# Нечеткий поиск по тексту в PostgreSQL с помощью pg_trgm



5 июня 2017

В продолжение темы о полнотекстовом поиске в PostgreSQL хотелось бы также рассказать о расширении под названием pg_trgm. Данное расширение предназначено для поиска текстовых документов по триграммам, то есть, всем подпоследовательностям из трех букв, входящих в индексируемый текст. На практике такой поиск интересен, помимо прочего, тем, что позволяет находить документы по запросам, содержащим опечатки.

Итак, пример создания индекса:

```
-- расширение входит в состав PostgreSQL
CREATE EXTENSION pg_trgm;
-- также можно использовать gist
CREATE INDEX articles_trgm_idx ON articles
  USING gin (title gin_trgm_ops);
```

С помощью процедуры show_trgm можно посмотреть получившиеся триграммы:

```
SELECT show_trgm(title) FROM articles LIMIT 3;
```

Пример ответа:

```
-[ RECORD 1 ]----------------------------------------------------
show_trgm | {"  a"," ac",acc,ble,cce,ces,com,eco,ess,ibl,ing,lec,mp...
-[ RECORD 2 ]----------------------------------------------------
show_trgm | {"  a"," an",ana,arc,chi,his,ism,nar,rch,"sm "}
-[ RECORD 3 ]----------------------------------------------------
show_trgm | {"  a"," af",afg,anh,ani,fgh,gha,han,his,ist,nhi,nis,or...
```

Поиск с использованием построенного индекса осуществляется так:

```
SELECT title, similarity(title, 'Straustrup') FROM articles
  WHERE title % 'Straustrup';
```

Результат:

```
-[ RECORD 1 ]----------
title      | Bjarne Stroustrup
similarity | 0.35
```

Как видите, документ был найден, невзирая на опечатку в поисковом запросе. Точно так же, к примеру, по запросу «phone» могут быть найдены документы, содержащие слово «iPhone», что при использовании обычного полнотекстового поиска не будет работать.

При поиске с использованием pg_trgm возвращаются документы, чей уровень similarity запросу выше определенного значения. По умолчанию это значение равно 0.3. Узнать текущее пороговое значение можно при помощи процедуры show_limit, а изменить его в рамках сессии — с помощью процедуры set_limit:

```
SELECT show_limit(), set_limit(0.4);
```

Результат:

```
 show_limit | set_limit
------------+-----------
        0.3 |       0.4
```

Помимо возможности поиска по запросам с опечатками pg_trgm также может быть использован для ускорения LIKE/ILIKE-запросов, а также поиска по регулярным выражениям:

```
EXPLAIN SELECT title FROM articles WHERE title LIKE '%Stroustrup%';
-- или:
EXPLAIN SELECT title FROM articles WHERE title ~* 'Stroustrup';
```

Пример плана запроса:

```
Bitmap Heap Scan on articles  (cost=60.02..71.40 rows=3 width=16)
  Recheck Cond: ((title)::text ~~ '%Stroustrup%'::text)
  -> Bitmap Index Scan on articles_trgm_idx  (cost=0.00..60.02 rows...
       Index Cond: ((title)::text ~~ '%Stroustrup%'::text)
```

Такое вот полезное расширение! А пользуетесь ли вы pg_trgm и если да, то довольны ли им?

*Метки: PostgreSQL, СУБД.*