

Dark Corners of PostgreSQL Statistics Analyzer

Oleksandr “Alex” Shulgin
Zalando SE

oleksandr.shulgin@zalando.de

PGDay.ru 2016

Statistics: What? Where? Why?

The Flipping Plan Problem

Deep Dive into Code

A Bird Eye's View

Solution?

Statistics: What? Where? Why?

Dark Corners of PostgreSQL Statistics Analyzer

• <code>pg_statistic</code>	<code>schemaname</code>		<code>name</code>	
	<code>tablename</code>		<code>name</code>	
	<code>attname</code>		<code>name</code>	
	<code>inherited</code>		<code>boolean</code>	
• <code>pg_stats</code>	<code>null_frac</code>		<code>real</code>	
	<code>avg_width</code>		<code>integer</code>	
	<code>n_distinct</code>		<code>real</code>	
• <code>MCV: =, <, ></code>	<code>most_common_vals</code>		<code>anyarray</code>	
	<code>most_common_freqs</code>		<code>real[]</code>	
	<code>histogram_bounds</code>		<code>anyarray</code>	
	<code>correlation</code>		<code>real</code>	
• <code>hist: <, ></code>	<code>most_common_elems</code>		<code>anyarray</code>	
	<code>most_common_elem_freqs</code>		<code>real[]</code>	
	<code>elem_count_histogram</code>		<code>real[]</code>	
	<code>len(mcv) ≤ statistics_target</code>			

Statistics: What? Where? Why?

The Flipping Plan Problem

Dark Corners of PostgreSQL Statistics Analyzer

```
SELECT *  
  FROM print_list  
  JOIN ...  
  JOIN ...  
 WHERE status = 'NOT_YET_PRINTED';
```

-> Index Scan using print_list_status_idx on print_list
(cost=0.27..1138.53 rows=6073 width=56)
(actual time=0.727..0.727 rows=0 loops=1)

-> Seq Scan (OMG!)

Dark Corners of PostgreSQL Statistics Analyzer

```

most_common_vals | {PRINTED,PREPARED,ERROR}      -- 2+ "ERROR"
most_common_freqs | {0.996567,0.002853,6.66667e-05}
histogram_bounds |

most_common_vals | {PRINTED,PREPARED}           -- 0 "ERROR"
most_common_freqs | {0.996543,0.003457}
histogram_bounds |

most_common_vals | {PRINTED}                    -- 1 "ERROR"
most_common_freqs | {0.997102}
histogram_bounds | {PREPARED,ERROR}

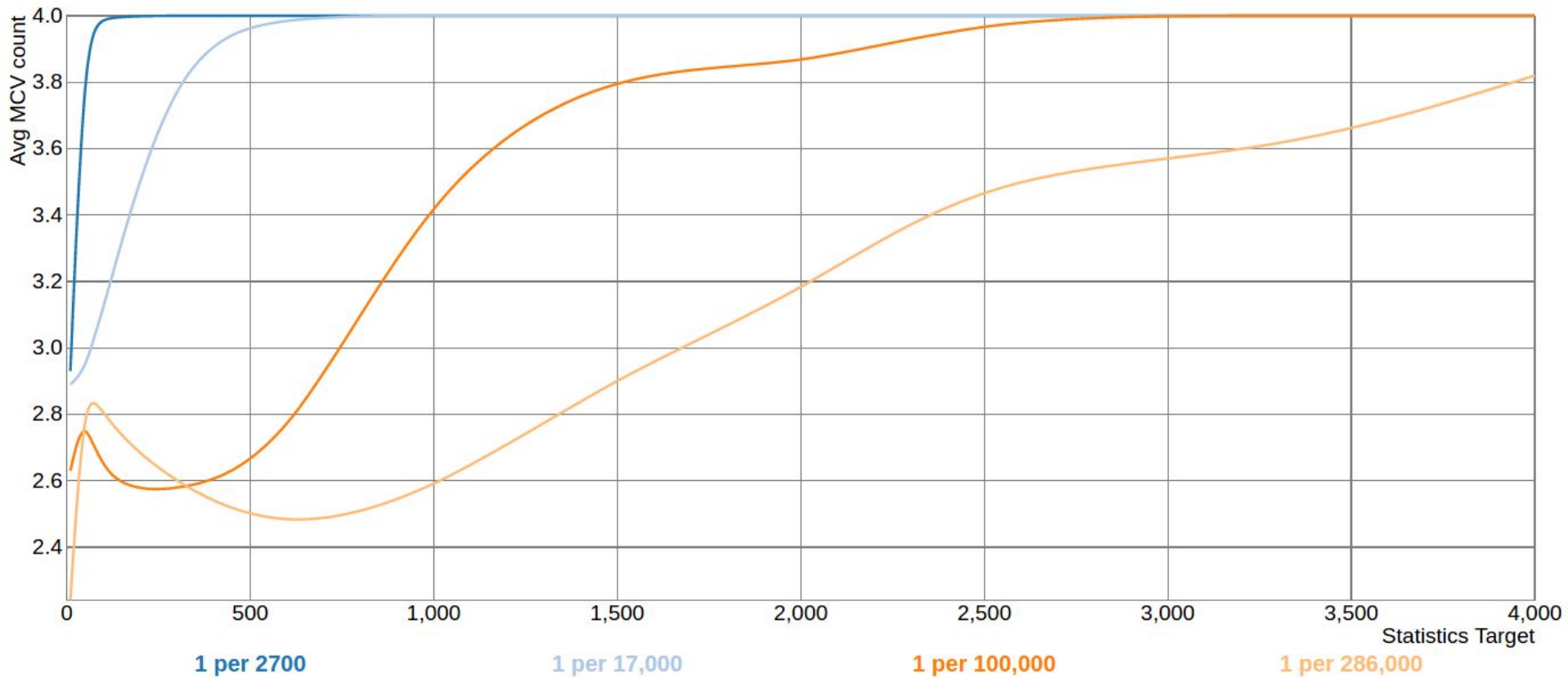
```

```

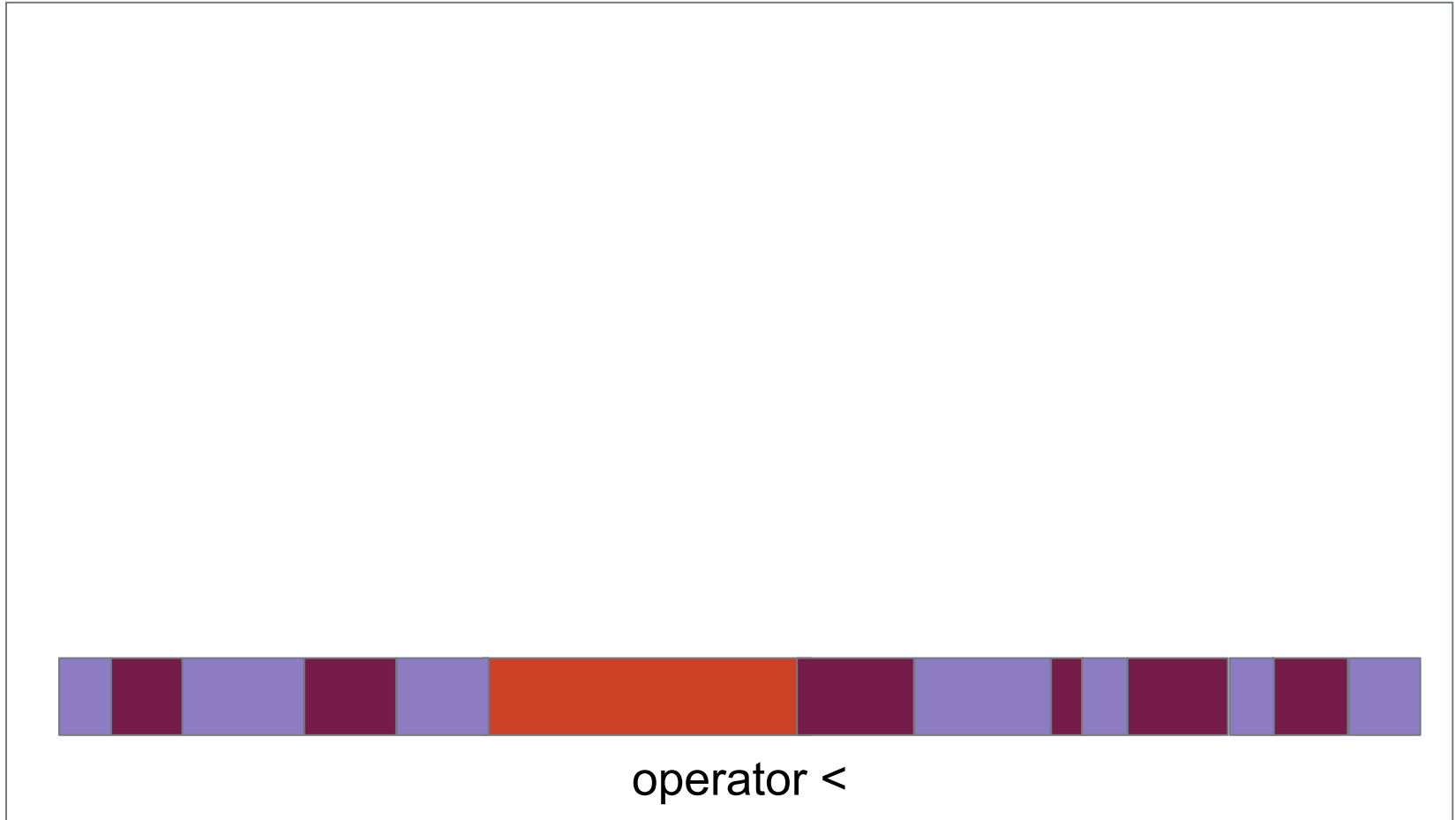
      status      | count | sample
(expectation)
-----+-----+-----
PRINTED           | 3551367 | 29914
PREPARED          |  10162 |    85
ERROR             |    159 |    1 <===== this*
TO_BE_PREPARED   |     2  |     0

```

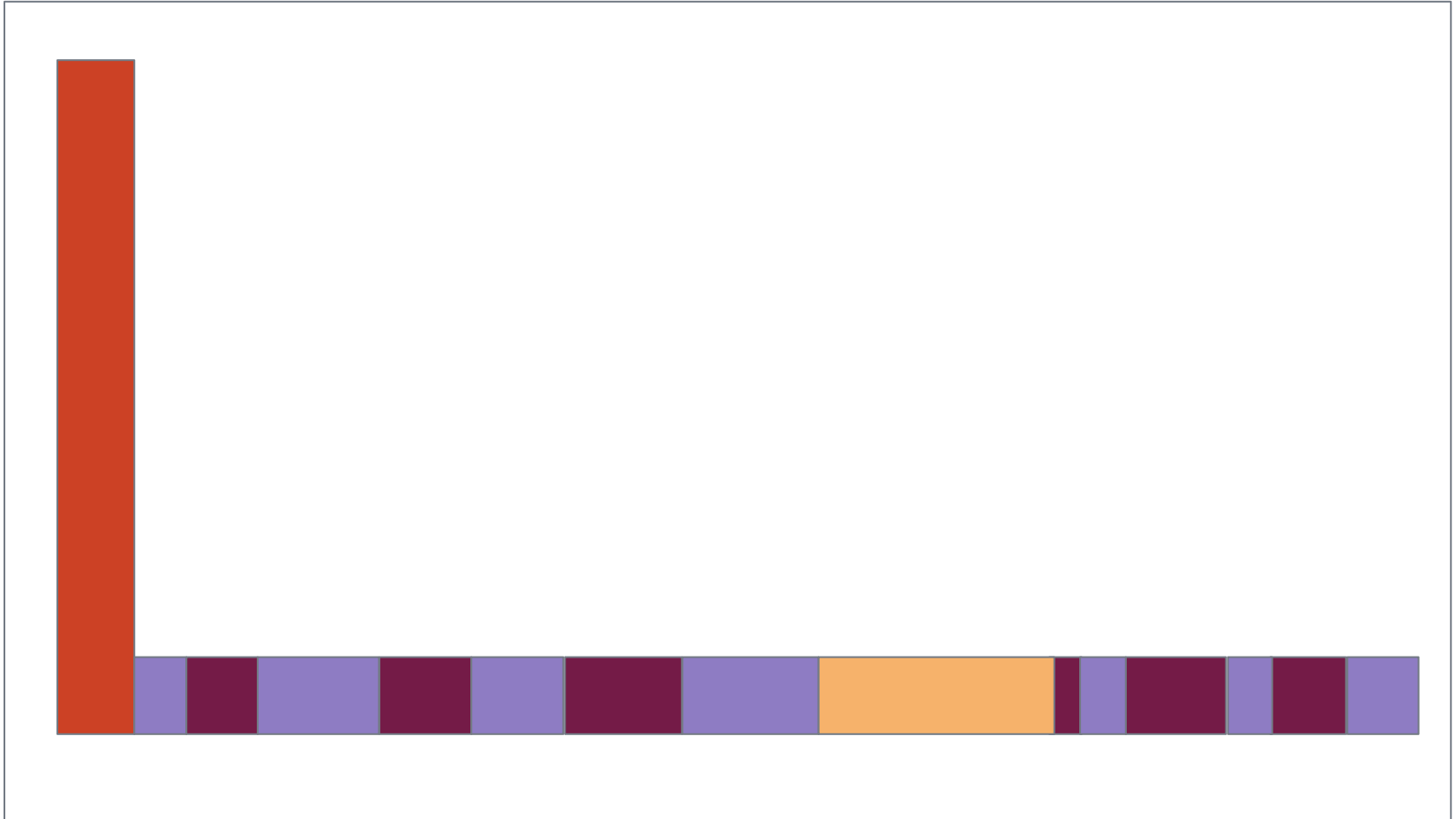
Dark Corners of PostgreSQL Statistics Analyzer



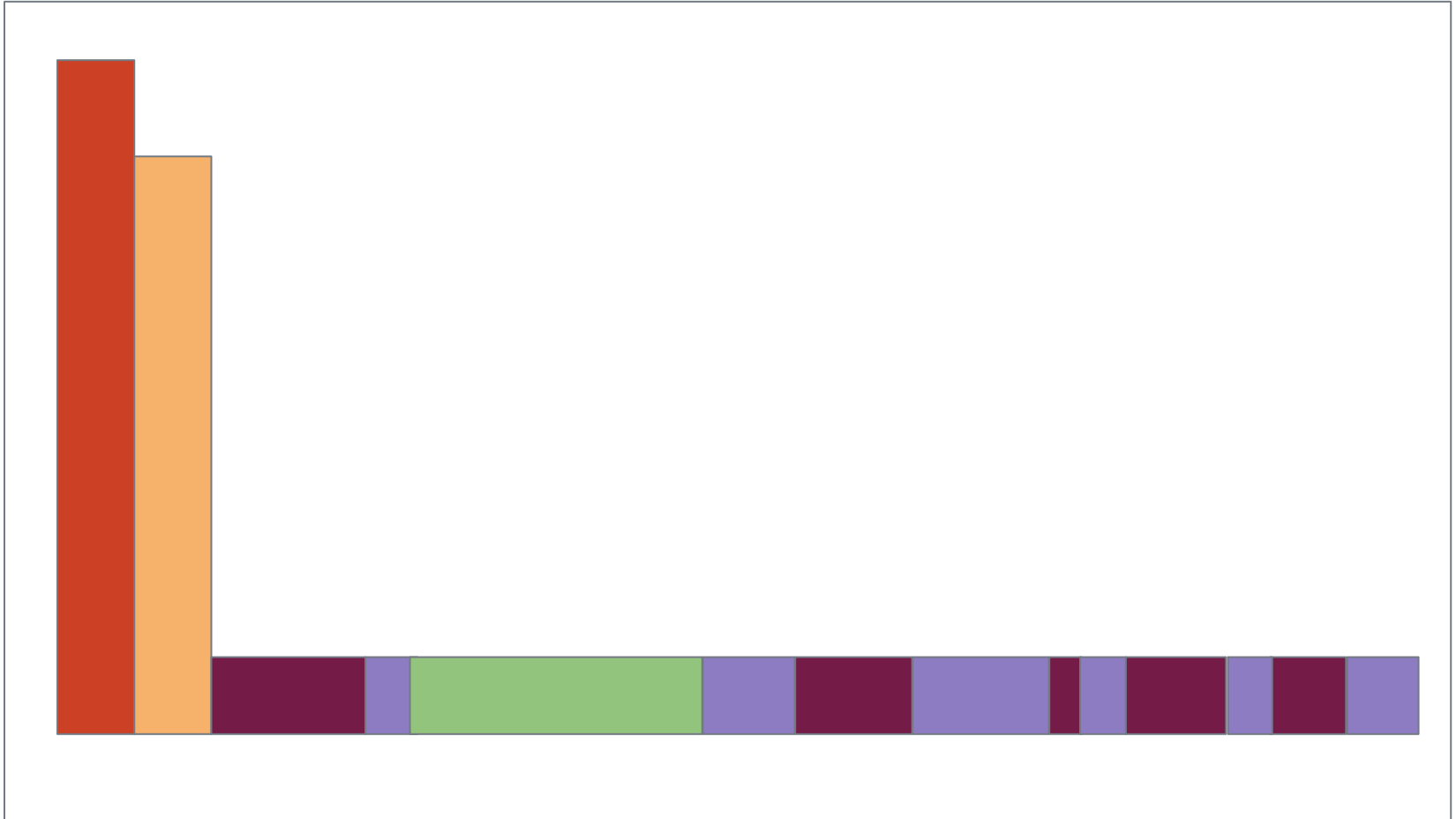
Dark Corners of PostgreSQL Statistics Analyzer



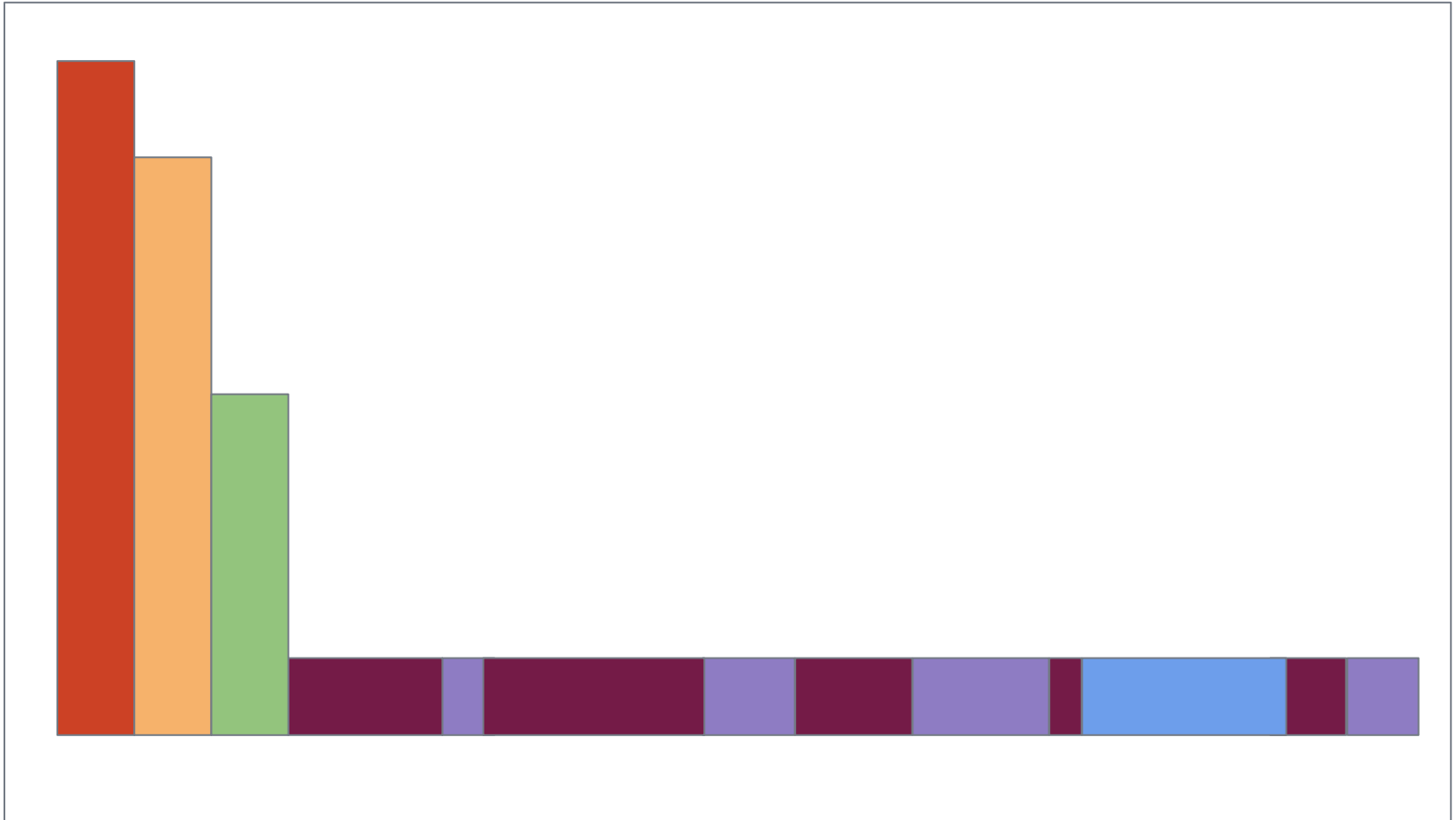
Dark Corners of PostgreSQL Statistics Analyzer



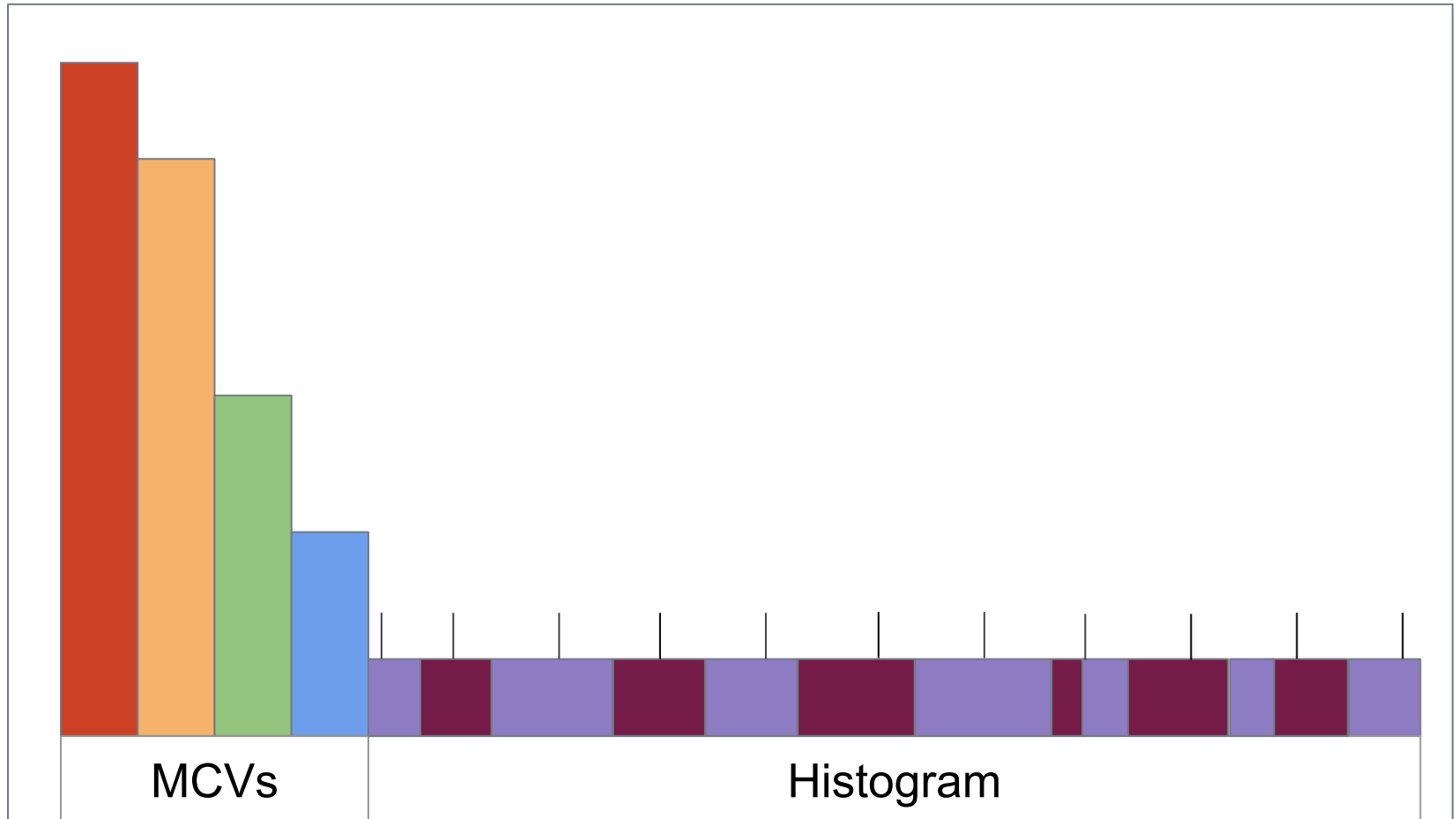
Dark Corners of PostgreSQL Statistics Analyzer



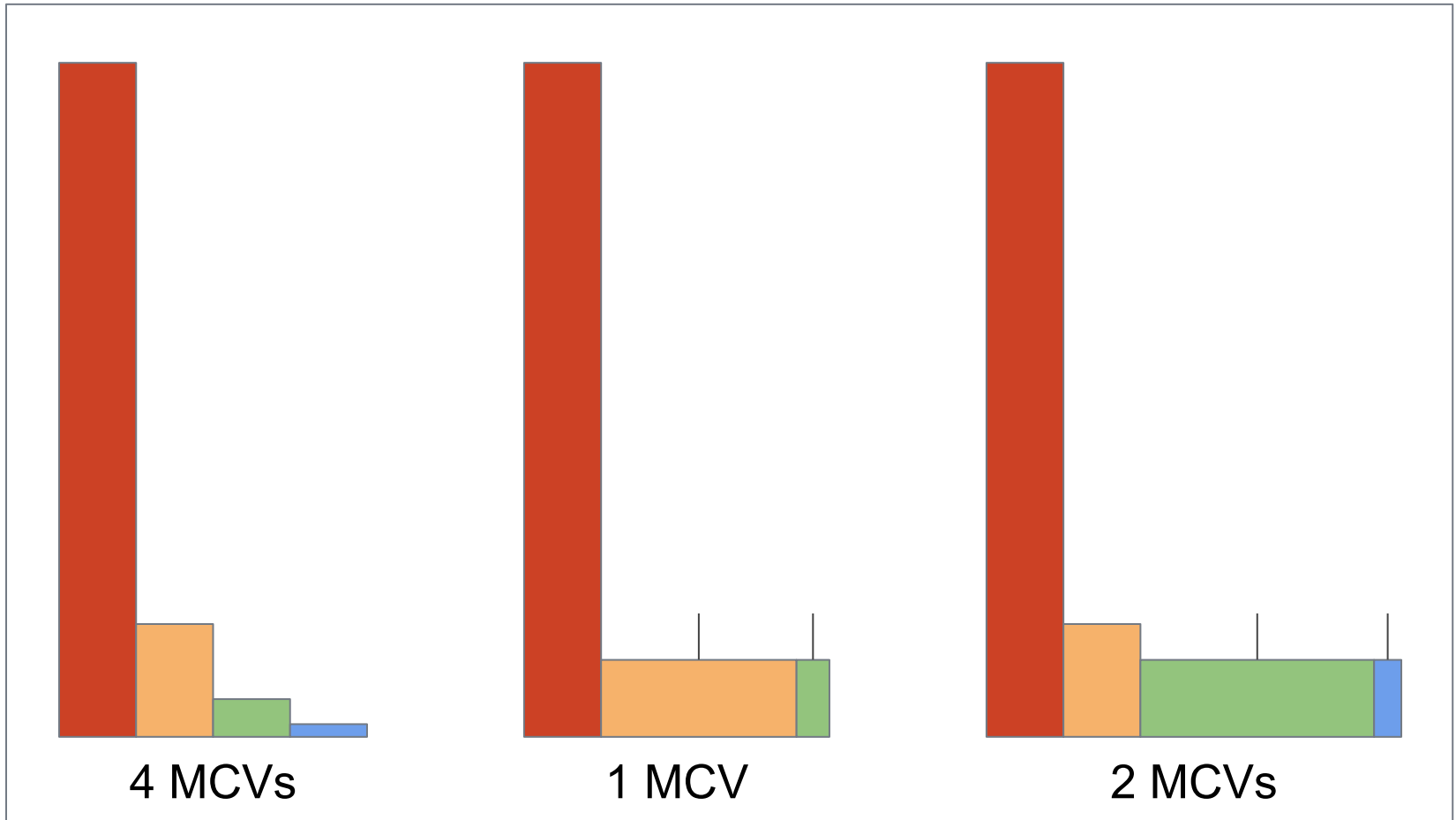
Dark Corners of PostgreSQL Statistics Analyzer



Dark Corners of PostgreSQL Statistics Analyzer



Dark Corners of PostgreSQL Statistics Analyzer



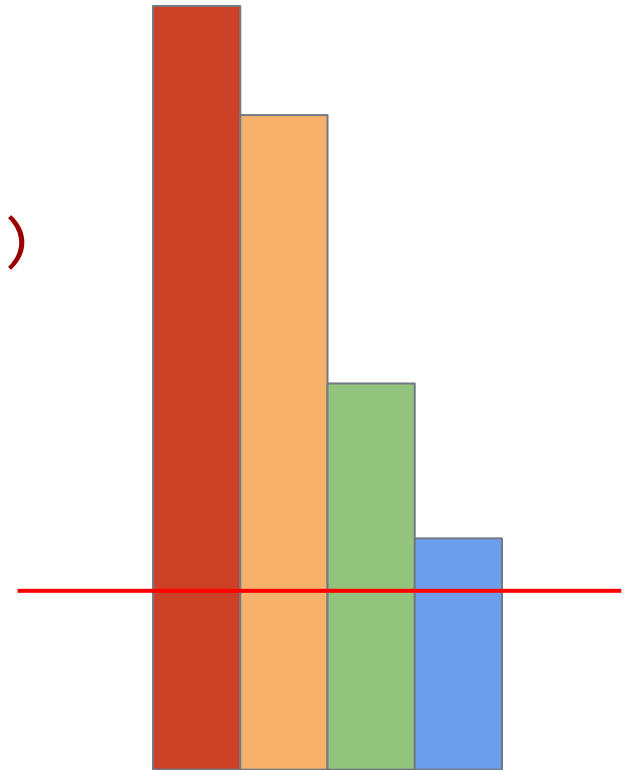
Statistics: What? Where? Why?

The Flipping Plan Problem

Deep Dive into Code

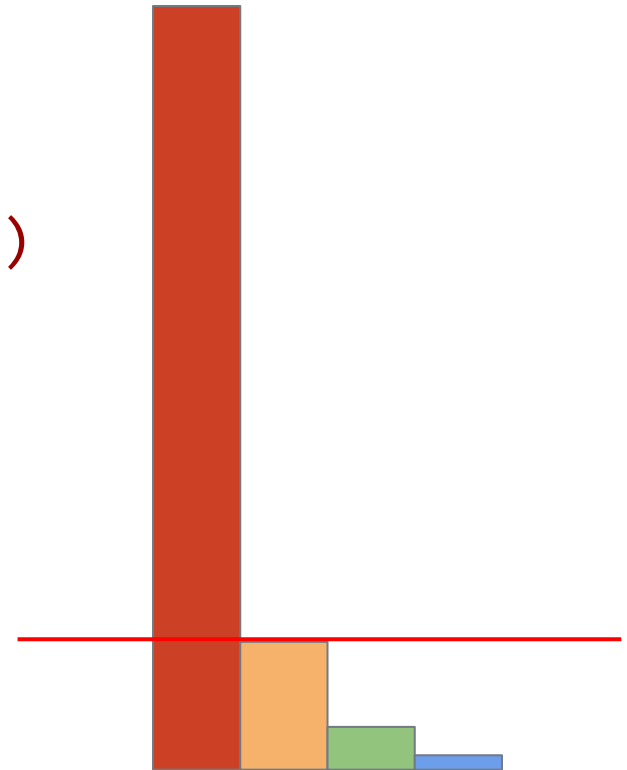
Dark Corners of PostgreSQL Statistics Analyzer

```
mincount = ...  
for (i = 0; i < num_mcv; i++)  
{  
    if (track[i].count < mincount)  
    {  
        num_mcv = i;  
        break;  
    }  
}
```



Dark Corners of PostgreSQL Statistics Analyzer

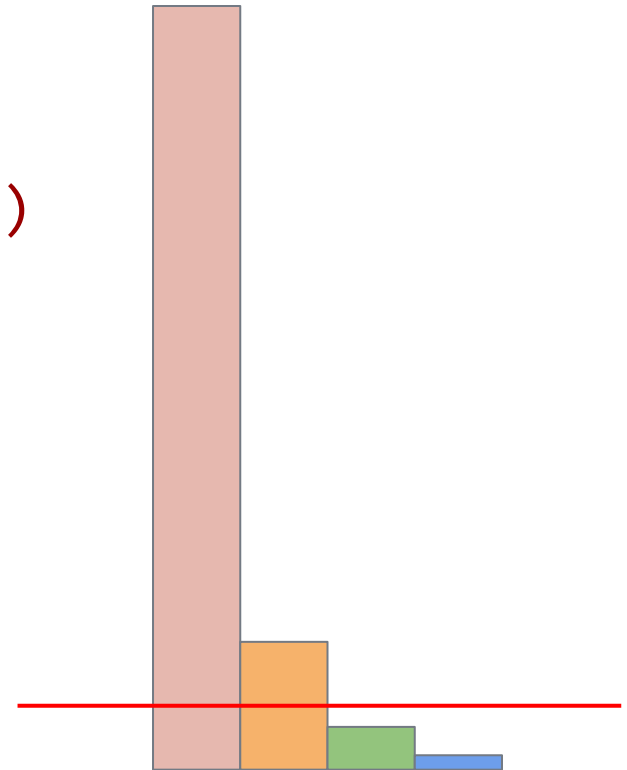
```
mincount = ...  
for (i = 0; i < num_mcv; i++)  
{  
    if (track[i].count < mincount)  
    {  
        num_mcv = i;  
        break;  
    }  
}
```



Dark Corners of PostgreSQL Statistics Analyzer

```
for (i = 0; i < num_mcv; i++)
{
    mincount = ...
    if (track[i].count < mincount)
    {
        num_mcv = i;
        break;
    }
}

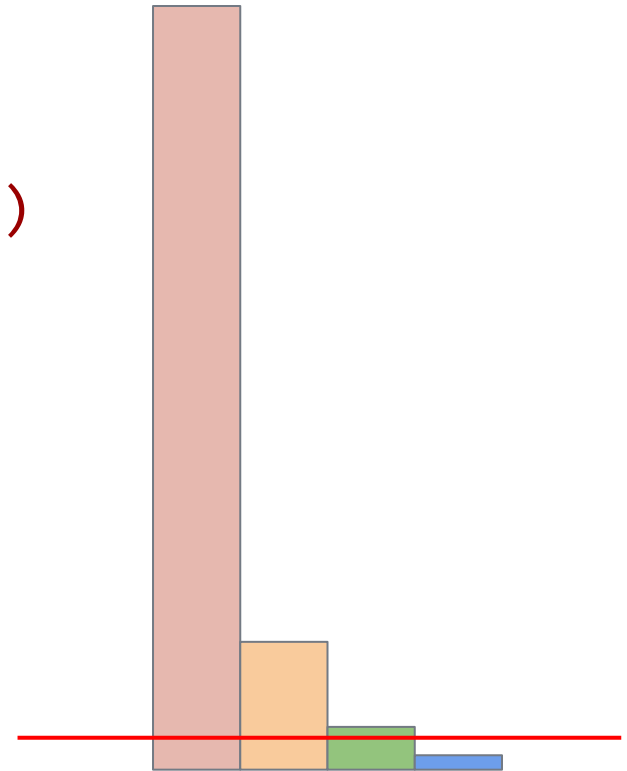
// i = 1
```



Dark Corners of PostgreSQL Statistics Analyzer

```
for (i = 0; i < num_mcv; i++)
{
    mincount = ...
    if (track[i].count < mincount)
    {
        num_mcv = i;
        break;
    }
}

// i = 2
```



Statistics: What? Where? Why?

The Flipping Plan Problem

Deep Dive into Code

A Bird Eye's View

Dark Corners of PostgreSQL Statistics Analyzer

```
WITH stats1 AS (  
    SELECT *,  
        current_setting('default_statistics_target')::int AS stats_target,  
  
        array_length(most_common_vals,1) AS num_mcv,  
        (SELECT SUM(f) FROM UNNEST(most_common_freqs) AS f) AS mcv_frac,  
  
        array_length(histogram_bounds,1) AS num_hist,  
        (SELECT COUNT(DISTINCT h)  
         FROM UNNEST(histogram_bounds::text::text[]) AS h) AS distinct_hist  
  
    FROM pg_stats  
    WHERE schemaname NOT IN ('pg_catalog', 'information_schema')  
),  
stats2 AS (  
    SELECT *,  
        distinct_hist::real/num_hist AS hist_ratio  
    FROM stats1  
)
```

Dark Corners of PostgreSQL Statistics Analyzer

“Before”

```
most_common_vals | {101,100}
most_common_freqs | {0.806367,0.1773}
hist_ratio       | 0.214286
histogram_bounds | {202,202,202,202,202,202,202,202,202,3001,302,302,302,
302,302,302,302,302,302,302,302,302,3031,3185,502,502,502,
502,502}
```

“After”

```
most_common_vals | {101,100,302,202,502,3001,3059,3029,3031,3140,3041,3095,
3100,3102,3192}
most_common_freqs | {0.803933,0.179,0.00656667,0.00526667,0.00356667,
0.000333333,0.000133333,0.0001,0.0001,0.0001,...}
hist_ratio       | 1
histogram_bounds | {3007,3011,3027,3056,3067,3073,3084,3087,3088,3106,3107,
3118,3134,3163,3204,3225,3247}
```

Dark Corners of PostgreSQL Statistics Analyzer

“Before”

```
WITH ...  
SELECT count(1),  
       min(hist_ratio)::real,  
       avg(hist_ratio)::real,  
       max(hist_ratio)::real,  
       stddev(hist_ratio)::real  
FROM stats2  
WHERE histogram_bounds IS NOT NULL;
```

count		21335
min		0.176471
avg		0.939104
max		1
stddev		0.14548

```
WITH ...  
SELECT count(1),  
       min(hist_ratio)::real,  
       avg(hist_ratio)::real,  
       max(hist_ratio)::real,  
       stddev(hist_ratio)::real  
FROM stats2  
WHERE distinct_hist < num_hist;
```

count		4128
min		0.176471
avg		0.686448
max		0.990099
stddev		0.172547

Dark Corners of PostgreSQL Statistics Analyzer

“After”

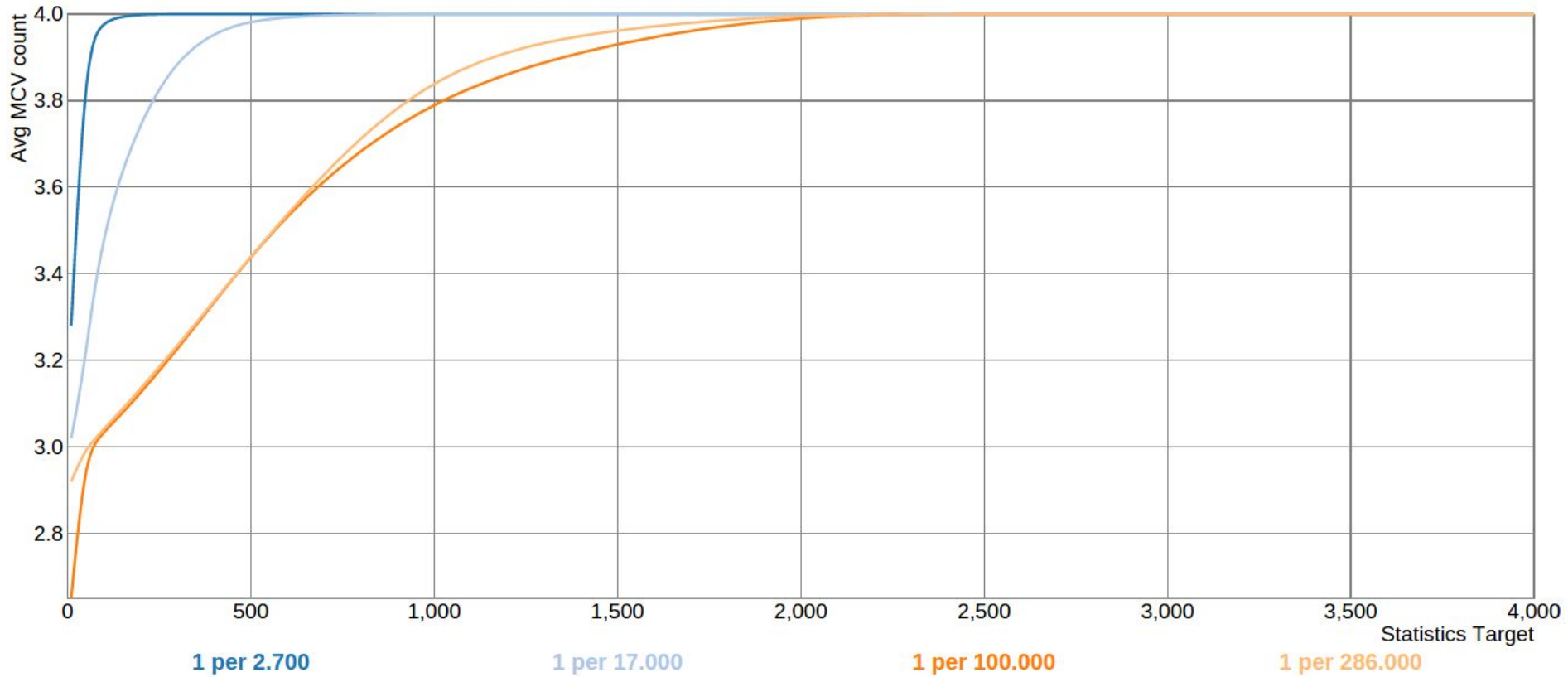
```
WITH ...  
SELECT count(1),  
       min(hist_ratio)::real,  
       avg(hist_ratio)::real,  
       max(hist_ratio)::real,  
       stddev(hist_ratio)::real  
FROM stats2  
WHERE histogram_bounds IS NOT NULL;
```

count		18314
min		0.448276
avg		0.988884
max		1
stddev		0.052899

```
WITH ...  
SELECT count(1),  
       min(hist_ratio)::real,  
       avg(hist_ratio)::real,  
       max(hist_ratio)::real,  
       stddev(hist_ratio)::real  
FROM stats2  
WHERE distinct_hist < num_hist;
```

count		1095	(was 4128)
min		0.448276	(was 0.176471)
avg		0.81408	(was 0.686448)
max		0.990099	
stddev		0.119637	

Dark Corners of PostgreSQL Statistics Analyzer



Dark Corners of PostgreSQL Statistics Analyzer

“Before”

```
WITH ...  
SELECT count(1),  
       min(num_mcv)::real,  
       avg(num_mcv)::real,  
       max(num_mcv)::real,  
       stddev(num_mcv)::real  
FROM stats2  
WHERE num_mcv IS NOT NULL;
```

count		27452
min		1
avg		32.7115
max		100
stddev		40.6927

```
WITH ...  
SELECT count(1),  
       min(num_mcv)::real,  
       avg(num_mcv)::real,  
       max(num_mcv)::real,  
       stddev(num_mcv)::real  
FROM stats2  
WHERE num_mcv < 100;
```

count		20980
min		1
avg		11.9541
max		99
stddev		18.4132

Dark Corners of PostgreSQL Statistics Analyzer

“After”

```
WITH ...  
SELECT count(1),  
       min(num_mcv)::real,  
       avg(num_mcv)::real,  
       max(num_mcv)::real,  
       stddev(num_mcv)::real  
FROM stats2  
WHERE num_mcv IS NOT NULL;
```

count		27527	
min		1	
avg		38.4341	(was 32.7115)
max		100	
stddev		43.3596	

```
WITH ...  
SELECT count(1),  
       min(num_mcv)::real,  
       avg(num_mcv)::real,  
       max(num_mcv)::real,  
       stddev(num_mcv)::real  
FROM stats2  
WHERE num_mcv < 100;
```

count		19329	
min		1	
avg		12.3222	(was 11.9541)
max		99	
stddev		19.6959	

Statistics: What? Where? Why?

The Flipping Plan Problem

Deep Dive into Code

A Bird Eye's View

Solution?

References

- [pgsql-hackers thread \(the patch\)](#)
- [pgsql-performance thread \(flipping plan\)](#)
- [commitfest entry](#)