# Improving Performance with Statistics

Pivotal® Greenplum Database

© 2016 Pivotal Software, Inc. All rights reserved.

# Agenda

- Impact of Statistics on Performance (Why should you ANALYZE)
- Collecting statistics about the database (`EXPLAIN ANALYZE`)
- Improving performance by using `ANALYZE`
- Increasing sampling and statistics
- Identifying when it is best to use `ANALYZE`

© 2016 Pivotal Software, Inc. All rights reserved.

# Impact of Statistics on Performance

- Pivotal Legacy Query Optimizer and Pivotal Query Optimizer generate the best possible plan in the search space of valid execution plans, based on costing.
- The costing of a query plan is dependent on the operators, as well as the cardinality estimation of the input to the execution node/operator.
- Number of distinct values for a given column will determine the estimated skew of that particular column, which will then influence whether we broadcast the tuples or redistribute.. Whether we pick stream aggregation vs. Hash aggregation
- Cardinality estimation is built on top of the statistics on the base table

© 2016 Pivotal Software, Inc. All rights reserved.

# When should you run `ANALYZE`

- Whenever the statistics are stale
  - How do you know? You don't!
  - The view in gp_toolkit is based on time not data distribution
  - So, if the new data you are loading has the same distributional characteristics as the current data – you probably don't need to run `ANALYZE`
  - Otherwise, run `ANALIZE`

© 2016 Pivotal Software, Inc. All rights reserved.

# `ANALYZE` and Database Statistics

Database statistics:

- Are used by the optimizer and query planner
- Should be updated with `ANALYZE`:
  - After loading data (`COPY`, `GPLOAD`, `gpfdist`, ...)
  - After large `INSERT`, `UPDATE`, and `DELETE` operations
  - After `CREATE INDEX` operations
  - After database restores from backups

Pivotal

© 2016 Pivotal Software, Inc. All rights reserved.

# `EXPLAIN ANALYZE` Estimated Costs

`EXPLAIN ANALYZE` provides cost estimates for the execution of the plan node as follows:

- Cost – Measured in units of disk page fetches
- Rows – The number of rows output by the plan node
- Width – Total bytes of all the widest row of the rows output by the plan node

© 2016 Pivotal Software, Inc. All rights reserved.

# Example

```
create table toy (a int, b int);
insert into toy select i%10, i from generate_series(1,100) I;
student=# explain analyze select * from foo where a < 10;
                                                  QUERY PLAN
-------------------------------------------------------------------------------------------------------------
 Gather Motion 3:1  (slice1; segments: 3)  (cost=0.00..1.01 rows=1 width=8)
   Rows out:  3 rows at destination with 1.083 ms to end, start offset by 9.047 ms.
   ->  Seq Scan on foo  (cost=0.00..1.01 rows=1 width=8)
         Filter: a < 10
         Rows out:  3 rows (seg0) with 0.337 ms to first row, 0.345 ms to end, start offset by 9.725 ms.
 Slice statistics:
   (slice0)    Executor memory: 318K bytes.
   (slice1)    Executor memory: 163K bytes avg x 3 workers, 163K bytes max (seg0).
 Statement statistics:
   Memory used: 128000K bytes
 Optimizer status: legacy query optimizer
 Total runtime: 10.233 ms
(12 rows)
```

- Copy the explain analyze into the comment box in http://planchecker.cfapps.io/

© 2016 Pivotal Software, Inc. All rights reserved.

# ANALYZE and VACUUM ANALYZE

## The ANALYZE command:

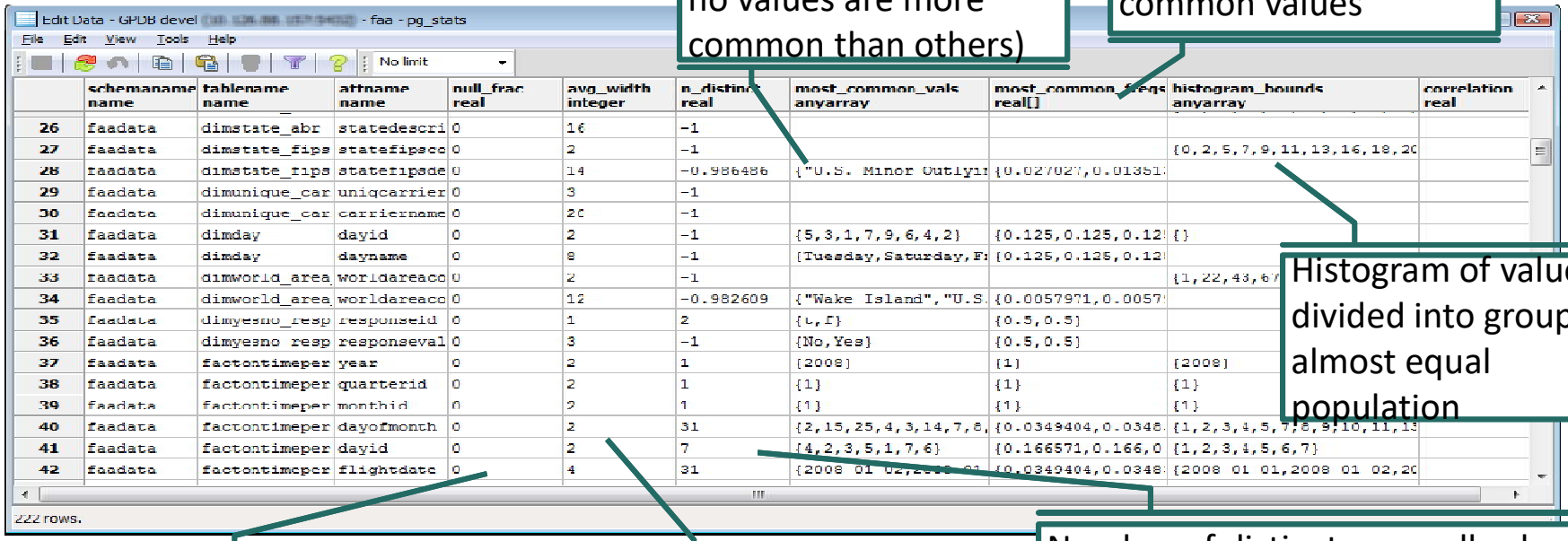
- Is used to generate database statistics
- Can be used on specific table and column names
  `ANALYZE [table [ (column [, ...] ) ]]`

## The VACUUM ANALYZE command:

- Is used to vacuum the database
- Generates database statistics
- Can be used on specific table and column names
  `VACUUM ANALYZE [table [(column [, ...] )]]`

© 2016 Pivotal Software, Inc. All rights reserved.

# Storage of Statistical Data in `pg_stats`




| | schemaname name | tablename name | attname name | null_frac real | avg_width integer | n_distinct real | most_common_vals anyarray | most_common_freqs real[] | histogram_bounds anyarray | correlation real |
|---|---|---|---|---|---|---|---|---|---|---|
| 26 | faadata | dimstate_abr | statedescri | 0 | 16 | -1 | | | | |
| 27 | faadata | dimstate_fips | statefipsco | 0 | 2 | -1 | | | {0,2,5,7,9,11,13,16,18,20 | |
| 28 | faadata | dimstate_fips | statefipsde | 0 | 14 | -0.986486 | {"U.S. Minor Outlyi | {0.027027,0.01351 | | |
| 29 | faadata | dimunique_car | uniqcarrier | 0 | 3 | -1 | | | | |
| 30 | faadata | dimunique_car | carriername | 0 | 20 | -1 | | | | |
| 31 | faadata | dimday | dayid | 0 | 2 | -1 | {5,3,1,7,9,6,4,2} | {0.125,0.125,0.12 | {} | |
| 32 | faadata | dimday | dayname | 0 | 8 | -1 | [Tuesday,Saturday,F | {0.125,0.125,0.12 | | |
| 33 | faadata | dimworld_area | worldareaco | 0 | 2 | -1 | | | {1,22,43,67 | |
| 34 | faadata | dimworld_area | worldareaco | 0 | 12 | -0.982609 | {"Wake Island","U.S. | {0.0057971,0.0057 | | |
| 35 | faadata | dimyesno_resp | responseid | 0 | 1 | 2 | {t,f} | {0.5,0.5} | | |
| 36 | faadata | dimyesno_resp | responseval | 0 | 3 | -1 | {No,Yes} | {0.5,0.5} | | |
| 37 | faadata | factontimeper | year | 0 | 2 | 1 | [2008] | {1} | [2008] | |
| 38 | faadata | factontimeper | quarterid | 0 | 2 | 1 | {1} | {1} | {1} | |
| 39 | faadata | factontimeper | monthid | 0 | 2 | 1 | {1} | {1} | {1} | |
| 40 | faadata | factontimeper | dayofmonth | 0 | 2 | 31 | {2,15,25,4,3,14,7,8, | {0.0349404,0.0348 | {1,2,3,4,5,7,8,9,10,11,13 | |
| 41 | faadata | factontimeper | dayid | 0 | 2 | 7 | {4,2,3,5,1,7,6} | {0.166571,0.166,0 | {1,2,3,4,5,6,7} | |
| 42 | faadata | factontimeper | flightdate | 0 | 4 | 31 | {2008-01-02,2008-01 | {0.0349404,0.0348 | {2008-01-01,2008-01-02,20 | |

Common values (null if no values are more common than others)

Frequency of most common values

Histogram of values divided into groups of almost equal population

Fraction of column entries that are null

Average stored width (bytes) of non-null entries

Number of distinct non-null values (0 – unknown; < 0 – negative of the fraction of the number of rows; > 0 – number of distinct values

© 2016 Pivotal Software, Inc. All rights reserved.

# Use `SET STATISTICS` to Increase Sampling

Sampling for statistics:

- Can be increased for a given column with `ALTER TABLE … SET STATISTICS`
- **Defaults to 25 – NOT RECOMMENDED TO CHANGE THIS**
- May improve query planner estimates for columns used in query predicates and joins (`WHERE` clause)

```
ALTER TABLE customer ALTER customer_id
SET STATISTICS 35;
```

- **Can impact the time it takes to `ANALYZE` if statistics has larger values – do so at your own peril.**

Pivotal

© 2016 Pivotal Software, Inc. All rights reserved.

# The `default_statistics_target` Parameter

The `default_statistics_target` parameter:

- Is used to increase sampling for statistics collected for ALL columns
- Can improve query planner estimates
- Is set to 25 by default
- Can increase the time for `ANALYZE`, but can improve query planner's estimate
- Is overridden by `SET STATISTICS` on a column

Pivotal

© 2016 Pivotal Software, Inc. All rights reserved.

# Effect of Updating the Statistics Value

**pg_stats view of originairportid column in factontimeperformance table**

```
 faadata     | factontimeperformance | originairportid |         0 |         4 |     293.083 | {ATL,ORD
,DFW,DEN,LAX,PHX,IAH,DTW,LAS,SFO,SLC,MCO,MSP,EWR,CLT,BOS,JFK,LGA,SEA,BWI,PHL,SAN,DCA,MDW,MEM} | {0.06
1137,0.0500212,0.0408238,0.0352377,0.031034,0.0276767,0.0276485,0.0242347,0.0229087,0.0208492,0.02079
28,0.0203696,0.0193539,0.0193257,0.0187615,0.0184793,0.0181408,0.0158838,0.0157991,0.0152631,0.014049
9,0.01326,0.0126675,0.0121033,0.0113133} | {ABE,ATL,AUS,BOS,CLE,CVG,DEN,DFW,DTW,EYW,HOU,IAH,JFK,LAX,L
GA,MCO,MKE,OAK,ORD,PDX,PHX,RNO,SEA,SJC,SNA,YUM} |
(1 row)
```

```
faa=# alter table factontimeperformance alter originairportid set statistics 50; analyze factontimepe
rformance;
ALTER TABLE
ANALYZE
```

**Updated statistics on originairportid column**

```
 faadata     | factontimeperformance | originairportid |         0
,DFW,DEN,LAX,PHX,IAH,LAS,DTW,SFO,SLC,MCO,MSP,EWR,CLT,BOS,JFK,SEA,LGA,BWI,PHL,MDW,SAN,DCA,IAD,MEM,TPA,
FLL,CVG,MIA,CLE,STL,BNA,HNL,HOU,RDU,PDX,MCI,SJC,OAK,SMF,DAL,AUS,SNA,SAT,MKE,PIT,IND,MSY,ABQ} | {0.065
0631,0.049255,0.0410246,0.0357173,0.0311764,0.0282248,0.0269618,0.0241095,0.022875,0.0212431,0.020306
5,0.0194125,0.0193841,0.0190578,0.0178516,0.016986,0.0167305,0.0160778,0.0160494,0.0151128,0.014829,0
.0135377,0.012743,0.0117639,0.0117213,0.0113382,0.0112104,0.00987654,0.00973464,0.00939407,0.00918121
,0.00869874,0.00864198,0.00861359,0.00810274,0.00800341,0.00779055,0.00776217,0.00770541,0.00767703,0
.00716617,0.0071236,0.00672627,0.0066695,0.0065276,0.00649922,0.00617284,0.00611608,0.00605932,0.0057
3294} | {ABE,ATL,BNA,BOS,BWI,CIC,CLT,CVG,DCA,DEN,DFW,DTW,EWR,FAR,GNV,HOU,IAD,IAH,ITO,JFK,LAS,LAX,LGA,
MCI,MDT,MEM,MKE,MSP,OAK,ONT,ORD,PDX,PHL,PHX,PSP,RNO,SAN,SEA,SFO,SJC,SLC,SNA,TPA,YUM} |
(1 row)
```

**Sampling size has increased, improving statistics for the column**



© 2016 Pivotal Software, Inc. All rights reserved.

# Updating the Total Number of Entries in a Table

**Examination of the number of rows (tuples) and pages**

```
faa=# SELECT relname, relkind, reltuples, relpages
faa-# FROM pg_class
faa-# WHERE relname LIKE 'factontimeperformance';
         relname          | relkind |  reltuples   | relpages
--------------------------+---------+--------------+----------
 factontimeperformance | r       | 2.09121e+07 |   176772
(1 row)
```

```
faa=# analyze factontimeperformance;
ANALYZE
```

**Examination of the number of rows (tuples) and pages**

```
faa=# SELECT relname, relkind, reltuples, relpages
FROM pg_class
WHERE relname LIKE 'factontimeperformance';
         relname          | relkind | reltuples   | relpages
--------------------------+---------+-------------+----------
 factontimeperformance | r       | 2.1276e+07 |   180611
(1 row)
```

**Number of rows and pages is updated to accurately reflect newly added rows**

© 2016 Pivotal Software, Inc. All rights reserved.

# The `gp_analyze_relative_error` Parameter

The `gp_analyze_relative_error` server configuration parameter:

- Sets the estimated acceptable error in the cardinality of the table; a value of 0.5 is equivalent to an acceptable error of 50%
- Defaults to 0.25
- Allows the number of rows sampled to be increased if the relative error faction is decreased
- Is set in the following manner:
  `gp_analyze_relative_error = .25`

© 2016 Pivotal Software, Inc. All rights reserved.

# ANALYZE Best Practices

Consider the following:

- For large data warehouse environments it may not be feasible to run `ANALYZE` on the entire database or on all columns in a table
- Run `ANALYZE` for
  - Columns used in a `JOIN` condition
  - Columns used in a `WHERE` clause
  - Columns used in a `SORT` clause
  - Columns used in a `GROUP BY` or `HAVING` clause

Pivotal

© 2016 Pivotal Software, Inc. All rights reserved.

# Wrapping Up

In this module we covered:

- Collecting statistics about the database
- Improving performance by using `ANALYZE`
- Increasing sampling and statistics
- Identifying when it is best to use `ANALYZE`

© 2016 Pivotal Software, Inc. All rights reserved.


Pivotal
A NEW PLATFORM FOR A NEW ERA