# Администрирование PostgreSQL в Avito

Сергей Бурладян
sburladyan@avito.ru

2016, Москва

# Debian GNU/Linux

- Утилиты управления postgres
  - pg_createcluster
  - pg_dropcluster
  - pg_ctlcluster
  - pg_lscluster

- /etc/postgresql/9.2/main/
- /var/lib/postgresql/9.2/main/

# Debian GNU/Linux

- /etc/init.d/postgresql
- /usr/share/postgresql-common/init.d-functions

```
do_ctl_all() {
...
        if [ "$1" = "stop" ] || [ "$1" = "restart" ]; then
            ERRMSG=$(pg_ctlcluster --force "$2" "$name" $1 2>&1)
        else
            ERRMSG=$(pg_ctlcluster "$2" "$name" $1 2>&1)
        fi
```

- /usr/bin/pg_ctlcluster
  - pg_ctl -m fast
  - pg_ctl -m immediate
  - kill -9 $PID

# Архив

- local.sh
- remote.sh

- archive_command = local.sh
- local.sh: cat WAL | ssh arch_srv remote.sh

- local.sh
  - ssh -o ControlMaster=auto -o ControlPersist=yes
  - /usr/local/bin/vmtouch "$wal"

# Архив

- remote.sh

```
if [[ $wal == ???????????????????????? ]]; then
    cat > /tmp/new_wal
    pbzip2 -p"$pmax" < /tmp/new_wal > "$arh_dir"/"$wal".tmp
else
    cat > "$arh_dir"/"$wal".tmp
fi
mv "$arh_dir"/"$wal".tmp "$arh_dir"/"$wal"
```

  - получение WAL со stdin
  - сжатие в несколько потоков
  - пропуск служебных файлов: 00000013.history, 000000010000000000000002.00000020.backup
  - атомарное появление файла в архиве

# Архив

- Очистка старых WAL
  - pg_archivecleanup
  - backup_label

```
LAST_NEEDED_WAL=$(grep 'START WAL LOCATION.\+(file .\+)'
"$SRC_DIR"/backup_label 2> /dev/null | sed -e 's/.\+(file \(.\+\))/\1/')
```

# bash

- trap exit_clean EXIT
- массивы
  - hosts=( host1 host2 host3 host4 )
  - declare -A db_ver=( [host3]=9.4 )
- параллельная обработка:
  - parallel --gnu --halt 2 -j "$pmax" "$bindir"/parse_rdb.pl ::: "$fdir"/$fmask ::: "$key1" "$key2"
  - xargs
    - run () { }
    - export -f run
    - seq 0 3 | xargs -n1 -P4 $BASH -e -o pipefail -c 'run "$@"' --

# bash lock

```
LOCK_FILE="/var/tmp/cron_script1.sh.lock"
SUCCESS_FILE="/var/tmp/cron_script1.success"

{
    check_and_lock "$LOCK_FD" "$LOCK_FILE"

    set -e
    set -o pipefail

    # working

    touch "$SUCCESS_FILE"

} {LOCK_FD}>> "$LOCK_FILE"

check_and_lock ()
{
    local fd=$1
    local fname=$2

    flock -n "$fd" || { echo "LOCK_FILE '$fname' locked, abort" >&2; exit 1; }
    { echo -n "$$ "; date; } > "$fname"
}
```

# bash lock

```
{
    check_and_lock "$LOCK_FD" "$LOCK_FILE"

    {
        set -e
        set -o pipefail

        # working

        touch "$SUCCESS_FILE"

    } {LOCK_FD}>&-

} {LOCK_FD}>> "$LOCK_FILE"
```

# bash fsync

- dd conv=fsync

- coreutils sync

  Usage: sync [OPTION] [FILE]...

# psql экранирование

```
tbl_name='my table'
psql -c "select * from $tbl_name"
ERROR:  syntax error
```

- ```
  psql -vTBL_NAME="$tbl_name" -f- <<'EOF'
      select * from :"TBL_NAME"
  EOF
  ```

- :VAR – raw
- :"VAR" – quote_ident
- :'VAR' – quote_literal
  ```

# psql copy

- psql | psql

- pg_dump --table=TABLE | pg_restore

- set -o pipefail
- if [[ $? -ne 0 ]]; then cleanup; fi

# psql copy

```
psql \
    -vSRC_SCHEMA="$src_schema" -vSRC_TBL="$src_tbl" \
    -vDST_SCHEMA="$dst_schema" -vDST_TBL="$dst_tbl" \
    -vEOC='\.' \
    -f- <<'EOF' |
\echo begin;
\echo copy :"DST_SCHEMA".:"DST_TBL" from stdin;
copy :"SRC_SCHEMA".:"SRC_TBL" to stdout;
\echo :EOC
\echo commit;
EOF
    psql -f-
```

# psql copy

- psql | pv | psql

- ограничение скорости

  pv -L "$BPS"

- отображение прогресса

  2,73GiB 0:00:05 [ 558MiB/s]

# Файл конфигурации

- git → puppet → server
- include_if_exists 'localvars.conf'

```
file { '/etc/postgresql/9.2/main/localvars.conf':
  mode    => '0644',
  owner    => postgres,
  group    => postgres,
  content    => "# hostname
avito.hostname = '${hostname}'
",
}
```

# HA для read

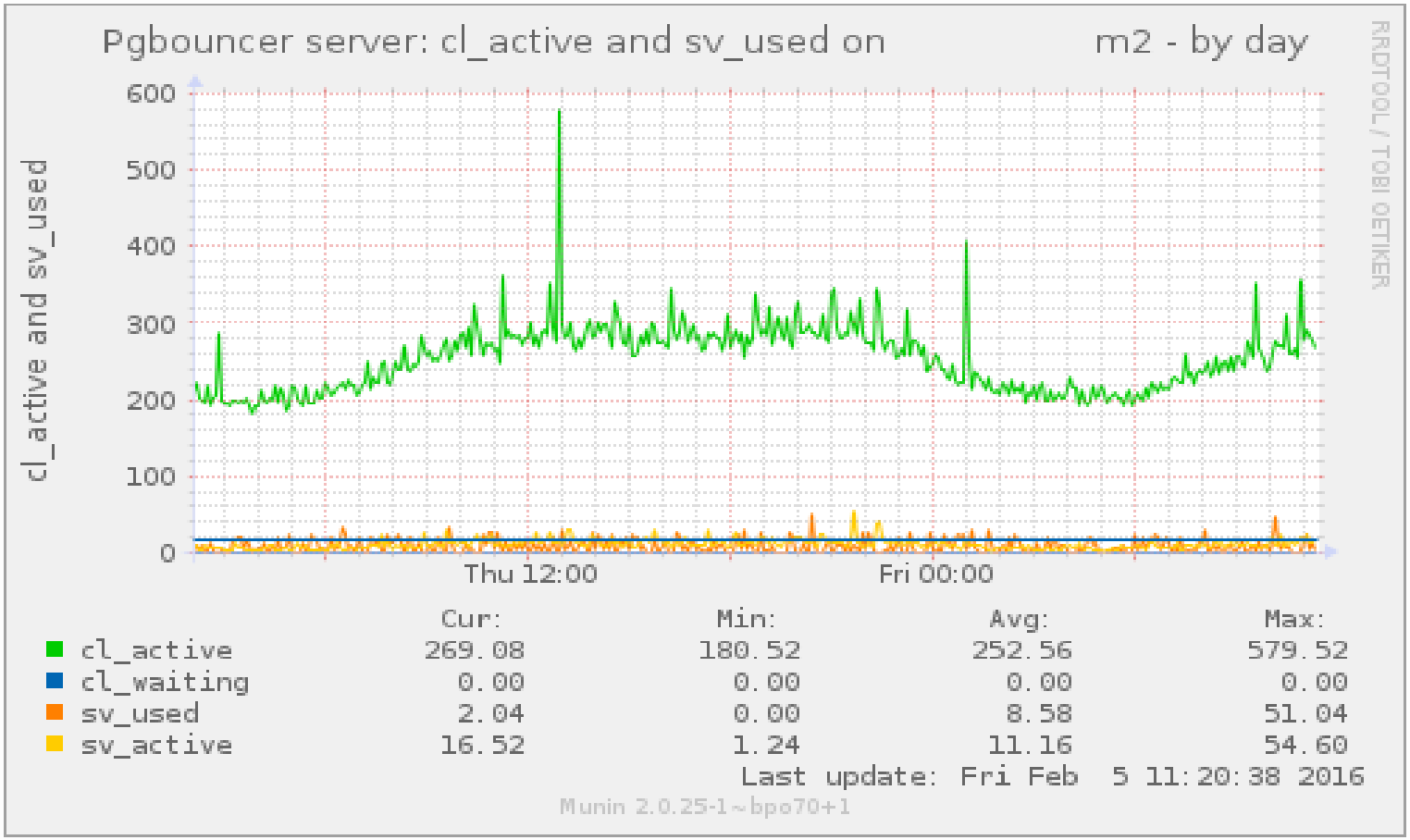
- haproxy для читающей нагрузки
- check.pl доступен через xinetd
- хранимка в базе — плавный вывод

# pgbouncer

- idle in transaction
  - nice
  - 2 баунсера

- avito_db1 = host=localhost user=webuser pool_size=50 datestyle='ISO,DMY' connect_query='select pool_init();'

# Мониторинг cl_waiting



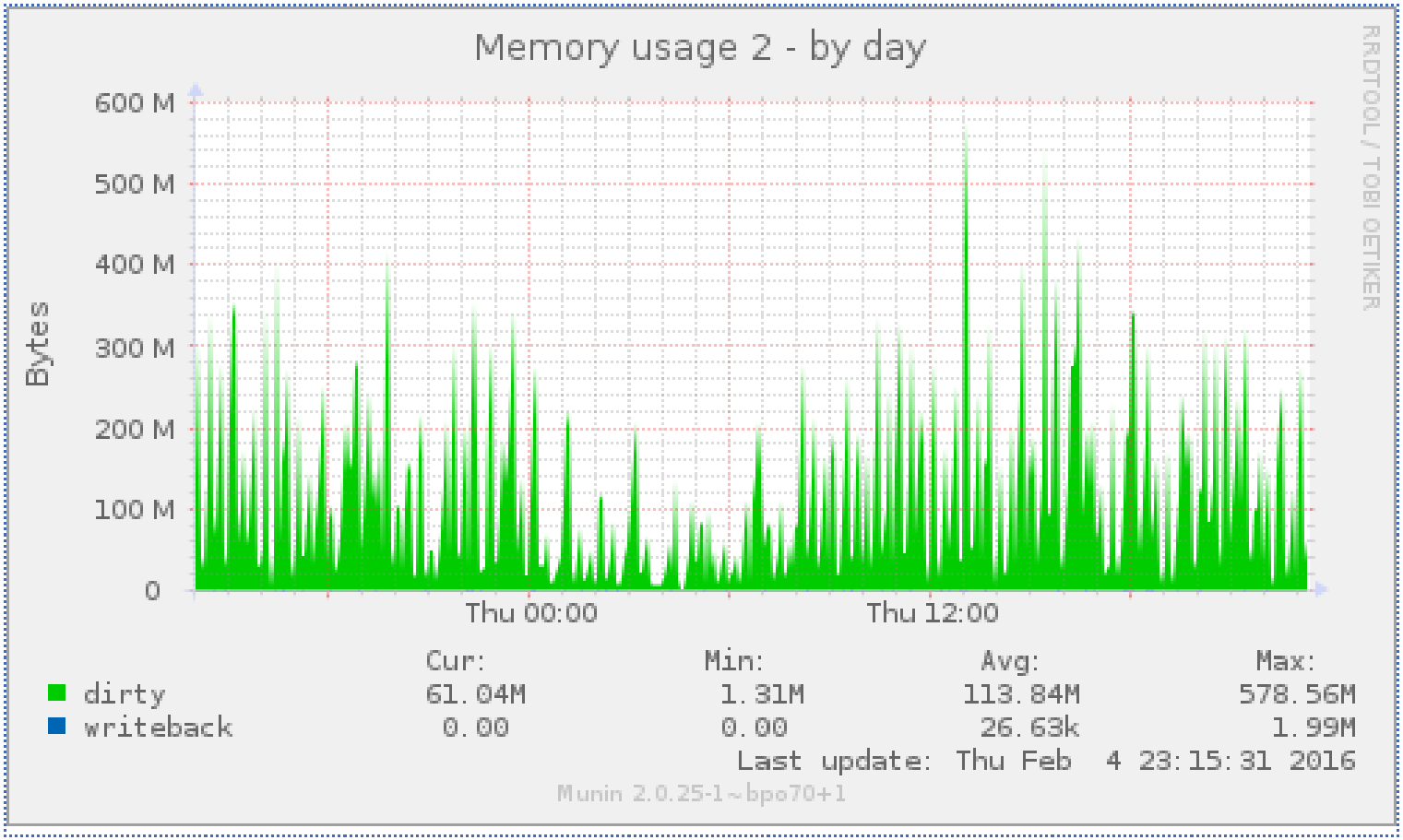
# Мониторинг cron

- munin plugins/mfile

- success файл
- mtime=$(($(date +%s) - $(date +%s -r "$filename")))

  echo "mfile.value $mtime"

# Мониторинг грязная память



# plproxy

- CREATE SERVER db_cluster FOREIGN DATA WRAPPER plproxy OPTIONS (

  p0 'host=localhost port=6433 dbname=db01',

  p1 'host=localhost port=6433 dbname=db02',

  p10 'host=localhost port=6433 dbname=db11',

  p11 'host=localhost port=6433 dbname=db12',

- p000 'host=localhost port=6433 dbname=db01',

  p001 'host=localhost port=6433 dbname=db02',

# plproxy

- pgbouncer

  db01 = host=node01 port=6432 dbname=db01

  db02 = host=node01 port=6432 dbname=db02

  db03 = host=node02 port=6432 dbname=db01

  db04 = host=node02 port=6432 dbname=db02

# Отладка

- ps U postgres

  16183 ?  Ss  0:00 postgres: postgres db_test [local] idle

- perf top -p 16183

  62.28%  postgres  postgres  [.] FunctionCall2Coll

  23.90%  postgres  postgres  [.] eqjoinsel

  12.37%  postgres  postgres  [.] int4eq

  0.43%  postgres  [kernel.kallsyms]  [k] memcpy

- gdb -p 16183

  (gdb) c

  (gdb) bt

# Отладка

- (gdb) break int4eq

  commands

  silent

  printf "%d = %d\n", fcinfo->arg[0], fcinfo->arg[1]

  cont

  end

  244 = 739

  244 = 46

  244 = 153

  244 = 1161

  …

- default_statistics_target!

# Прогресс выполнения запроса

- /proc/$PID/fd/

  /proc/$PID/fdinfo/
- cat /proc/16183/fdinfo/788

  pos: 15106048

  flags: 0100002
- ls -l /proc/16183/fd/788

  /proc/16183/fd/788 -> /var/lib/postgresql/9.2/main/base/16777/1620003615.7
- select relname from pg_class where relfilenode = 1620003615;

  relname

  ---

  user_id_phones_ix

# DDL

- alter table, autovacuum

- deadlock_timeout
- statement_timeout

# Seq scan по модулю

- select heavy_cpu_func(...) from ... where id % 4 = 0;

  select heavy_cpu_func(...) from ... where id % 4 = 1;

  select heavy_cpu_func(...) from ... where id % 4 = 2;

  select heavy_cpu_func(...) from ... where id % 4 = 3;
- synchronize_seqscans on

# Очередь на advisory lock

- select ... from test_q_events where pg_try_advisory_xact_lock(q.id)

  select ... from test_q_events where pg_try_advisory_xact_lock(q.id)

  -- process

  delete from test_q_events where id in (…)

  -- process

- recheck!

- 9.5

# WAL архивирование

- перегрузка HDD
- WAL ушли из кеша ФС
- archive_command долгая

- отправлять свежие горячие WAL в архив отдельно

# Debian 8 systemd

- $ sudo pg_ctlcluster 9.2 dev stop

  Redirecting stop request to systemctl

  - Jan 20 14:49:02 sql-host16 postgresql@9.2-main[38035]: pg_ctl: server does not shut down
  - Jan 20 14:49:02 sql-host16 systemd[1]: postgresql@9.2-main.service: control process exited, code=exited status=1
  - Jan 20 14:50:32 sql-host16 systemd[1]: postgresql@9.2-main.service stop-sigterm timed out. Killing.
  - Jan 20 14:50:32 sql-host16 systemd[1]: postgresql@9.2-main.service: main process exited, code=killed, status=9/KILL
  - Jan 20 14:50:33 sql-host16 systemd[1]: Unit postgresql@9.2-main.service entered failed state.
- TimeoutStopSec!

# Debian 8 systemd

- /etc/systemd/system.conf

  #DefaultTimeoutStartSec=90s

  #DefaultTimeoutStopSec=90s
- man systemd.kill

  SendSIGKILL= Defaults to "yes"

- не через systemd

  sudo pg_ctlcluster 9.2 main stop -- -m f

  sudo -u postgres pg_ctlcluster 9.2 main stop -m f

Спасибо за внимание!