

Managing Resources with PostgreSQL

FOSDEM PGDay 2015

Cédric Villemain `cedric@2ndQuadrant.fr`

30th January 2015



PostgreSQL Expertise
Development, Support, Training

What resources ?

- CPU
- RAM
- Disk

How ?

- with Concurrency
- with Quota or Limit

Where ?

Operating System

- user process (limits.conf, ...)
- container (cgroup, jail, ...)
- virtualization (VMWare, Xen, ...)

Where ?

PostgreSQL

- Main configuration (postgresql.conf)
- per object
- per role
- per database
- per role in database
- pg_service

GUCs with restart

- `max_connections`
- `shared_buffers`

GUCs without restart

- temp_buffers
- work_mem
- synchronous_commit
- temp_tablespace

More GUCs without restart

- `temp_file_limit`
- `statement_timeout`
- `lock_timeout`

Resources per object

- FUNCTION
- ROLE
- DATABASE
- SYSTEM

Function example

```
ALTER FUNCTION update_datamart()  
    SET temp_file_limit = -1;
```

Database example

```
ALTER DATABASE datawarehouse  
  WITH CONNECTION LIMIT 10; --hard limit
```

```
ALTER DATABASE datawarehouse  
  SET temp_tablespace TO dwh_tblspc;
```

Role example

```
ALTER ROLE web_user  
  WITH CONNECTION LIMIT 10; -- hard limit but ...
```

```
ALTER ROLE dba  
  SET work_mem = '128MB';
```

Role in Database example

```
ALTER ROLE ALL IN DATABASE devel  
    SET synchronous_commit TO off;
```

System example

```
ALTER SYSTEM
    SET temp_buffers = '12MB';

SELECT pg_reload_conf();
```

Connection Limit

- PostgreSQL flood & DoS
- pooling & bouncing

PgBouncer

- `pool_size = 20`
- `max_client_conn = 2000`
`my_db = user=web_user host=remote_host pool_size=4`
- bonus: PAUSE / RESUME

top-like

- pg_activity (python, system only)
- pg_top (C, PostgreSQL extension)

pg_proctab

- pg_cputime()
- pg_loadavg()
- pg_memusage()
- pg_proctab()

Execution depends on loadavg

```
if (select load1<1 from pg_loadavg())  
then update_datamart();  
end if;
```



What about effective_cache_size ?

```
select pg_size_pretty(memcached * 4096)
from pg_memusage;
-- don't forget to count shared_buffers
```



Are we writing a lot on disk?

```
select pg_size_pretty(wchar) as requested_write,  
       pg_size_pretty(writes) as really_written,  
       (writes * 100 / wchar) as percent_really_written  
from pg_proctab()  
where pid = pg_backend_pid()
```



And next ?

Questions ?

Now is the time to ask!

