indexes;

-- Dump database on remote host to file
$ pg_dump -U username -h hostname databasename > dump.sql

-- Import dump into existing database
$ psql -d newdb -f dump.sql
```

From:

<http://wiki.dreamapps.com/wiki/> - **DreamApps Wiki**

Permanent link:

<http://wiki.dreamapps.com/wiki/doku.php?id=deployment:postgresql&rev=1514500248>

Last update: **2017/12/28 22:30**

