

План запроса



Вывод команды EXPLAIN

План – это дерево в классическом математическом представлении, поэтому чтение плана выполняется снизу вверх.

Пример плана, который выводится командой EXPLAIN:

```
adb=# EXPLAIN SELECT * FROM table2 where id < 101;
```

QUERY PLAN

```
-----  
Gather Motion 4:1 (slice1; segments: 4) (cost=0.00..455.07 rows=200000 width=30)
```

```
  -> Seq Scan on table2 (cost=0.00..432.21 rows=50000 width=30)
```

```
    Filter: (id1 < 101)
```

```
Optimizer: Pivotal Optimizer (GPORCA) version 3.80.0
```



Вывод команды EXPLAIN

QUERY PLAN

Gather Motion 4:1 (slice1; segments: 4) (cost=0.00..455.07 rows=200000 width=30)

-> Seq Scan on table2 (cost=0.00..432.21 rows=50000 width=30)

Filter: (id1 < 101)

Optimizer: Pivotal Optimizer (GPORCA) version 3.80.0

- Первая строка снизу указывает на то, какой оптимизатор использовался при построении плана.
- В данном случае использовался родной оптимизатор GPORCA.



Вывод команды EXPLAIN

Gather Motion 4:1 (slice1; segments: 4) (cost=0.00..455.07 rows=200000 width=30)

-> Seq Scan on table2 (cost=0.00..432.21 rows=50000 width=30)

Filter: (id1 < 101)

Optimizer: Pivotal Optimizer (GPORCA) version 3.80.0

- Вторая строка снизу – узел операции простого сканирования файла таблицы (выборка данных).
- Для узлов приводятся:
 1. Название операции.
 2. Объект.
 3. Условия (в данном примере filter для условия WHERE).
 4. Параметры узла в виде набора значений в круглых скобках.

Уровень узла в дереве определяется отступом (табуляцией), а не порядком следования в выводе плана.



Лабораторная работа 13. Планы запросов

13.1 Создайте таблицу table1 со следующими параметрами:

- Поля: id1 int, id2 int, gen1 text, gen2 text.
- Первичным ключом сделайте поля id1, id2, gen1.

13.2 Создайте таблицу table2 со следующими параметрами:

- Возьмите набор полей table1 с помощью директивы LIKE.

13.3 Проверить, какое количество внешних таблиц присутствует в бд



Лабораторная работа 13. Планы запросов

13.4 Сгенерируйте данные и вставьте их в обе таблицы(200 тысяч и 400 тысяч значений соответственно):

```
insert into table1 select gen,gen, gen::text || 'text1', gen::text || 'text2' from  
generate_series(1,200000) gen;  
insert into table2 select gen,gen, gen::text || 'text1', gen::text || 'text2' from  
generate_series(1,400000) gen;
```

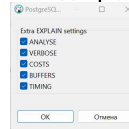
13.5 С помощью директивы EXPLAIN просмотрите план соединения таблиц table1 и table2 по ключу id1.



Лабораторная работа 13. Планы запросов

13.6 Используя таблицы `table1` и `table2` реализовать план запроса:

- План запроса встроенного инструмента dbeaver.
- С помощью директивы EXPLAIN.



13.7 Реализовать запросы с использованием Joins, Group by, вложенного подзапроса. Экспортировать план в файл, используя

psql -qAt -f explain.sql > analyze.json

Для достижения наилучших результатов используйте EXPLAIN (ANALYZE, COSTS, VERBOSE, BUFFERS, FORMAT JSON)

13.8 Сравнить полученные результаты а пункте 13.6 локально с результатом на сайте <https://tatiyants.com/pev/#/plans/new> сделайте вывод.

В отчете предоставить: ERD-диаграмму; вывести первые 20 значений каждой таблицы; предоставить результат выполнения запросов Joins, Group by, вложенного подзапроса; результаты работы профилирования EXPLAIN (ANALYZE, COSTS, VERBOSE, BUFFERS, FORMAT JSON).