



# Сбор и использование статистики.



Проверить, идет ли запись

# Меня хорошо видно && слышно?



Ставим "+", если все хорошо  
"-", если есть проблемы



Тема вебинара

# Сбор и использование статистики.



## Ведущий разработчик PostgreSQL/Greenplum в Сбере

Специалист в области разработки и проектирования витрин данных в PostgreSQL/Greenplum, а также в области разработки хранимых процедур в таких СУБД как PostgreSQL/Greenplum, Oracle, MS SQL Server.



# Правила вебинара



Активно  
участвуем



Off-topic обсуждаем  
в учебной группе

<https://t.me/+8Z8WGt1v7tAzYWMY>



Задаем вопрос  
в чат или голосом



Вопросы вижу в чате,  
могу ответить не сразу

# Маршрут вебинара



Рассмотрим сбор статистики

Системные представления статистики

Расширение `pg_stat_statements`

Рефлексия. ДЗ

# Цели вебинара

К концу занятия вы сможете

1. Настраивать сбор статистики
2. Пользоваться системными представлением с статистикой postgres'a
3. Пользоваться расширением pg\_stat\_statements

# Смысл

## Зачем вам это уметь

1. Для построения хороших планов запросов, статистика должна быть актуальна
2. Для анализа проблемных мест (не используется индекс, много seq scan по таблице)
3. Для нахождения медленных запросов

# Немного про процесс сбора статистики

- Сборщик статистики немного увеличивает нагрузку на сервер
- [track\\_activities](#) - включает мониторинг текущих команд, выполняемых любым серверным процессом. По умолчанию on
- [track\\_counts](#) - определяет необходимость сбора статистики по обращениям к таблицам и индексам. По умолчанию on
- [track\\_functions](#) - включает отслеживание использования пользовательских функций. По умолчанию none (отключён)
- [track\\_io\\_timing](#) - включает мониторинг времени чтения и записи блоков. По умолчанию off так как для этого требуется постоянно запрашивать текущее время у операционной системы, что может значительно замедлить работу на некоторых платформах



# Немного про autovacuум

Autovacuум здорового человека



Autovacuум курильщика



А при чём тут **autovacuum**? Мы же про статистику хотели поговорить!

Analyze и analyse – одно и то же

<https://postgrespro.ru/docs/postgrespro/14/sql-analyze>

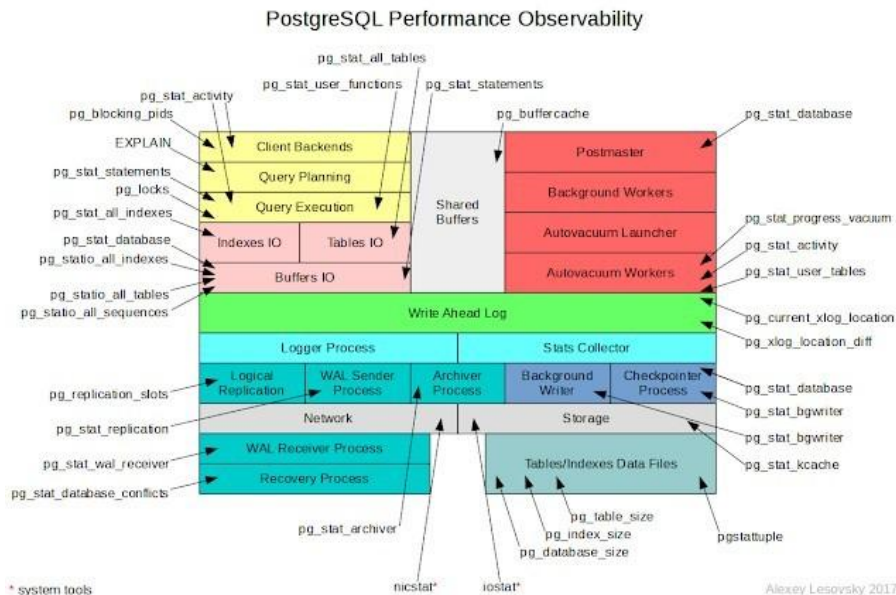
# Default\_statistics\_target

```
Default_statistics_target = N (default value 100)  
Row_analyze = N * 300  
ALTER TABLE.. ALTER COLUMN.. SET STATISTICS 0..10000
```



# Системные представления Postgres

- pg\_stat\_database
- pg\_stats
- pg\_statistic\_ext
- pg\_stat\_activity
- pg\_stat\_user\_tables
- pg\_stat\_user\_indexes



Полный список представлений в документации:

<https://postgrespro.ru/docs/postgresql/14/monitoring-stats> (таблица 28.2)

# pg\_stat\_database

Основные колонки:

- blks\_hit – количество блоков, полученных из кэша PostgreSQL
- blks\_read – количество блоков, прочитанных с диска
- xact\_commit – количество закомиченных транзакций
- xact\_rollback – количество транзакций, где был выполнен откат транзакции

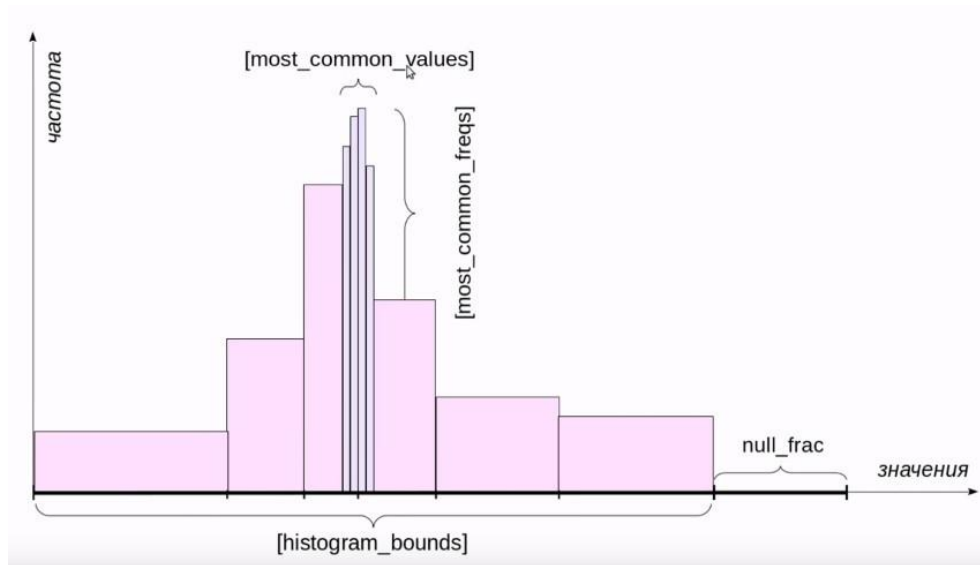
Отсюда мы можем получить следующую информацию:

- Как много информации мы получаем из кэша
- Как часто у нас бывают проблемы с транзакциями

# pg\_stats

Основные колонки:

- null\_frac
- n\_distinct
- most\_common\_vals
- most\_common\_freqs
- histogram\_bounds
- correlation



# pg\_statistic\_ext

```
Create statistics stat_name(dependencies|ndistinct)on field_name1, field_name2 from table_name;
```

- stxdistinct - (Уникальные фамилия, имя, отчество)
- stxdependencies - (Регион – Город)
- mcv

<https://postgrespro.ru/docs/postgresql/14/sql-createstatistics>

# pg\_stat\_activity

- pid
- backend\_start
- wait\_event\_type
- wait\_event
- state
- query
- backend\_type



\* Длина столбца query track\_activity\_query\_size (значение в байтах)

<https://postgrespro.ru/docs/postgresql/14/monitoring-stats#WAIT-EVENT-TABLE>

# pg\_stat\_user\_tables

- relname
- seq\_scan
- seq\_tup\_read
- n\_tup\_upd
- n\_tup\_hot\_upd
- n\_live\_tup
- n\_dead\_tup

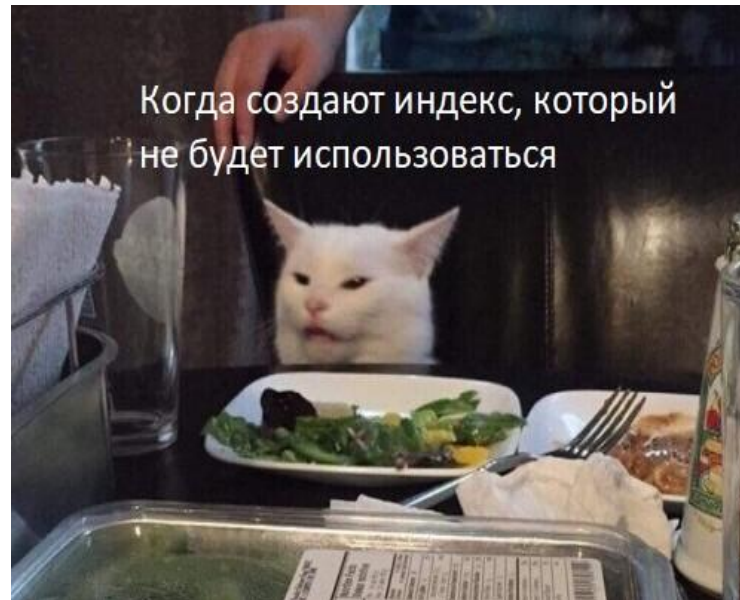






# pg\_stat\_user\_indexes

- relname
- indexrelname
- idx\_scan
- idx\_tup\_read
- idx\_tup\_fetch



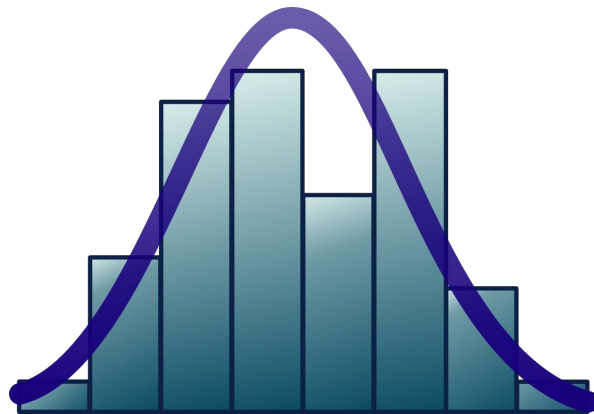
# Минитест

<https://forms.gle/vo7fEsM8qEJfVZQq7>

3-5 минут

# pg\_stat\_statements

- rows — суммарное количество возвращенных строк;
- shared\_blks\_hit — количество страниц, которые были в кэше БД;
- shared\_blks\_read — количество страниц, которые были прочитаны с диска, чтобы выполнить запросы такого типа;
- shared\_blks\_dirtied — количество страниц, которые были изменены;
- shared\_blks\_written — количество страниц, которые были записаны на диск;



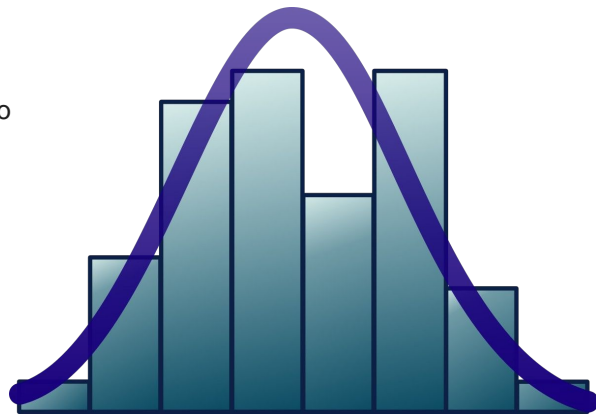
Немного про снижение производительности:

<https://www.depesz.com/2014/10/14/what-logging-has-least-overhead/>

<https://postgrespro.ru/docs/postgresql/14/monitoring-stats>

# pg\_stat\_statements

- `local_blks_hit`, `local_blks_read`, `local_blks_dirtied`, `local_blks_written` — то же самое, что предыдущие 4, только для временных таблиц и индексов;
- `temp_blks_read` — сколько страниц временных данных было прочитано;
- `temp_blks_written` — сколько страниц временных данных было записано (используется при сортировке на диске, джойнах и других временных операциях);
- `blk_read_time` — сколько времени суммарно заняло чтение с диска;
- `blk_write_time` — сколько времени суммарно заняла запись на диск.



# Рефлексия

# Результаты опроса. Рефлексия

[https://docs.google.com/forms/d/1adp4SOZEyYZTxDzdy6qXM\\_alhknRghAGL\\_1wthsFerU/edit#responses](https://docs.google.com/forms/d/1adp4SOZEyYZTxDzdy6qXM_alhknRghAGL_1wthsFerU/edit#responses)

1. В каком представлении мы можем посмотреть статистику по таблицам?
2. В каком представлении мы можем посмотреть статистику по индексам?
3. Что мы делаем с индексом, который был создан давно, но у него `idx_scan = 0`?

**Заполните, пожалуйста,  
опрос о занятии  
по ссылке <https://otus.ru/polls/79288/>**



# ДЗ

# Домашнее задание

В результате выполнения ДЗ вы научитесь пользоваться различными вариантами соединения таблиц.

В данном задании тренируются навыки:

- написания запросов с различными типами соединений

Необходимо:

- 1) Реализовать прямое соединение двух или более таблиц
- 2) Реализовать левостороннее (или правостороннее) соединение двух или более таблиц
- 3) Реализовать кросс соединение двух или более таблиц
- 4) Реализовать полное соединение двух или более таблиц
- 5) Реализовать запрос, в котором будут использованы разные типы соединений
- 6) Сделать комментарии на каждый запрос
- 7) К работе приложить структуру таблиц, для которых выполнялись соединения

Задание со звездочкой\*

Придумайте 3 своих метрики на основе показанных представлений, отправьте их через ЛК, а так же поделитесь с коллегами в слаке

Спасибо за внимание!

# Приходите на следующие вебинары



## **Ведущий разработчик PostgreSQL/Greenplum в Сбере**

Специалист в области разработки и проектировании витрин данных в PostgreSQL/Greenplum, а также в области разработки хранимых процедур в таких СУБД как PostgreSQL/Greenplum, Oracle, MS SQL Server.