# PostgreSQL
# インデックス検査ツール

## amcheck

2016-10-02 JPUG 合宿

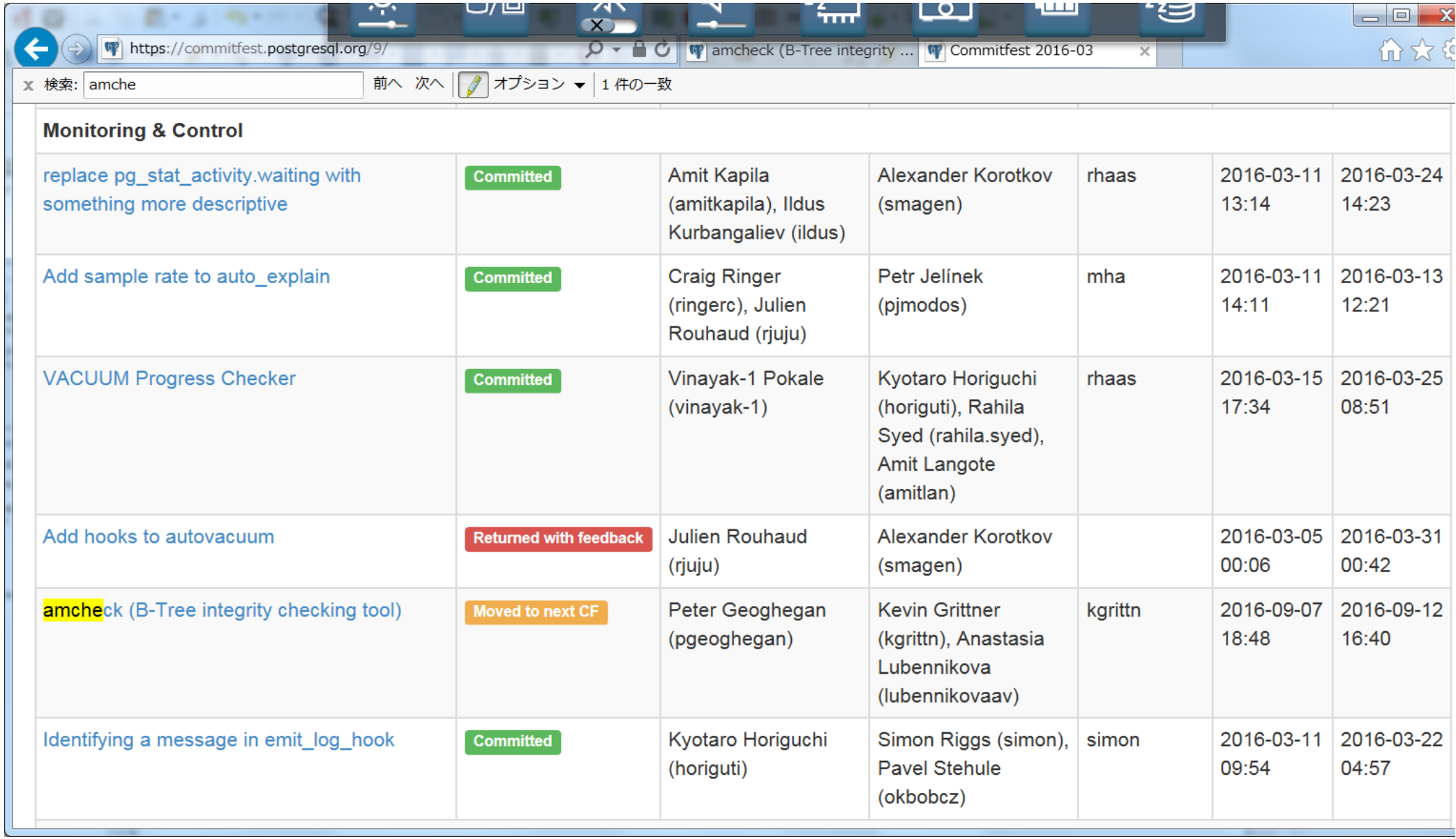
# 課題

- インデックスは相変わらず壊れることがある
- 今使っているインデックスは壊れているのかを知りたい？
- 運用中 / 稼働中にとめずにチェックしたい
- 検査して、それでクラッシュしても困る

# amcheck

- Peter Geoghegan at heroku.com
- 2016 年 3 月に試作初版
- 本体に付属するつもりで作成
- しかし、 9.6 入りはならず

https://github.com/petergeoghegan/amcheck




**Monitoring & Control**

| | | | | | | |
|---|---|---|---|---|---|---|
| replace pg_stat_activity.waiting with something more descriptive | Committed | Amit Kapila (amitkapila), Ildus Kurbangaliev (ildus) | Alexander Korotkov (smagen) | rhaas | 2016-03-11 13:14 | 2016-03-24 14:23 |
| Add sample rate to auto_explain | Committed | Craig Ringer (ringerc), Julien Rouhaud (rjuju) | Petr Jelínek (pjmodos) | mha | 2016-03-11 14:11 | 2016-03-13 12:21 |
| VACUUM Progress Checker | Committed | Vinayak-1 Pokale (vinayak-1) | Kyotaro Horiguchi (horiguti), Rahila Syed (rahila.syed), Amit Langote (amitlan) | rhaas | 2016-03-15 17:34 | 2016-03-25 08:51 |
| Add hooks to autovacuum | Returned with feedback | Julien Rouhaud (rjuju) | Alexander Korotkov (smagen) | | 2016-03-05 00:06 | 2016-03-31 00:42 |
| amcheck (B-Tree integrity checking tool) | Moved to next CF | Peter Geoghegan (pgeoghegan) | Kevin Grittner (kgrittn), Anastasia Lubennikova (lubennikovaav) | kgrittn | 2016-09-07 18:48 | 2016-09-12 16:40 |
| Identifying a message in emit_log_hook | Committed | Kyotaro Horiguchi (horiguti) | Simon Riggs (simon), Pavel Stehule (okbobcz) | simon | 2016-03-11 09:54 | 2016-03-22 04:57 |

# amcheck 機能

- Btree のみ対応
- インデックスを検査する SQL 関数を提供
  - bt_index_check
    - AccessShareLock のみ、親子関係チェック無し
  - bt_index_parent_check
    - AccessExclusiveLock 有
  - 内部で 20 種類くらいの検査
    - （ PostgreSQL の API から出るエラーに加えて、 ）
  - 構造が正しいかを検査する
    - 値が合っているか、ヒープにあるエントリーが欠けていないか、は検査してくれない

# 導入

```
[harukat@o6-harukat amcheck]$ ls
amcheck--0.1.sql  amcheck.control  build   expected    Makefile   sql
amcheck.c         bootstrap.sh     debian  LICENSE.md  README.md  Vagrantfile
[harukat@o6-harukat amcheck]$ make install
[harukat@o6-harukat ~]$ psql db1
psql (9.6.0)
Type "help" for help.

db1=# CREATE TABLE t1 (id int primary key, txt text, ts timestamp(0));
CREATE TABLE
db1=# CREATE INDEX ON t1 (ts);
CREATE INDEX
db1=# CREATE INDEX ON t1 (txt);
CREATE INDEX
db1=# \d t1+
                  Table "public.t1"
 Column |              Type              | Modifiers
--------+--------------------------------+-----------
 id     | integer                        | not null
 txt    | text                           |
 ts     | timestamp(0) without time zone |
Indexes:
    "t1_pkey" PRIMARY KEY, btree (id)
    "t1_ts_idx" btree (ts)
    "t1_txt_idx
```

# 壊す

```
db1=# SELECT unnest, pg_relation_filenode(unnest)
        FROM unnest(ARRAY['t1_txt_idx','t1_pkey', 't1_ts_idx']);
   unnest   | pg_relation_filenode
------------+----------------------
 t1_txt_idx |                16436
 t1_pkey    |                16434
 t1_ts_idx  |                16435
(3 rows)

db1=# CREATE EXTENTION pageinspect;

db1=# SELECT * FROM bt_metap('t1_txt_idx');
 magic  | version | root | level | fastroot | fastlevel
--------+---------+------+-------+----------+-----------
 340322 |       2 |    3 |     1 |        3 |         1
(1 row)

db1=# SELECT * FROM bt_page_stats('t1_ts_idx',3);
 blkno | type | live_items | dead_items | avg_item_size | page_size |
free_size | btpo_prev | btpo_next | btpo | btpo_flags
-------+------+------------+------------+---------------+-----------
+-----------+-----------+-----------+------+------------
     3 | r    |         28 |          0 |            15 |      8192 |
 7596 |         0 |         0 |    1 |          2
(1 row)
```

# 壊れている場合の出力

```
[harukat@o6-harukat ~]$ pg_ctl stop
[harukat@o6-harukat ~]$ bvi $PGDATA/base/16384/16407
[harukat@o6-harukat ~]$ pg_ctl start
[harukat@o6-harukat ~]$ psql db1

db1=# SELECT bt_index_check('t1_pkey');
 bt_index_check
----------------


(1 row)

db1=# SELECT bt_index_parent_check('t1_pkey');
 bt_index_parent_check
-----------------------


(1 row)

db1=# SELECT * FROM bt_index_check('t1_txt_idx');
ERROR:  page order invariant violated for index "t1_txt_idx"
DETAIL:  Lower index tid=(115,10) (points to index tid=(122,1)) higher
index tid=(115,11) (points to index tid=(858806836,23130)) page lsn=0/0.

db1=# SELECT bt_index_parent_check('t1_ts_idx');
ERROR:  right link/left link pair in index "t1_ts_idx" not in mutual
agreement
DETAIL:  Deleted block=4 left block=2 link reported block=3.
```

# 同時実行も大丈夫

- pgbench で繰り返し UPDATE をかけているテーブルに amcheck を 10 秒おき実行
  - amcheck 実行もほぼブロックせず
    - 32 文字 ×100 万件インデックスで 1 秒以内くらい
    - つまり、超巨大インデックスだと、それなり
  - pgbench の遅延も僅か
    - latency average = 32.510 ms
    - latency average = 33.402 ms