

5 февраля 2020 / Москва

pg_repack & deferred constraints

Николай Аверин, Backend Developer



bloat



live tuple

bloat

← dead tuple

bloat

Green	Green
Grey	Green
Grey	Grey
Grey	Green
Green	Grey
Green	Grey
Grey	Green
Green	Grey
Green	Green
Green	Green
Green	Grey

bloat

Green	Green
Grey	Green
Grey	Grey
Grey	Green
Green	Grey
Green	Grey
Grey	Green
Green	Grey
Green	Green
Grey	Green
Green	Grey

Grey	Green
Green	Grey
Grey	Grey
Grey	Grey
Grey	Grey
Grey	Green
Green	Green
Green	Grey
Grey	Green
Green	Grey
Green	Green

Grey	Green
Grey	Grey
Green	Grey
Green	Grey
Green	Green
Grey	Green
Green	Grey
Green	Grey
Green	Green
Grey	Green
Green	Grey

White	White
White	White
White	White
Grey	Green
Green	Green
Green	Green
Grey	Grey
Grey	Green
Green	Grey
Grey	Green
Green	Green

bloat

Green	Green
Grey	Green
Grey	Grey
Grey	Green
Green	Grey
Green	Grey
Grey	Green
Green	Grey
Green	Green
Grey	Green
Green	Grey

Grey	Grey
Grey	Grey
Grey	Grey
Grey	Grey
Grey	Grey
Grey	Grey
Grey	Grey
Grey	Grey
Grey	Grey
Grey	Grey
Grey	Grey

Grey	Green
Grey	Grey
Green	Grey
Green	Grey
Green	Grey
Grey	Green
Green	Grey
Green	Grey
Grey	Green
Green	Grey
Green	Grey

White	White
White	White
White	White
Grey	Green
Green	Green
Green	Green
Grey	Grey
Grey	Green
Green	Grey
Grey	Green
Green	Green

bloat

Green	Green
White	Green
White	White
White	White
Green	White
Green	White
White	Green
Green	White
Green	Green
White	Green
Green	White

White	White
White	White
White	White
White	White
White	White
White	White
White	White
White	White
White	White
White	White
White	White

White	Green
White	White
Green	White
Green	White
Green	White
White	Green
Green	White
Green	White
White	Green
Green	White
Green	White

White	White
White	White
White	White
White	Green
Green	Green
Green	Green
White	White
White	Green
Green	White
White	Green
Green	Green

А у меня есть bloat?

sql scripts - pgexperts*

extension - pgstattuple

5-15 % bloat is OK

https://github.com/pgexperts/pgx_scripts/tree/master/bloat

Способы борьбы с bloat-ом

0. AUTOVACUUM

Способы борьбы с bloat-ом

0. AUTOVACUUM

1. VACUUM FULL

Способы борьбы с bloat-ом

0. AUTOVACUUM

1. VACUUM FULL - `AccessExclusiveLock`

Способы борьбы с bloat-ом

0. AUTOVACUUM

1. VACUUM FULL - `AccessExclusiveLock`

2. CLUSTER

Способы борьбы с bloat-ом

0. AUTOVACUUM

1. VACUUM FULL - `AccessExclusiveLock`

2. CLUSTER - `AccessExclusiveLock`

Способы борьбы с bloat-ом

0. AUTOVACUUM

1. VACUUM FULL - AccessExclusiveLock

2. CLUSTER - AccessExclusiveLock

3. REINDEX

Способы борьбы с bloat-ом

0. AUTOVACUUM

1. VACUUM FULL - AccessExclusiveLock

2. CLUSTER - AccessExclusiveLock

3. REINDEX - AccessExclusiveLock

Способы борьбы с bloat-ом

0. AUTOVACUUM

1. VACUUM FULL - `AccessExclusiveLock`

2. CLUSTER - `AccessExclusiveLock`

3. REINDEX - `AccessExclusiveLock` [CONCURRENTLY - pg 12]

Способы борьбы с bloat-ом

0. AUTOVACUUM

1. VACUUM FULL - *AccessExclusiveLock*

2. CLUSTER - *AccessExclusiveLock*

3. REINDEX - *AccessExclusiveLock* [CONCURRENTLY - pg 12]

4. CREATE INDEX CONCURRENTLY

Способы борьбы с bloat-ом

0. AUTOVACUUM

1. VACUUM FULL - **AccessExclusiveLock**

2. CLUSTER - **AccessExclusiveLock**

3. REINDEX - **AccessExclusiveLock** [CONCURRENTLY - pg 12]

4. CREATE INDEX CONCURRENTLY - **ShareUpdateExclusiveLock**

Способы борьбы с bloat-ом

0. AUTOVACUUM

1. VACUUM FULL - **AccessExclusiveLock**

2. CLUSTER - **AccessExclusiveLock**

3. REINDEX - **AccessExclusiveLock** [CONCURRENTLY - pg 12]

4. CREATE INDEX CONCURRENTLY - **ShareUpdateExclusiveLock**

5. pg_repack, pgcompact, pgcompacttable, ...

Алгоритм pg_repack

bloated table



Алгоритм pg_repack

bloated table



log table

Алгоритм pg_repack

bloated table



i

trigger

log table

Алгоритм pg_repack

bloated table



trigger

log table

new table

Алгоритм pg_repack

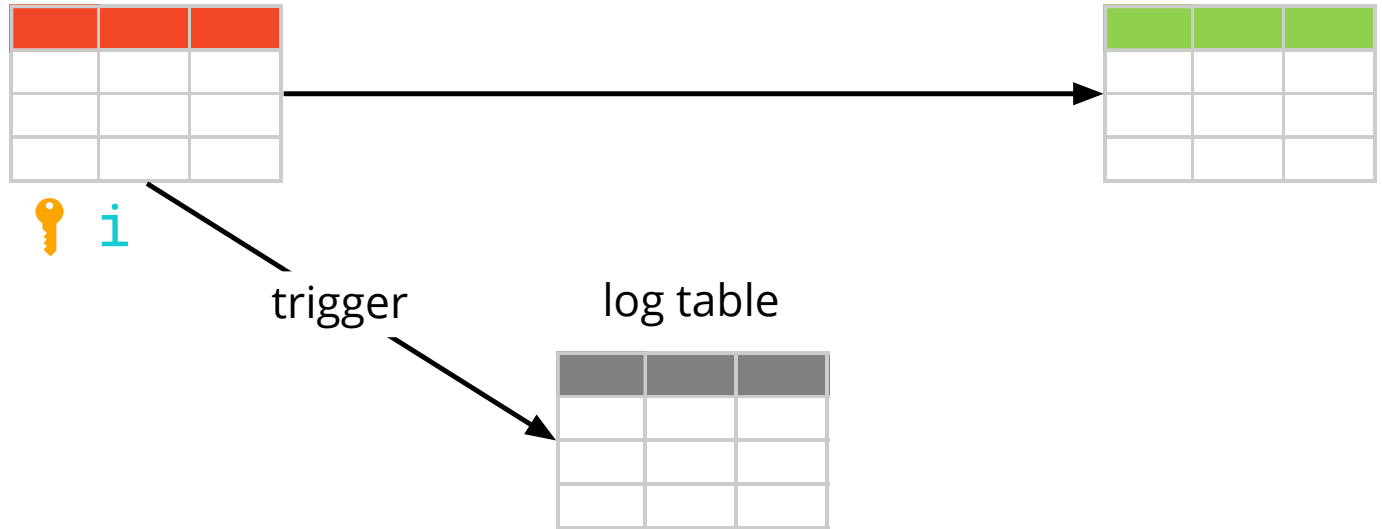
bloated table



trigger

log table

new table



Алгоритм pg_repack

bloated table



trigger

log table

new table

i

Алгоритм pg_repack

bloated table

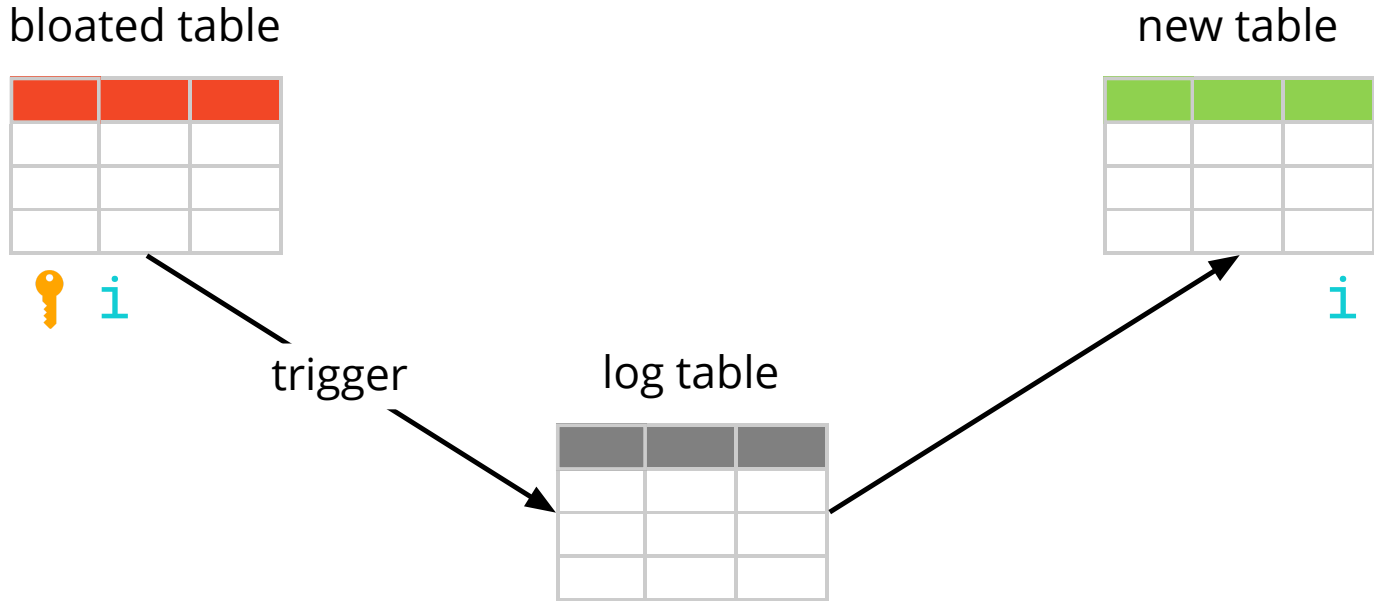


trigger

new table

i

log table



Алгоритм pg_repack

new table



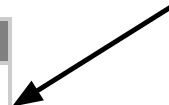
bloated table

i



log table

trigger



Алгоритм pg_repack

new table



Попытка №1

```
$ ./pg_repack -t widgets -o board_id,id  
INFO: repacking table "widgets"  
ERROR: query failed: ERROR:  
        duplicate key value violates unique constraint "index_16508"  
DETAIL: Key (board_id, z_index)=(100500, 42) already exists.
```

deferred constraints

```
create table cars
(  
  name text constraint pk_cars primary key,  
  ord integer not null constraint uk_cars unique  
);
```

name	ord
audi	1
bmw	2
toyota	3

deferred constraints

```
begin;  
update cars set ord = 2 where name = 'audi';  
update cars set ord = 1 where name = 'bmw';  
commit;
```


deferred constraints

```
begin;  
update cars set ord = 2 where name = 'audi';  
update cars set ord = 1 where name = 'bmw';  
commit;
```

[23505] ERROR: duplicate key value violates unique constraint "uk_cars" Detail: Key (ord)=(2) already exists.

deferred constraints

```
create table cars
(
  name text constraint pk_cars primary key,
  ord integer not null constraint uk_cars unique
  deferrable initially deferred
);
```

deferred constraints

	NOT DEFERRABLE	INITIALLY IMMEDIATE	INITIALLY DEFERRED
CHECK	row	row	row
NOT NULL	row	row	row
UNIQUE	row	statement	transaction
PRIMARY KEY	row	statement	transaction
FOREIGN KEY	statement	statement	transaction
EXCLUDE	statement	statement	transaction

deferred constraints

- Меньше операций
- Проще код
- Мешает оптимизатору
- Труднее дебажить

Алгоритм pg_repack

bloated table

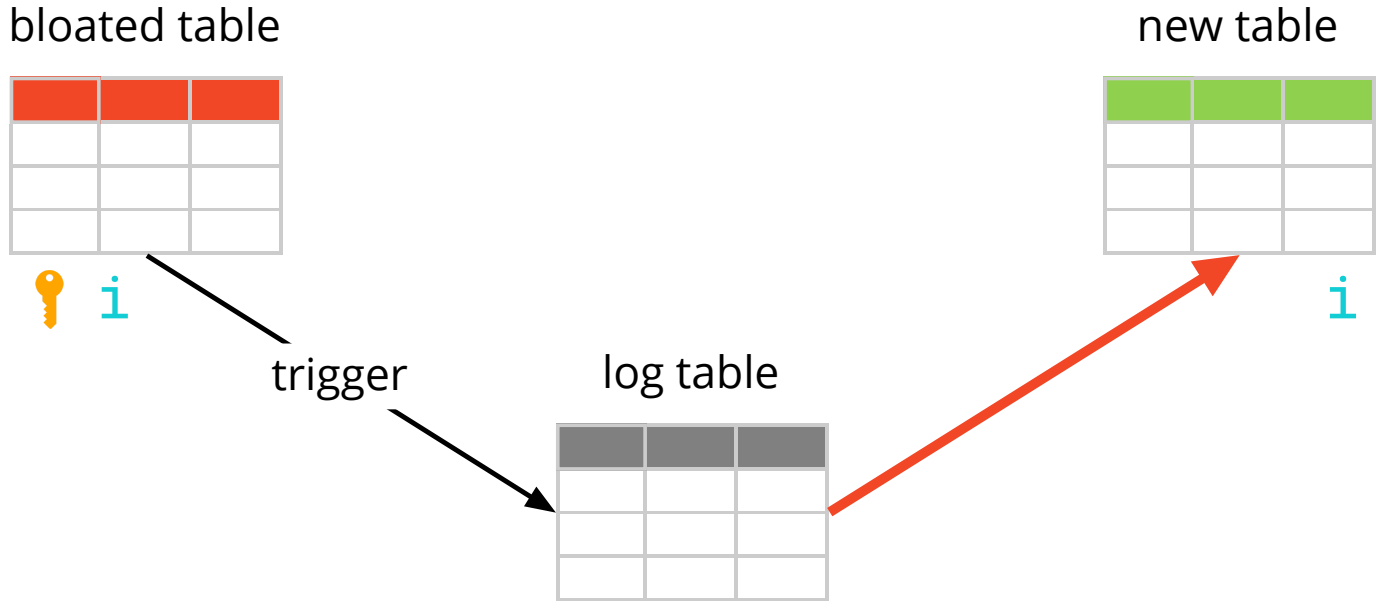


trigger

log table

new table

i



Алгоритм pg_repack

bloated table



trigger

log table

new table

i



Попытка №2

```
CREATE UNIQUE INDEX CONCURRENTLY  
uk_widgets__board_z_index_immediate ON widgets (board_id, z_index);  
  
-- run pg_repack  
  
DROP INDEX CONCURRENTLY uk_widgets__board_z_index_immediate;
```

pg_repack

CONSTRAINTS --> INDEXES

pg_repack

CONSTRAINTS --> CONSTRAINTS

pg_repack

CONSTRAINTS --> CONSTRAINTS

```
INSERT INTO test_table (val) VALUES (1);
FOR i IN 1..10000 LOOP
BEGIN
    INSERT INTO test_table VALUES (1) RETURNING id INTO v_id;
    UPDATE test_table set val = i where id = v_id;
    COMMIT;
END;
END LOOP;
```

Попытка №3

```
$. /pg_repack -t widgets -o board_id,id  
INFO: repacking table "widgets"  
ERROR: query failed: ERROR:  
      duplicate key value violates unique constraint "index_16508"  
DETAIL: Key (board_id, z_index)=(100500, 42) already exists.
```

Ah shit, here we go again.

Алгоритм pg_repack

bloated table



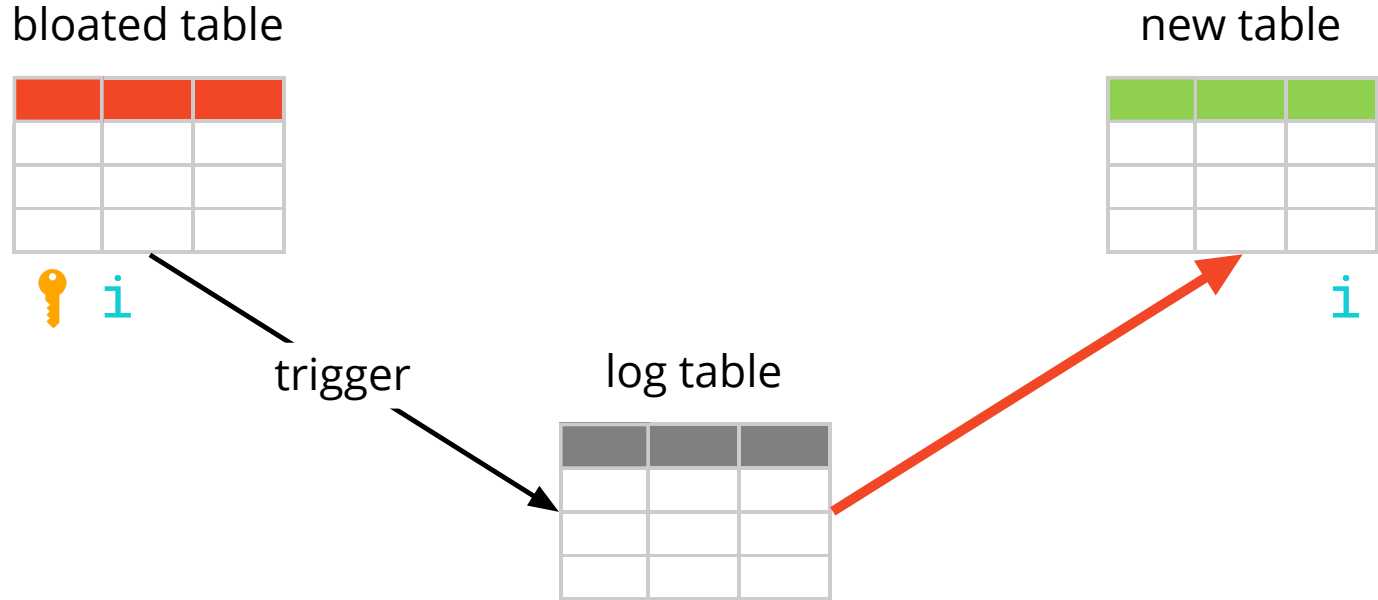
i

trigger

new table

i

log table



pg_repack

```
for (;;)
{
    num = apply_log(connection, table, APPLY_COUNT);

    if (num > MIN_TUPLES_BEFORE_SWITCH)
        continue; /* there might be still some tuples, repeat. */
    ...
}
```

pg_repack

CONSTRAINTS --> CONSTRAINTS

```
INSERT INTO test_table (val) VALUES (1);
FOR i IN 1..10000 LOOP
BEGIN
    INSERT INTO test_table VALUES (1) RETURNING id INTO v_id;
    UPDATE test_table set val = i where id = v_id;
    COMMIT;
END;
END LOOP;
```

pg_repack

CONSTRAINTS --> CONSTRAINTS

```
INSERT INTO test_table (val) VALUES (1);
FOR i IN 1..10000 LOOP
BEGIN
    INSERT INTO test_table VALUES (1) RETURNING id INTO v_id;
    UPDATE test_table set val = i where id = v_id;
    UPDATE test_table set val = i where id = v_id;
    COMMIT;
END;
END LOOP;
```

Попытка №4



Выводы

1. Мониторьте ваш bloat

Выводы

1. Мониторьте ваш bloat
2. Настраивайте autovacuum

Выводы

1. Мониторьте ваш bloat
2. Настраивайте autovacuum
3. Не бойтесь использовать и дорабатывать сторонние решения

Спасибо за внимание!

Николай Аверин

averin@miro.com

vk.com/nikolai_averin