

解读EXPLAIN执行计划

朱贤文 成都文武信息技术有限公司

主题

公司介绍

获取执行计划

基本内容解读

理解常见的操作

书写高性能SQL

有问必答

关于文武信息技术

- 成立于2014年6月，天府软件园B区7栋611
- 专注于高性能数据库的研发、销售和服务
- 国内提供PostgreSQL数据库商业支持和服务的专业厂商
- 服务的客户涉及银行，金融，通信，生产制造，互联网等行业

WwIT
驱动数据 成就未来

获取执行计划

EXPLAIN SELECT * FROM tenk1 WHERE unique1 < 100;

QUERY PLAN

Bitmap Heap Scan on tenk1 (cost=5.07..229.20 rows=101 width=244)

Recheck Cond: (unique1 < 100)

-> Bitmap Index Scan on tenk1_unique1 (cost=0.00..5.04 rows=101 width=0)

Index Cond: (unique1 < 100)

获取执行计划

EXPLAIN SELECT * FROM tenk1 WHERE unique1 < 100;

EXPLAIN analyze SELECT * FROM tenk1 WHERE unique1 < 100;

EXPLAIN update xxxxx;

EXPLAIN insert xxxxx;

EXPLAIN delete xxxxx;

执行计划

QUERY PLAN

Bitmap Heap Scan on tenk1 (cost=5.07..229.20 rows=101 width=244)

Recheck Cond: (unique1 < 100)

-> **Bitmap Index Scan on tenk1_unique1 (cost=0.00..5.04 rows=101 width=0)**

Index Cond: (unique1 < 100)

执行计划

QUERY PLAN

Bitmap Heap Scan on tenk1 (**cost=5.07..229.20** rows=101 width=244)

Recheck Cond: (unique1 < 100)

-> Bitmap Index Scan on tenk1_unique1 (**cost=0.00..5.04** rows=101 width=0)

Index Cond: (unique1 < 100)

执行计划

```
explain analyze select * from t limit 100;
```

QUERY PLAN

```
-----  
Limit (cost=0.00..9.33 rows=100 width=608) (actual time=0.008..0.152 rows=100 loops=1)  
  -> Seq Scan on t (cost=0.00..93333.86 rows=999986 width=608) (actual time=0.007..0.133 rows=100 loops=1) Total  
runtime: 0.181 ms (3 rows)
```


执行计划

```
explain select * from test where id = 50;
```

```
QUERY PLAN
```

```
-----  
Index Scan using test_pkey on test (cost=0.28..8.29 rows=1 width=36)  
  Index Cond: (id = 50)
```

执行计划

```
#disable index scan
```

```
explain select * from test where id = 50;
```

```
QUERY PLAN
```

```
-----
```

```
Bitmap Heap Scan on test (cost=4.28..8.30 rows=1 width=13)
```

```
  Recheck Cond: (id = 50)
```

```
    -> Bitmap Index Scan on test_pkey (cost=0.00..4.28 rows=1 width=0)
```

```
      Index Cond: (id = 50)
```

执行计划

```
#disable index scan and bitmap scan  
explain select * from test where id = 50;
```

QUERY PLAN

```
-----  
Seq Scan on test (cost=0.00..18.50 rows=1 width=13)  
  Filter: (id = 50)
```

执行计划

Index Scan using test_pkey on test (cost=0.28..8.29 rows=1 width=36)

Bitmap Heap Scan on test (cost=4.28..8.30 rows=1 width=13)

Seq Scan on test (cost=0.00..18.50 rows=1 width=13)

执行计划

Nested Loop (cost=0.00..10715.90 rows=26284 width=4449) (actual time=0.054..291.131 rows=26284 loops=1)
-> Index Scan using books_index_title on books (cost=0.00..3306.28 rows=26284 width=3357) (actual time=0.033..50.773 rows=26284 loops=1)
-> Index Scan using categories_pkey on categories (cost=0.00..0.27 rows=1 width=1092) (actual time=0.002..0.003 rows=1 **loops=26284**)
Index Cond: (categories.id = books.category_id) Total runtime: 312.212 ms

执行计划 seq scan

```
explain analyze select * from pg_class;
```

QUERY PLAN

Seq Scan on pg_class (cost=0.00..10.92 rows=292 width=202) (actual time=0.009..0.049 rows=295 loops=1)

```
explain analyze select * from pg_class limit 2;
```

QUERY PLAN

Limit (cost=0.00..0.07 rows=2 width=202) (actual time=0.014..0.014 rows=2 loops=1)

-> Seq Scan on pg_class (cost=0.00..10.92 rows=292 width=202) (actual time=0.009..0.009 rows=2 loops=1)

执行计划 index scan

```
explain analyze select * from pg_class where oid = 1247;
```

QUERY PLAN

Index Scan using pg_class_oid_index on pg_class (cost=0.15..8.17 rows=1 width=202) (actual time=0.007..0.007 rows=1 loops=1)

Index Cond: (oid = 1247::oid)

```
explain analyze select * from pg_class order by oid limit 10;
```

QUERY PLAN

Limit (cost=0.15..1.67 rows=10 width=206) (actual time=0.017..0.029 rows=10 loops=1)

-> **Index Scan using pg_class_oid_index on pg_class** (cost=0.15..44.53 rows=292 width=206) (actual time=0.014..0.026 rows=10 loops=1)

执行计划Index Scan

```
explain analyze select id from test order by id asc limit 10;
```

QUERY PLAN

Limit (cost=0.29..0.55 rows=10 width=4) (actual **time**=0.039..0.042 rows=10 loops=1)

-> **Index Only Scan** using test_pkey on test (cost=0.29..2604.29 rows=100000 width=4) (actual **time**=0.036..0.038 rows=10 loops=1) Heap Fetches: 0

```
explain analyze select * from pg_class where oid < 1247 order by oid desc limit 10;
```

QUERY PLAN

Limit (cost=0.15..4.03 rows=10 width=206) (actual **time**=0.012..0.026 rows=10 loops=1)

-> **Index Scan Backward** using pg_class_oid_index on pg_class (cost=0.15..37.84 rows=97 width=206) (actual **time**=0.009..0.022 rows=10 loops=1)

Index Cond: (oid < 1247::oid)

执行计划bitmap scan

```
explain analyze select * from test where i < 100000;
```

QUERY PLAN

Bitmap Heap Scan on test (cost=4.37..39.99 rows=10 width=8) (actual time=0.025..0.110 rows=13 loops=1)

Recheck Cond: (i < 100000)

-> **Bitmap Index Scan** on i1 (cost=0.00..4.37 rows=10 width=0) (actual time=0.013..0.013 rows=13 loops=1)

Index Cond: (i < 100000)

执行计划

explain analyze select * from test where i < 5000000 or i > 950000000;

QUERY PLAN

Bitmap Heap Scan on test (cost=107.36..630.60 rows=5323 width=8) (actual time=1.023..4.353 rows=5386 loops=1)

Recheck Cond: ((i < 5000000) **OR** (i > 950000000))

-> **BitmapOr** (cost=107.36..107.36 rows=5349 width=0) (actual time=0.922..0.922 rows=0 loops=1)

-> **Bitmap Index Scan** on i1 (cost=0.00..12.25 rows=527 width=0) (actual time=0.120..0.120 rows=491

loops=1) **Index** Cond: (i < 5000000)

-> **Bitmap Index Scan** on i1 (cost=0.00..92.46 rows=4822 width=0) (actual time=0.799..0.799 rows=4895

loops=1) **Index** Cond: (i > 950000000)

执行计划

Function Scan

Sort

Limit

HashAggregate

Hash Join / Hash

Nested Loop

Merge Join

Hash Left Join

Hash Right Join

Merge Left Join

Merge Right Join

Nested Loop Left Join

WwIT
驱动数据 成就未来