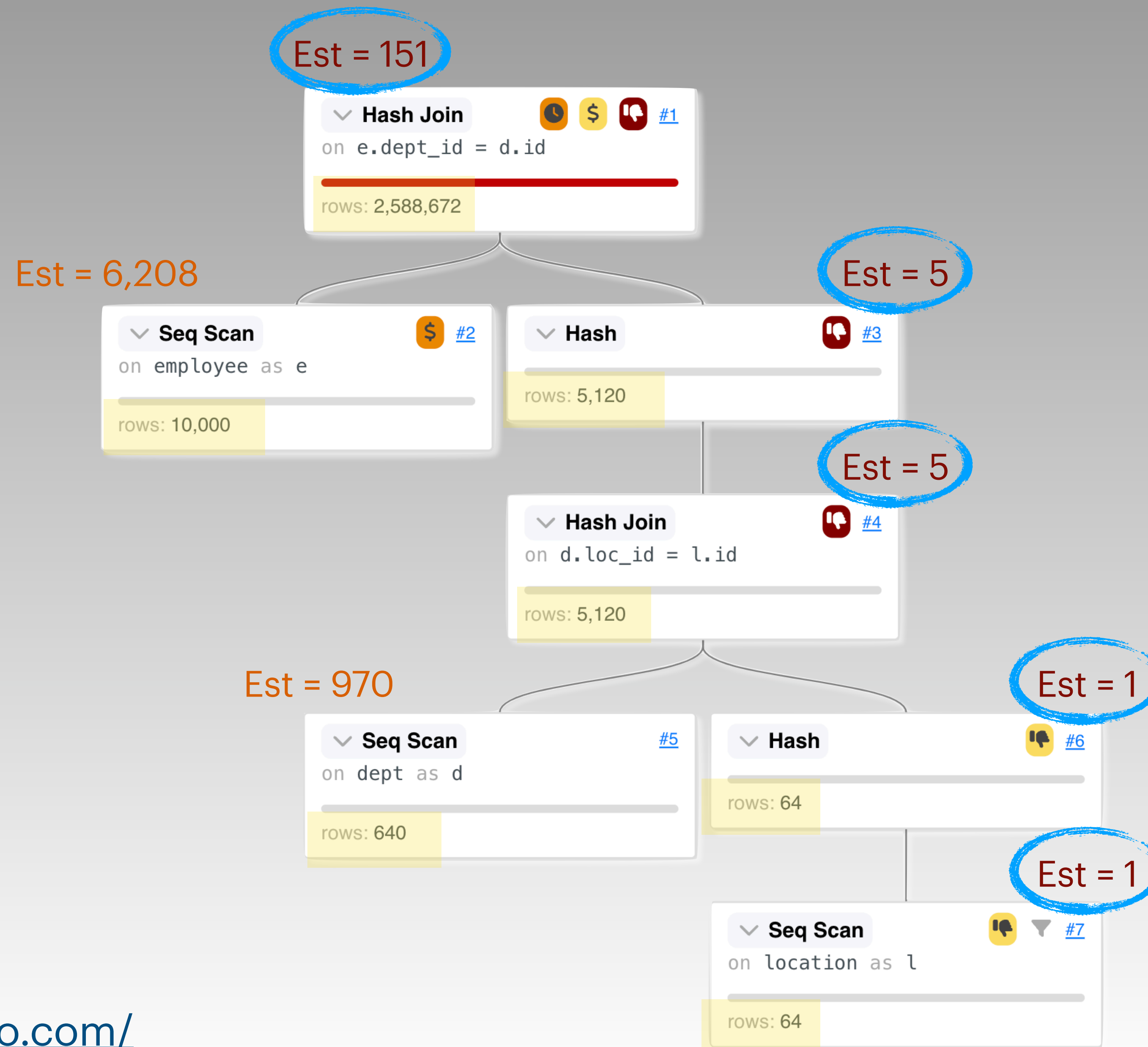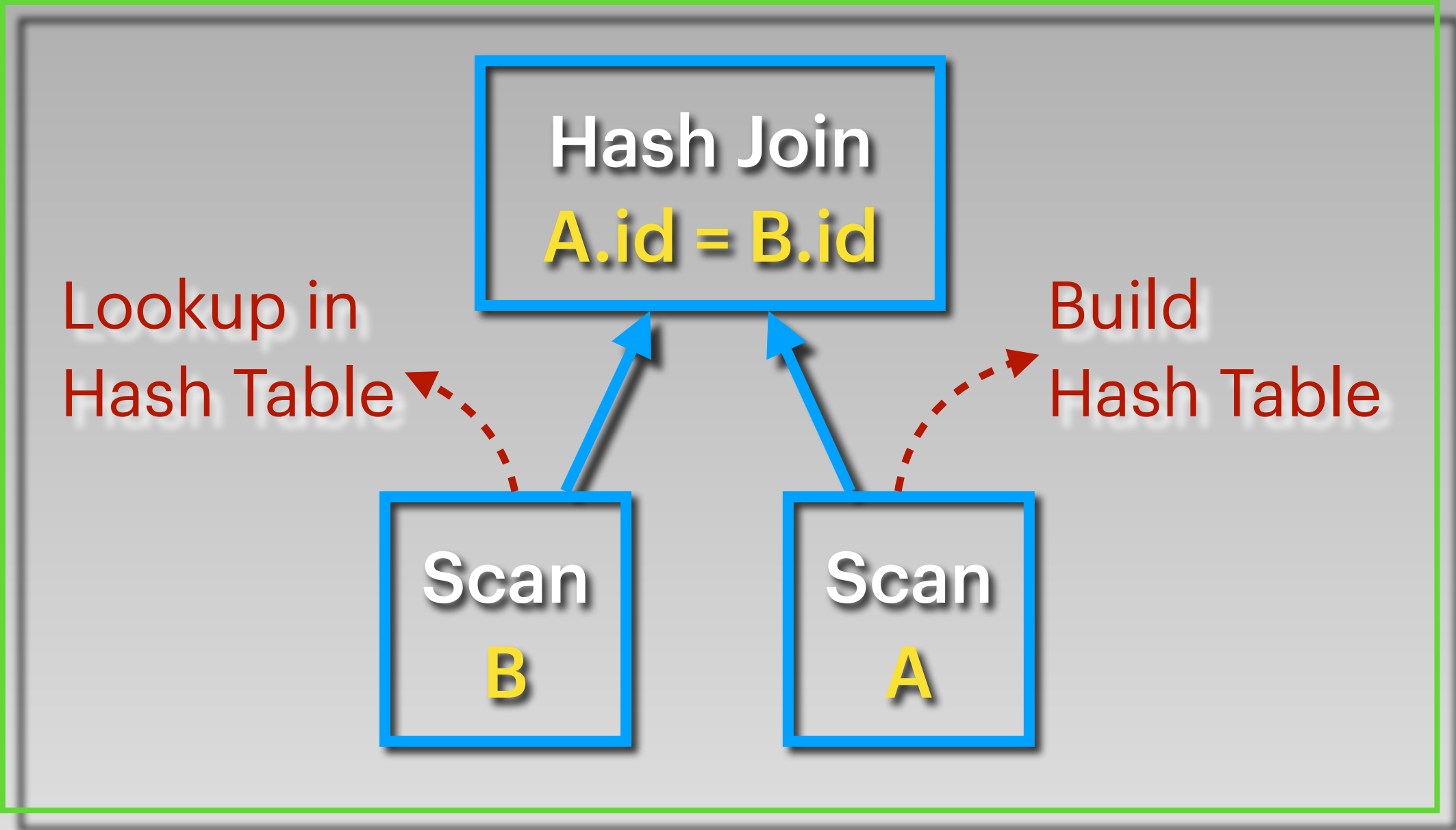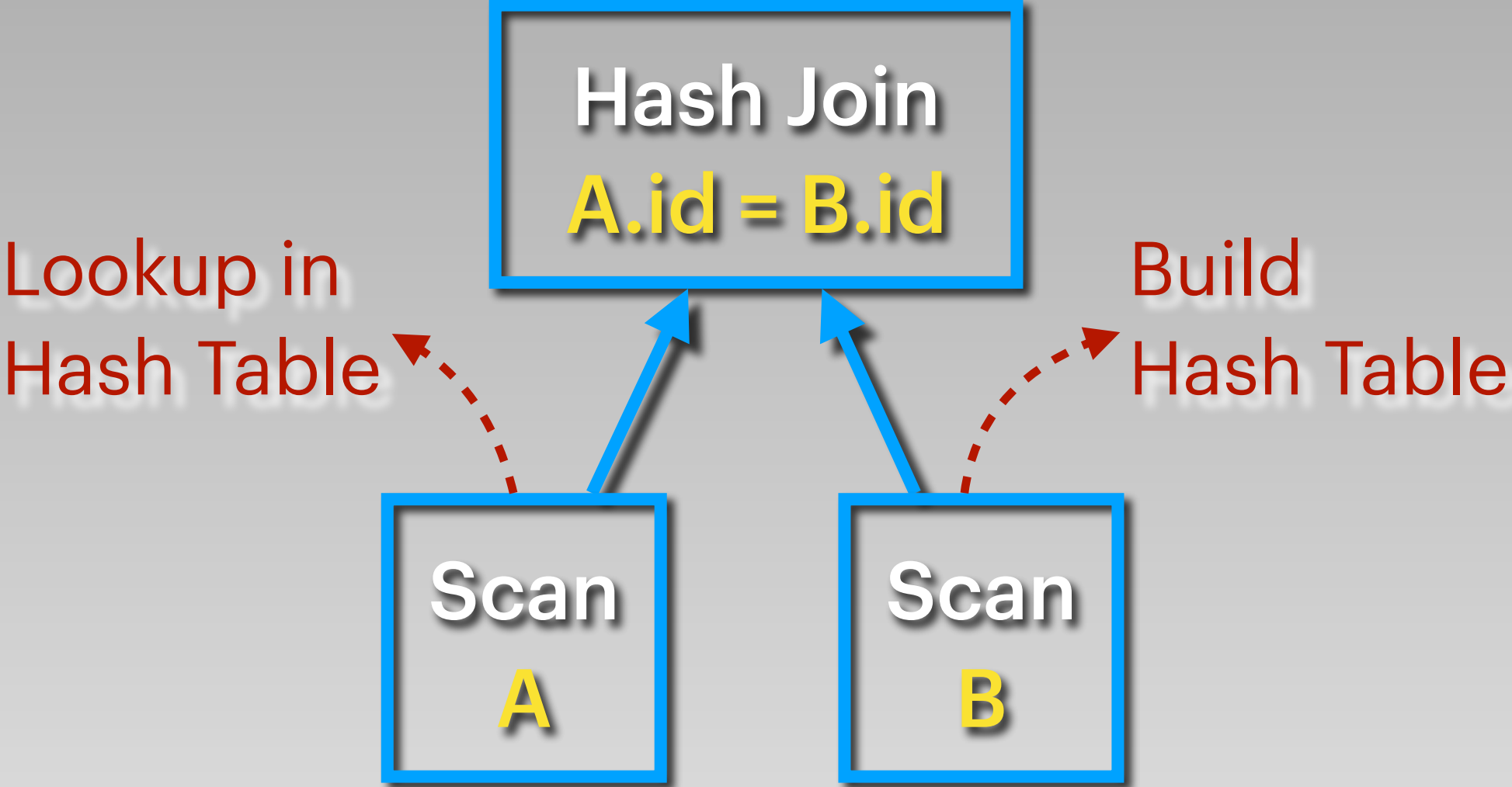```
EXPLAIN ANALYZE SELECT e.name, d.name FROM employee e JOIN dept d ON e.dept_id = d.id JOIN location l ON d.loc_id = l.id
WHERE country = 'UK' AND city = 'London';

                                                    QUERY PLAN

 Hash Join (cost=41.71..192.62 rows=151 width=96) (actual time=2.082..130.412 rows=2588672 loops=1)
   Hash Cond: (e.dept_id = d.id)
   -> Seq Scan on employee e (cost=0.00..126.08 rows=6208 width=52) (actual time=0.035..0.603 rows=10000 loops=1)
   -> Hash (cost=41.65..41.65 rows=5 width=52) (actual time=2.059..2.061 rows=5120 loops=1)
         Buckets: 8192 (originally 1024) Batches: 1 (originally 1) Memory Usage: 288kB
         -> Hash Join (cost=18.26..41.65 rows=5 width=52) (actual time=0.103..1.087 rows=5120 loops=1)
               Hash Cond: (d.loc_id = l.id)
               -> Seq Scan on dept d (cost=0.00..19.70 rows=970 width=56) (actual time=0.010..0.089 rows=640 loops=1)
               -> Hash (cost=18.25..18.25 rows=1 width=4) (actual time=0.079..0.079 rows=64 loops=1)
                     Buckets: 1024 Batches: 1 Memory Usage: 11kB
                     -> Seq Scan on location l (cost=0.00..18.25 rows=1 width=4) (actual time=0.023..0.059 rows=64 loops=1)
                           Filter: (((country)::text = 'UK'::text) AND ((city)::text = 'London'::text))
                           Rows Removed by Filter: 96
 Planning Time: 0.451 ms
 Execution Time: 192.303 ms
(15 rows)
```

Est = 151

Hash Join  #1
on e.dept_id = d.id
rows: 2,588,672

Est = 6,208

Seq Scan  #2
on employee as e
rows: 10,000

Est = 5

Hash  #3
rows: 5,120

Est = 5

Hash Join  #4
on d.loc_id = l.id
rows: 5,120

Est = 970

Seq Scan  #5
on dept as d
rows: 640

Est = 1

Hash  #6
rows: 64

Est = 1

Seq Scan  #7
on location as l
rows: 64

https://explain.dalibo.com/

Amr Elhelw's
TECH
VAULT

```
SELECT ... FROM A JOIN B ON A.id = B.id;
```

Hash Join
A.id = B.id

Lookup in
Hash Table

Build
Hash Table

Scan
A

Scan
B

Hash Join
A.id = B.id

Lookup in
Hash Table

Build
Hash Table

Scan
B

Scan
A

	Estimated no. of rows	Actual no. of rows
Table A	100	200,000
Table B	5,000	1,000

Amr Elhelw's
TECH
VAULT

```
SELECT relname, reltuples, relpages FROM pg_class WHERE relname IN ('location','dept','employee');

  relname  | reltuples | relpages
-----------+-----------+----------
 employee  |        -1 |        0
 dept      |        -1 |        0
 location  |        -1 |        0
(3 rows)

SELECT tablename, attname, null_frac, n_distinct FROM pg_stats WHERE tablename IN ('location','dept','employee');

 tablename | attname | null_frac | n_distinct
-----------+---------+-----------+------------
(0 rows)

ANALYZE;

SELECT relname, reltuples, relpages FROM pg_class WHERE relname IN ('location','dept','employee');

  relname  | reltuples | relpages
-----------+-----------+----------
 dept      |       640 |        4
 employee  |     10000 |       64
 location  |       160 |        2
(3 rows)
```

```
EXPLAIN ANALYZE SELECT e.name, d.name FROM employee e JOIN dept d ON e.dept_id = d.id JOIN location l ON d.loc_id = l.id
WHERE country = 'UK' AND city = 'London';

                                                          QUERY PLAN
------------------------------------------------------------------------------------------------------------------------
 Hash Join  (cost=103.98..19007.99 rows=1664000 width=20) (actual time=2.646..125.921 rows=2588672 loops=1)
   Hash Cond: (e.dept_id = d.id)
   ->  Seq Scan on employee e  (cost=0.00..164.00 rows=10000 width=17) (actual time=0.036..0.529 rows=10000 loops=1)
   ->  Hash  (cost=62.38..62.38 rows=3328 width=11) (actual time=2.595..2.597 rows=5120 loops=1)
         Buckets: 8192 (originally 4096)  Batches: 1 (originally 1)  Memory Usage: 288kB
         ->  Hash Join  (cost=18.40..62.38 rows=3328 width=11) (actual time=0.301..1.436 rows=5120 loops=1)
               Hash Cond: (l.id = d.loc_id)
               ->  Seq Scan on location l  (cost=0.00..4.40 rows=26 width=4) (actual time=0.021..0.055 rows=64 loops=1)
                     Filter: (((country)::text = 'UK'::text) AND ((city)::text = 'London'::text))
                     Rows Removed by Filter: 96
               ->  Hash  (cost=10.40..10.40 rows=640 width=15) (actual time=0.268..0.269 rows=640 loops=1)
                     Buckets: 1024  Batches: 1  Memory Usage: 39kB
                     ->  Seq Scan on dept d  (cost=0.00..10.40 rows=640 width=15) (actual time=0.013..0.133 rows=640 loops=1)
 Planning Time: 0.498 ms
 Execution Time: 186.140 ms
(15 rows)
```
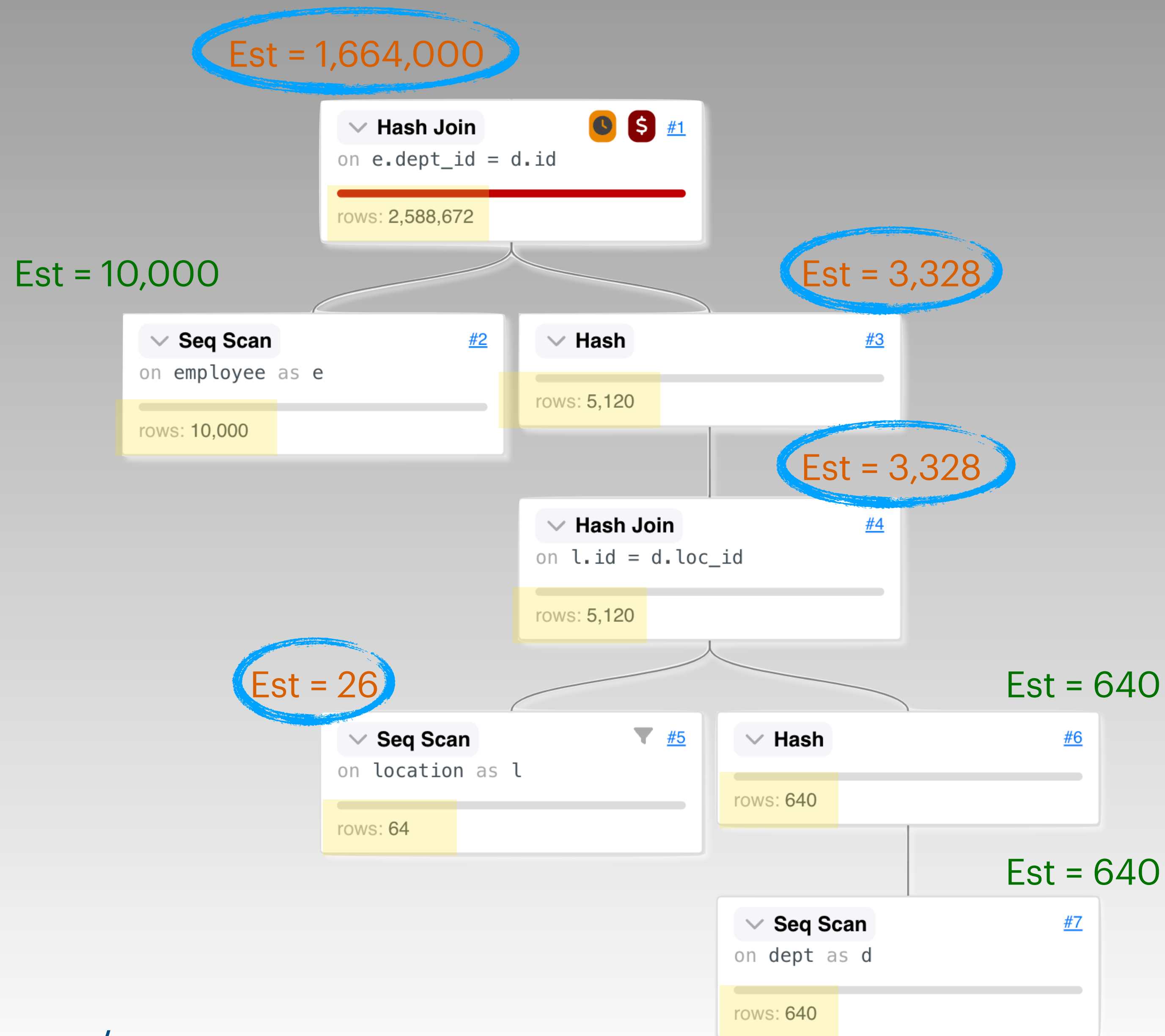
Est = 1,664,000

Hash Join 🕐 $ #1
on e.dept_id = d.id
rows: 2,588,672

Est = 10,000

Est = 3,328

Seq Scan #2
on employee as e
rows: 10,000

Hash #3
rows: 5,120

Est = 3,328

Hash Join #4
on l.id = d.loc_id
rows: 5,120

Est = 26

Est = 640

Seq Scan #5
on location as l
rows: 64

Hash #6
rows: 640

Est = 640

Seq Scan #7
on dept as d
rows: 640

https://explain.dalibo.com/

Amr Elhelw's
TECH
VAULT

Customer

Name	City	Country
...	London	UK
...	Madrid	Spain
...	New York	US
...	Seattle	US
...	Paris	France

Rows in 'customer' table = 1,000

No. of distinct values in 'city' column = 5

No. of distinct values in 'country' column = 4

```
SELECT ... FROM customer WHERE city = 'Seattle' AND country = 'US';
```

Assuming

Independence

Selectivity = 1/5   Selectivity = 1/4

Combined Selectivity = 1/5 * 1/4 = 1/20   Estimated Rows = 1000 * 1/20 = 50

No. of distinct (city, country) combinations = 5   Selectivity = 1/5   Estimated Rows = 1000 * 1/5 = 200

```
EXPLAIN ANALYZE SELECT e.name, d.name FROM employee e JOIN dept d ON e.dept_id = d.id JOIN location l ON d.loc_id = l.id
WHERE country = 'UK' AND city = 'London';

                                                              QUERY PLAN
-----------------------------------------------------------------------------------------------------------------------
 Hash Join  (cost=103.98..19007.99 rows=1664000 width=20) (actual time=2.646..125.921 rows=2588672 loops=1)
   Hash Cond: (e.dept_id = d.id)
   ->  Seq Scan on employee e  (cost=0.00..164.00 rows=10000 width=17) (actual time=0.036..0.529 rows=10000 loops=1)
   ->  Hash  (cost=62.38..62.38 rows=3328 width=11) (actual time=2.595..2.597 rows=5120 loops=1)
         Buckets: 8192 (originally 4096)  Batches: 1 (originally 1)  Memory Usage: 288kB
         ->  Hash Join  (cost=18.40..62.38 rows=3328 width=11) (actual time=0.301..1.436 rows=5120 loops=1)
               Hash Cond: (l.id = d.loc_id)
               ->  Seq Scan on location l  (cost=0.00..4.40 rows=26 width=4) (actual time=0.021..0.055 rows=64 loops=1)
                     Filter: (((country)::text = 'UK'::text) AND ((city)::text = 'London'::text))
                     Rows Removed by Filter: 96
               ->  Hash  (cost=10.40..10.40 rows=640 width=15) (actual time=0.268..0.269 rows=640 loops=1)
                     Buckets: 1024  Batches: 1  Memory Usage: 39kB
                     ->  Seq Scan on dept d  (cost=0.00..10.40 rows=640 width=15) (actual time=0.013..0.133 rows=640 loops=1)
 Planning Time: 0.498 ms
 Execution Time: 186.140 ms
(15 rows)
```

```
CREATE STATISTICS stats_loc ON country, city FROM location;
CREATE STATISTICS

ANALYZE location;
ANALYZE

SELECT m.* FROM pg_statistic_ext JOIN pg_statistic_ext_data ON (oid = stxoid), pg_mcv_list_items(stxdmcv) m WHERE stxname =
'stats_loc';

 index |     values      | nulls | frequency |     base_frequency
-------+-----------------+-------+-----------+----------------------
     0 | {London,UK}     | {f,f} |       0.4 | 0.1600000000000003
     1 | {Madrid,Spain}  | {f,f} |       0.2 | 0.0400000000000001
     2 | {"New York",US} | {f,f} |       0.2 | 0.0400000000000001
     3 | {Paris,France}  | {f,f} |       0.2 | 0.0400000000000001
(4 rows)
```

```
EXPLAIN ANALYZE SELECT e.name, d.name FROM employee e JOIN dept d ON e.dept_id = d.id JOIN location l ON d.loc_id = l.id
WHERE country = 'UK' AND city = 'London';

                                                          QUERY PLAN
--------------------------------------------------------------------------------------------------------------------------------
 Hash Join  (cost=222.64..46496.64 rows=4096000 width=20) (actual time=2.252..137.630 rows=2588672 loops=1)
   Hash Cond: (e.dept_id = d.id)
   ->  Seq Scan on employee e  (cost=0.00..164.00 rows=10000 width=17) (actual time=0.015..0.538 rows=10000 loops=1)
   ->  Hash  (cost=120.24..120.24 rows=8192 width=11) (actual time=2.225..2.226 rows=5120 loops=1)
         Buckets: 8192  Batches: 1  Memory Usage: 288kB
         ->  Hash Join  (cost=18.40..120.24 rows=8192 width=11) (actual time=0.283..1.251 rows=5120 loops=1)
               Hash Cond: (l.id = d.loc_id)
               ->  Seq Scan on location l  (cost=0.00..4.40 rows=64 width=4) (actual time=0.015..0.048 rows=64 loops=1)
                     Filter: (((country)::text = 'UK'::text) AND ((city)::text = 'London'::text))
                     Rows Removed by Filter: 96
               ->  Hash  (cost=10.40..10.40 rows=640 width=15) (actual time=0.257..0.258 rows=640 loops=1)
                     Buckets: 1024  Batches: 1  Memory Usage: 39kB
                     ->  Seq Scan on dept d  (cost=0.00..10.40 rows=640 width=15) (actual time=0.009..0.126 rows=640 loops=1)
 Planning Time: 0.547 ms
 Execution Time: 209.576 ms
(15 rows)
```

```sql
SELECT ... FROM T WHERE col1 + 6 * col2 < 120;
```

```sql
SELECT ... FROM T WHERE value_str LIKE '%st1%';
```

```sql
SELECT ... FROM T WHERE func(col1) = 0;
```

One solution is "Sampling"

Amr Elhelw's
TECH
VAULT