




OTUS

Онлайн-образование

The background of the slide is a blue-tinted aerial photograph of a dense city skyline, likely New York City. Overlaid on this is a semi-transparent network diagram consisting of white dots connected by thin white lines, creating a web-like pattern across the middle section of the slide.

Меня хорошо видно && слышно?

Ставьте  , если все хорошо
Напишите в чат, если есть проблемы
заодно проверяем, включена ли запись занятия

Ваш тренер:
Ржевский Михаил



Физический уровень PostgreSQL

Правила вебинара



Активно участвуем



Задаем вопрос в чат



Вопросы вижу в чате, могу ответить не сразу

Маршрут вебинара

**1. Работа с
PostgreSQL**

**3. Физическая
структура данных**

**2. Серверные
процессы и
память**

4. Практика

Цели вебинара

После занятия вы сможете

1

иметь представление об устройстве

PostgreSQL

2

знать процессы PostgreSQL и структуру

памяти

3

знать как PostgreSQL работает с

данными на физическом уровне

Смысл

| Зачем вам это уметь, в результате:

1

чтобы не удалить что-нибудь нужное в файловой системе (кто помнит про xlog?:)

2

чтобы не убить случайно нужный процесс и убить ненужный



работа с PostgreSQL

работа с PostgreSQL

Клиент PostgreSQL

- работает через **TCP** и **Unix socket**
- самописный софт как правило использует библиотеку jdbc
- протокол** взаимодействия **открыт**
- для каждого клиентского соединения порождается выделенный серверный процесс
- при этом клиентский процесс называют
- а серверный - **backend**
- мы будем работать через
- есть и GUI, например pgAdmin3/4, dBeaver

работа с PostgreSQL

Запуск

\$ psql -d база -U роль -h узел -p порт

Новое подключение в psql

=> \c[onnect] база роль узел порт

Информация о текущем подключении

=> \conninfo

работа с PostgreSQL

Как посмотреть конфигурационные файлы?

Зависит ли путь от ОС?

работа с PostgreSQL

Как посмотреть конфигурационные файлы?

```
# show hba_file;
```

```
# show config_file;
```

все параметры:

```
# show all;
```

```
# select name, setting, context, short_desc from pg_settings;
```

<https://postgrespro.ru/docs/postgrespro/12/view-pg-settings>

работа с PostgreSQL

По умолчанию postgres слушает только localhost

Что нужно сделать для подключения?

работа с PostgreSQL

1. включаем listener в postgresql.conf

`listen_addresses = '*'` *# IP адреса, на которых принимает постгрес подключения, например localhost, 10.*.*.*;*

`netstat -a | grep post`

show listen_addresses; -- второй вариант

alter system set listen_addresses = '';*

включаем вход по паролю в pg_hba.conf и меняем маску подсети

2. `host all all 0.0.0.0/0 md5`

в 14 версии md5 -> scram-sha-256

3. Если используется Google VPC или Yandex, то надо добавить порт

работа с PostgreSQL

4. задаем пароль юзеру postgres

```
ALTER USER postgres PASSWORD '123';
```

5. Перегружаем сервер

```
pg_ctlcluster 12 main restart
```

Вуаля

```
psql -h 192.168.0.20 -U postgres
```


работа с PostgreSQL

при настройке pg_hba.conf обратите внимание на метод аутентификации
обязательно md5 (13), scram-sha-256(14)

password - пароль не шифруется при передаче по сети

<https://postgrespro.ru/docs/postgrespro/13/auth-pg-hba-conf>

работа с PostgreSQL

<code>\?</code>	список команд psql
<code>\? variables</code>	переменные psql
<code>\h[elp]</code>	список команд SQL
<code>\h команда</code>	синтаксис команды SQL
<code>\q</code>	выход(до 11 версии)

<https://postgrespro.ru/docs/postgrespro/13/app-psql>

работа с PostgreSQL

- \l – список баз данных
- \du – список пользователей
- \dp (или \z) – список таблиц, представлений, последовательностей, прав доступа к ним
- \di – индексы
- \dt – список таблиц
- \dt+ — список всех таблиц с описанием
- \dt *s* — список всех таблиц, содержащих s в имени
- \dS – системные таблицы
- \d+ – описание таблицы
- \d “table_name” – описание таблицы
- \i запуск команды из внешнего файла, например \i /my/directory/my.sql
- \pset – команда настройки параметров форматирования
- \echo – выводит сообщение
- \set – устанавливает значение переменной среды. Без параметров

выводит список текущих переменных (\unset – удаляет).

работа с PostgreSQL

При запуске **psql** выполняются **команды**, записанные в двух файлах — общесистемном и пользовательском. **Общий системный файл** называется `psqlrc` и располагается в каталоге `/usr/local/pgsql/etc` при обычной сборке из исходных кодов.

Расположение этого каталога можно узнать командой **`pg_config --sysconfdir`**

Пользовательский файл находится в домашнем каталоге пользователя ОС и называется **`.psqlrc`**. Его расположение можно изменить, задав переменную окружения **`PSQLRC`**.

В эти файлы можно записать команды, настраивающие `psql` — например, изменить приглашение, включить вывод времени выполнения команд и т. п.

История команд сохраняется в файле **`.psql_history`** в домашнем каталоге пользователя. Расположение этого файла можно изменить, задав переменную окружения `PSQL_HISTORY` или переменную `psql HISTFILE`. По умолчанию хранится 500 последних команд; это число можно изменить переменной `psql HISTSIZE`.

работа с PostgreSQL

```
nano ~/.psqlrc
```

```
\set PROMPT1 '%M:%> %n@%/%R%#%x '
```

- %M задает имя хоста сервера БД (при соединении через сокет домена Unix это local);
- %> задает прослушивающий порт;
- %n указывает имя пользователя сессии;
- %/ относится к текущей базе данных;
- %R задает режимы (однотрочный, или single-line — ^; отключен, или disconnected — !, как правило используется символ =);
- %# задает пользователя (суперпользователь — #, обычный пользователь — >);
- %x относится к статусу транзакции — (как правило, не заполняется).

[Как настроить приглашение PostgreSQL с помощью psqlrc в Ubuntu 14.04](#)

работа с PostgreSQL

Расширенный вывод информации - вертикальный вывод колонок

```
select * from pg_stat_activity;
```

```
\x
```

Устанавливает или переключает режим развёрнутого вывода таблицы

```
select * from pg_stat_activity;
```

```
\gx
```

Метакоманда `\gx` подобна `\g`, но принудительно включает расширенный режим вывода для текущего запроса

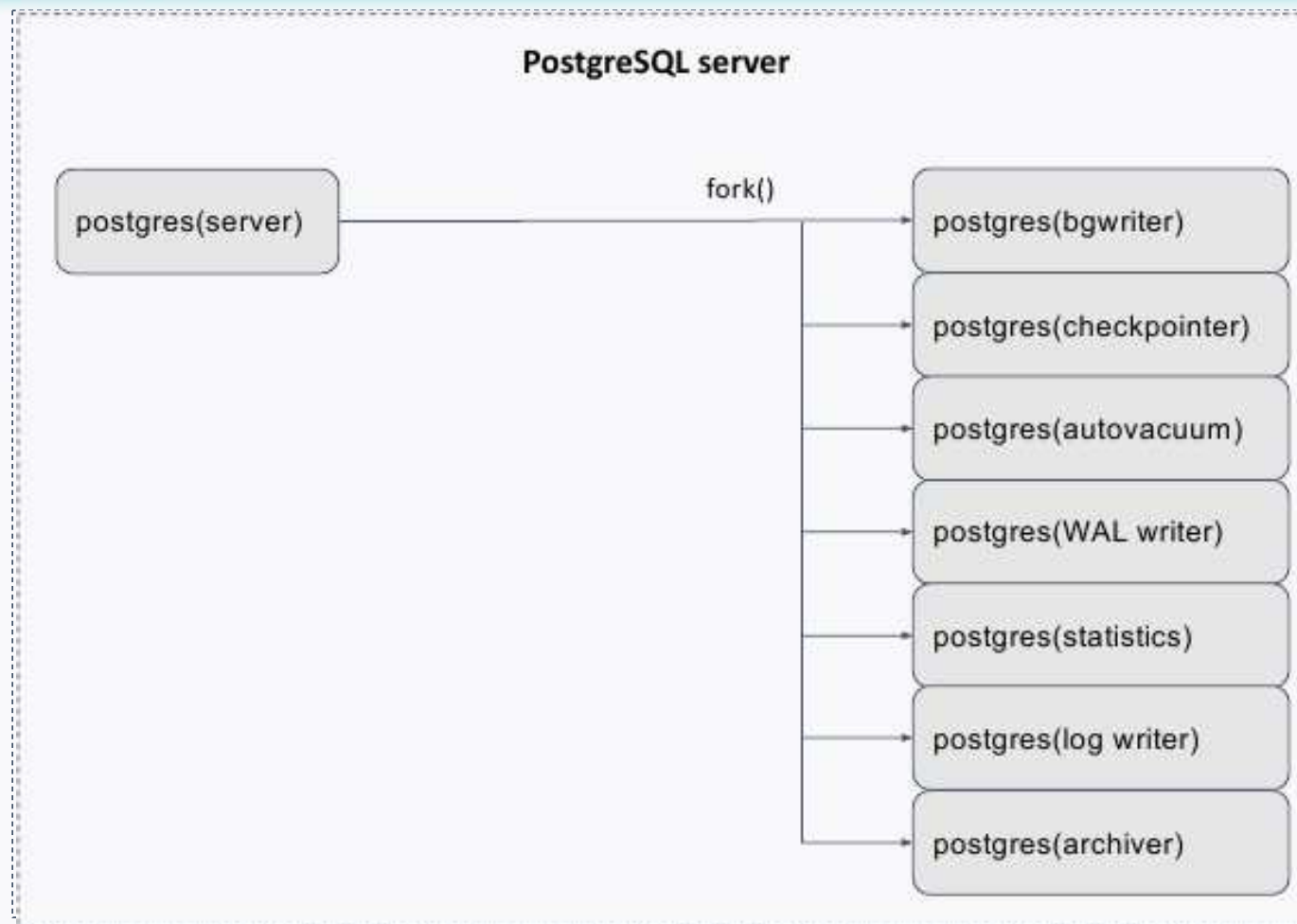
The image features a high-angle, aerial view of a city skyline, likely New York City, with numerous skyscrapers and buildings. The entire image is overlaid with a semi-transparent blue layer. A network of white lines and dots, resembling a molecular or digital structure, is superimposed on the blue background. The word "Вопросы?" is written in white, bold, sans-serif font in the center of the image.

Вопросы?

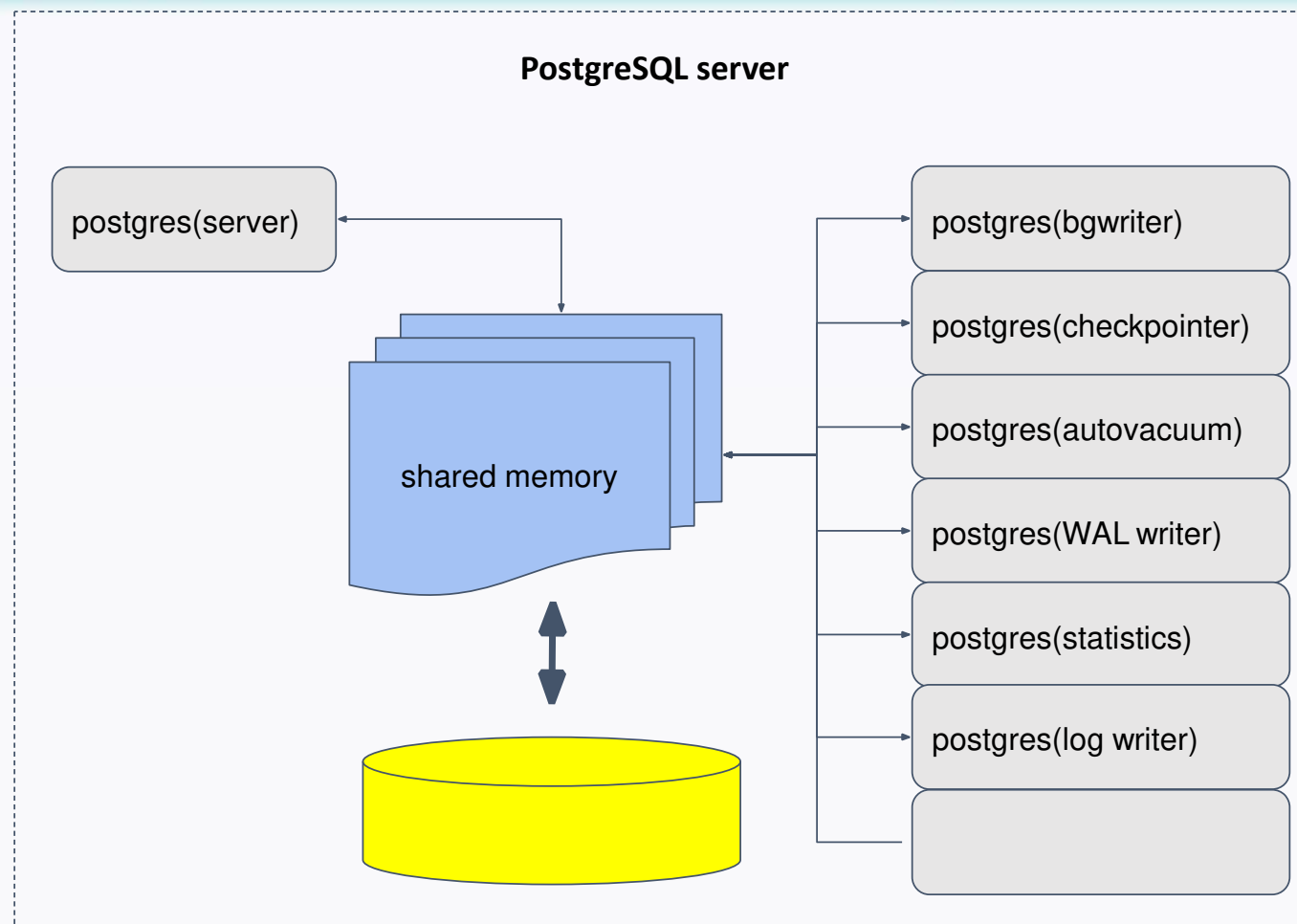
The background of the slide is a high-angle, aerial photograph of a city skyline, likely New York City, featuring numerous skyscrapers. The entire image is filtered with a monochromatic blue color. Overlaid on the cityscape is a faint, white geometric network pattern consisting of interconnected dots and lines, resembling a data or communication network. This pattern is most prominent in the central area where the title is located.

Серверные процессы и память

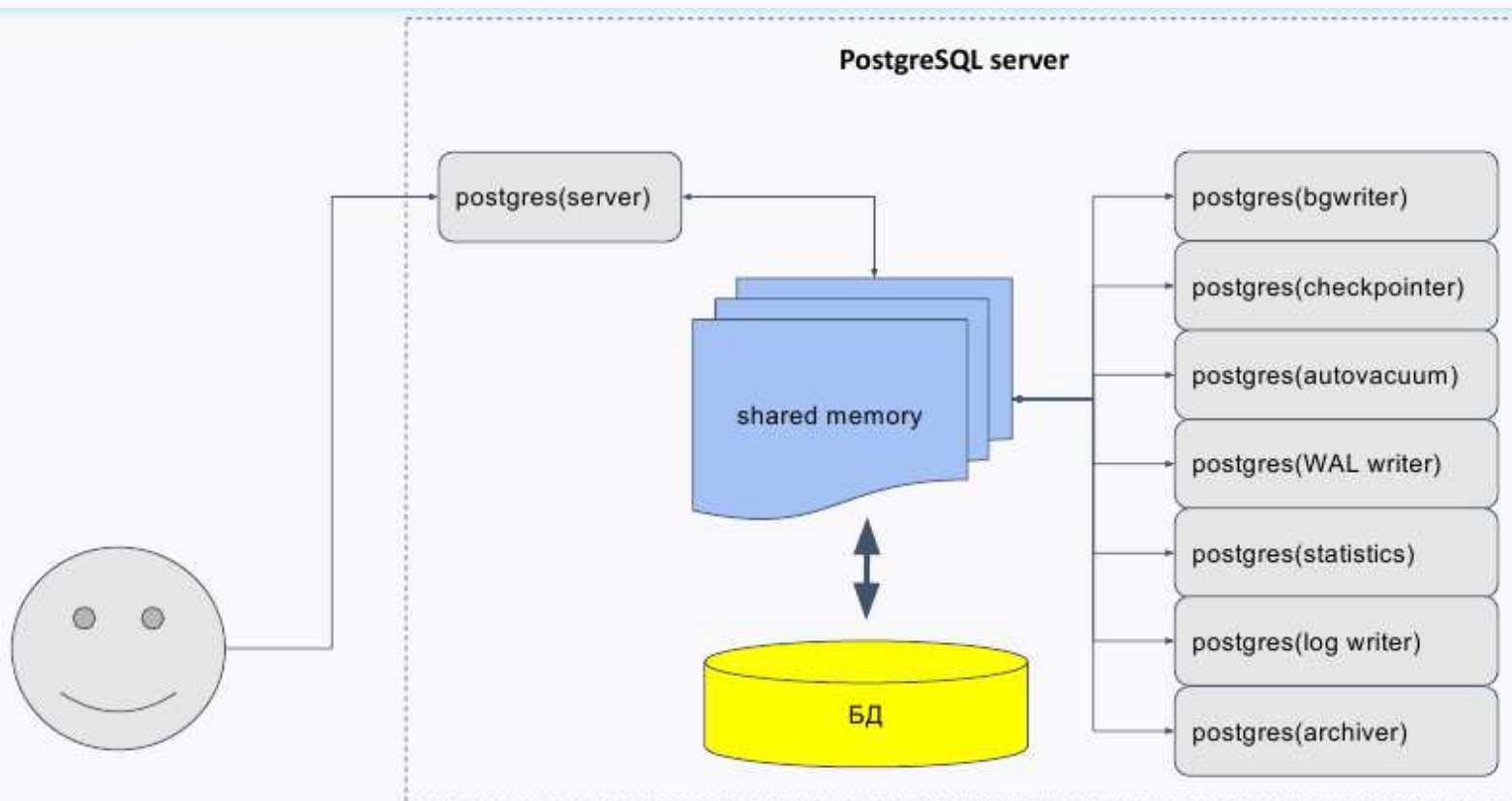
Серверные процессы и память



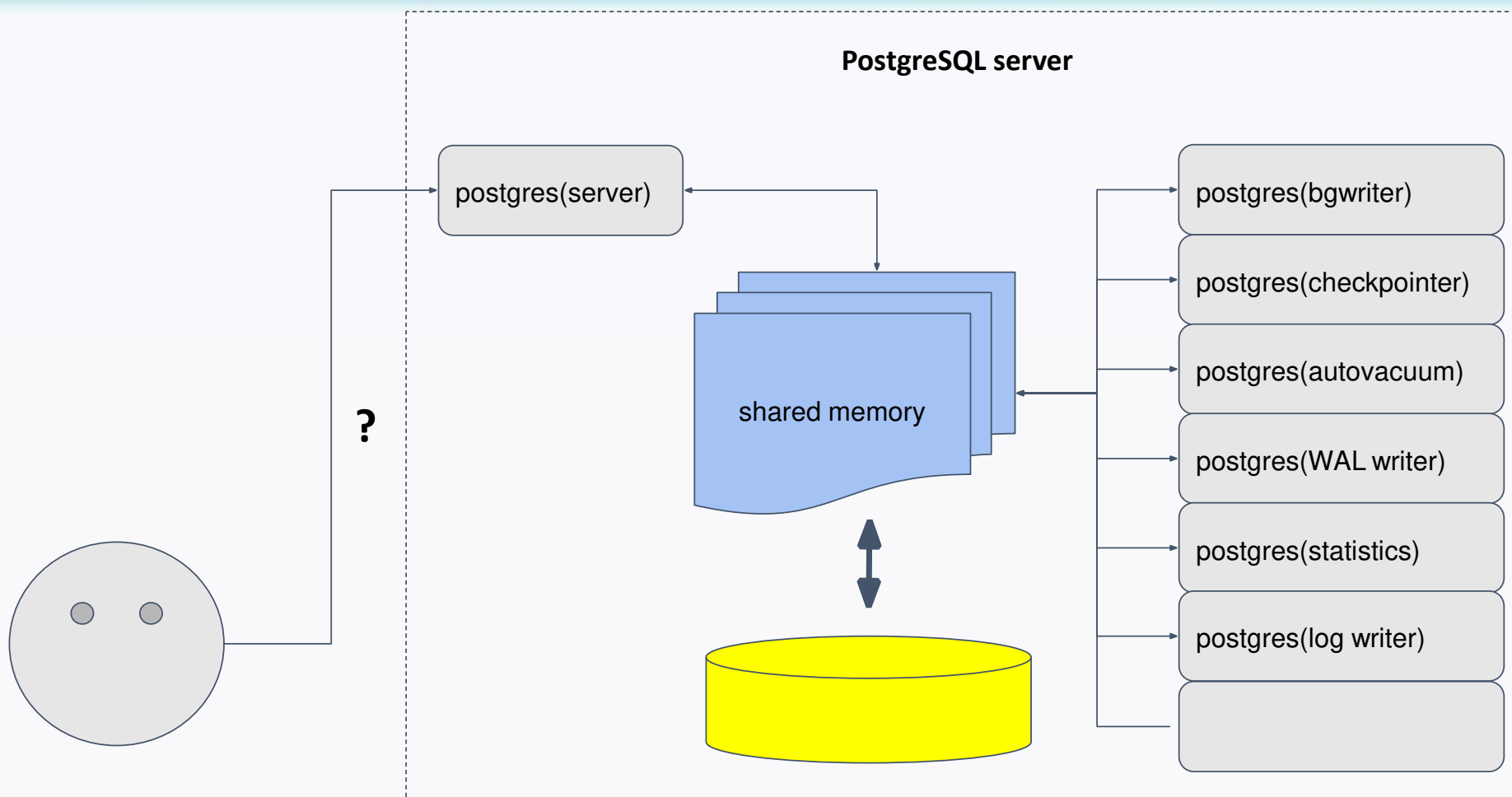
Серверные процессы и память



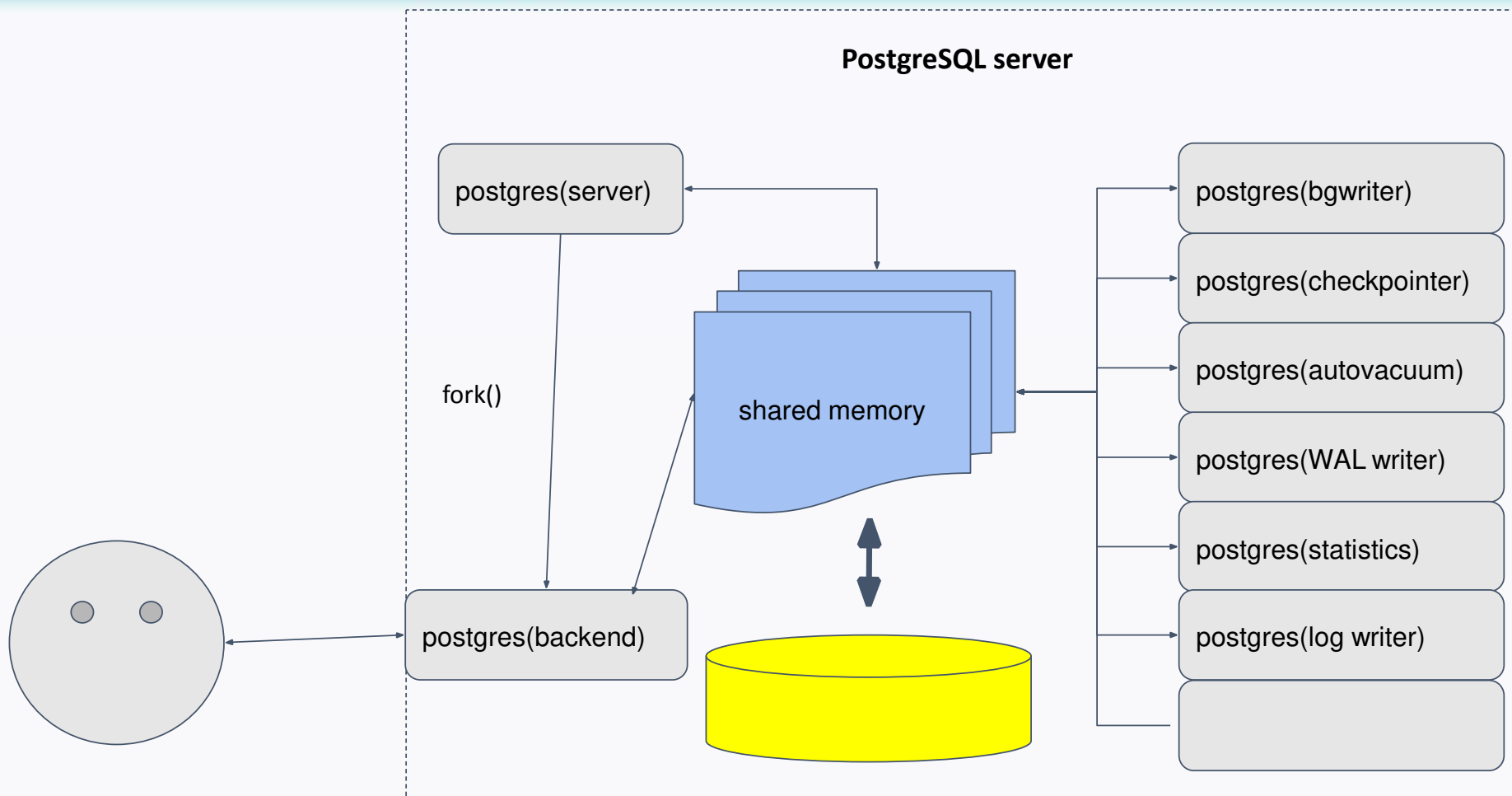
Серверные процессы и память



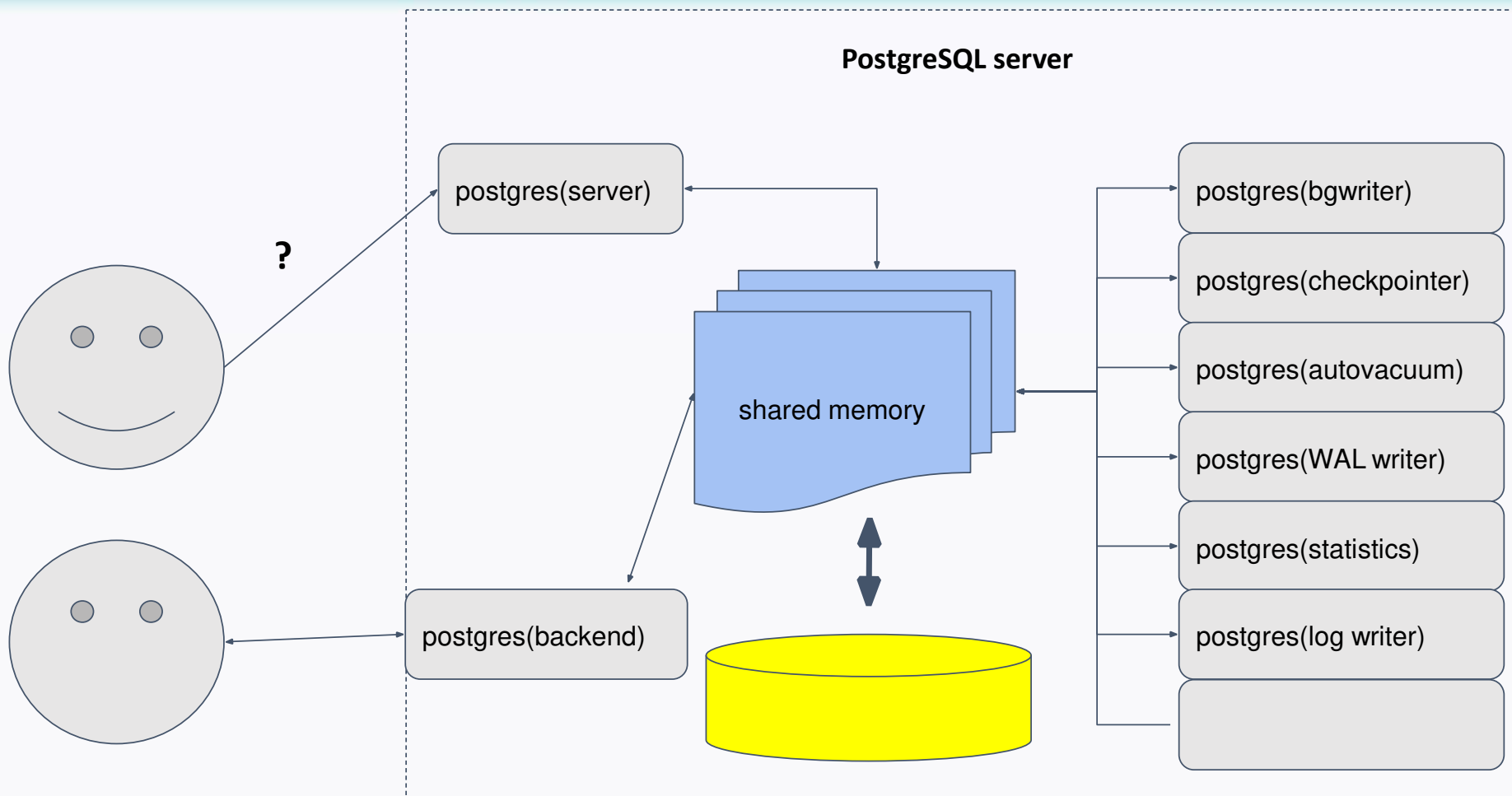
Серверные процессы и память



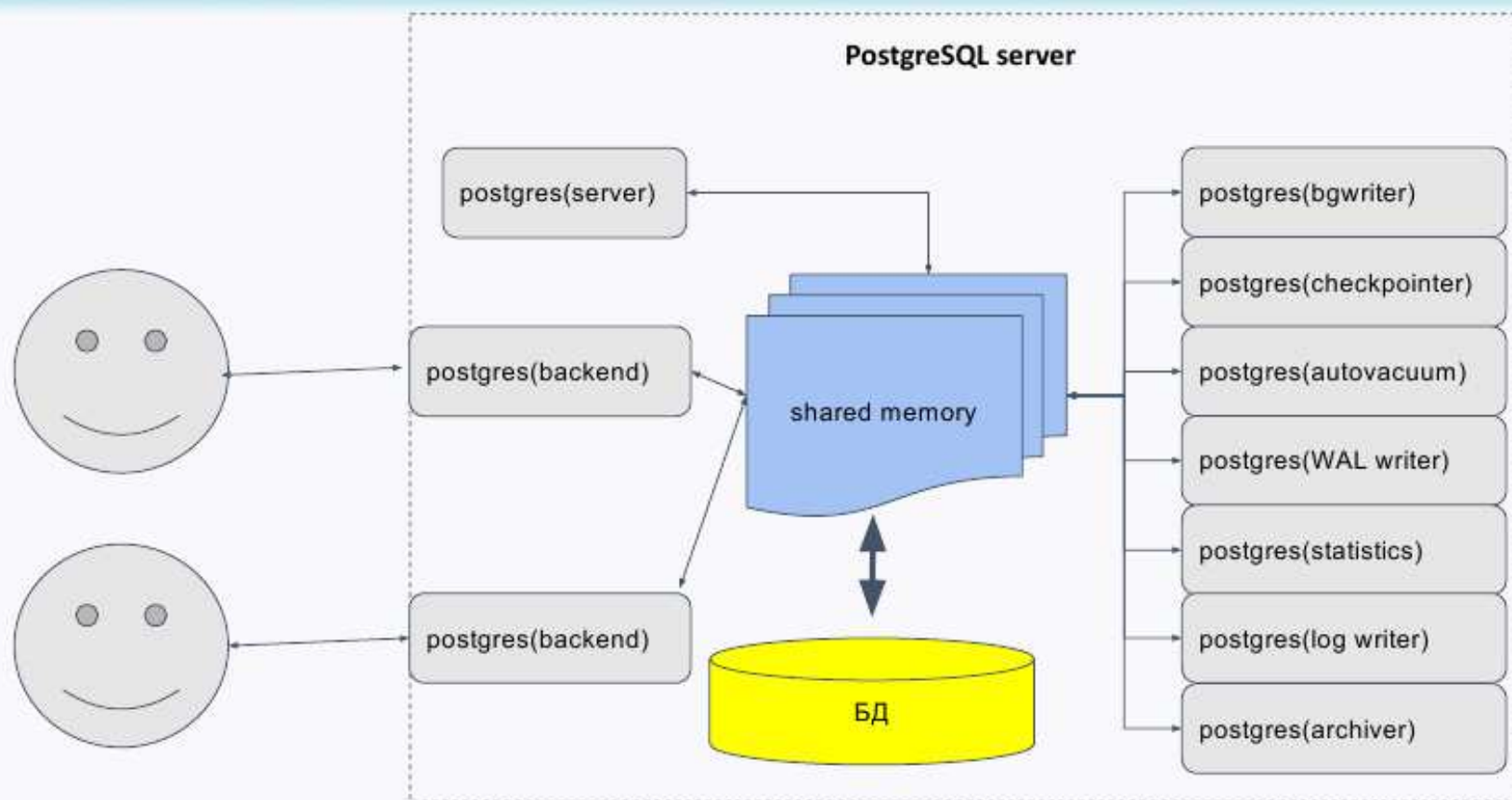
Серверные процессы и память



Серверные процессы и память



Серверные процессы и память



Серверные процессы и память

postgres server process (aka postmaster)

- первый процесс postgres
- запускается при старте сервиса
- порождает все остальные процессы
- создает shared memory
- слушает TCP и Unix socket

Серверные процессы и память

backend processes (aka postgres)

- запускается postmaster'ом
- обслуживает сессию
- работает пока сессия активна
- максимальное количество определяется параметром `max_connections` (по умолчанию 100)

Серверные процессы и память

background processes

- запускаются postmaster'ом
- при старте сервиса
- выделенная роль у каждого процесса
 - logger
 - checkpointer
 - writer
 - wal writer
 - Autovacuum launcher
 - archiver
 - stats collector

Серверные процессы и память

background processes

- запускаются postmaster'ом
- при старте сервиса
- выделенная роль у каждого процесса
 - **logger** (запись сообщений в лог файл)
 - **checkpointer** (запись грязных страниц из buffer cache на диск при наступлении checkpoint)
 - **bgwriter** (проактивная запись грязных страниц из buffer cache на диск)
 - **walwriter** (запись wal buffer в wal file)
 - **autovacuum** (периодический запуск autovacuum)
 - **archiver** (архивация и репликация WAL)
 - **statscollector** (сбор статистики использования по сессиям и таблицам)

Память сессии

- принадлежит backend процессу

- **work_mem (4 MB)**

эта память используется на этапе выполнения запроса для сортировок строк, например ORDER BY и DISTINCT

- **maintenance_work_mem (64MB)**

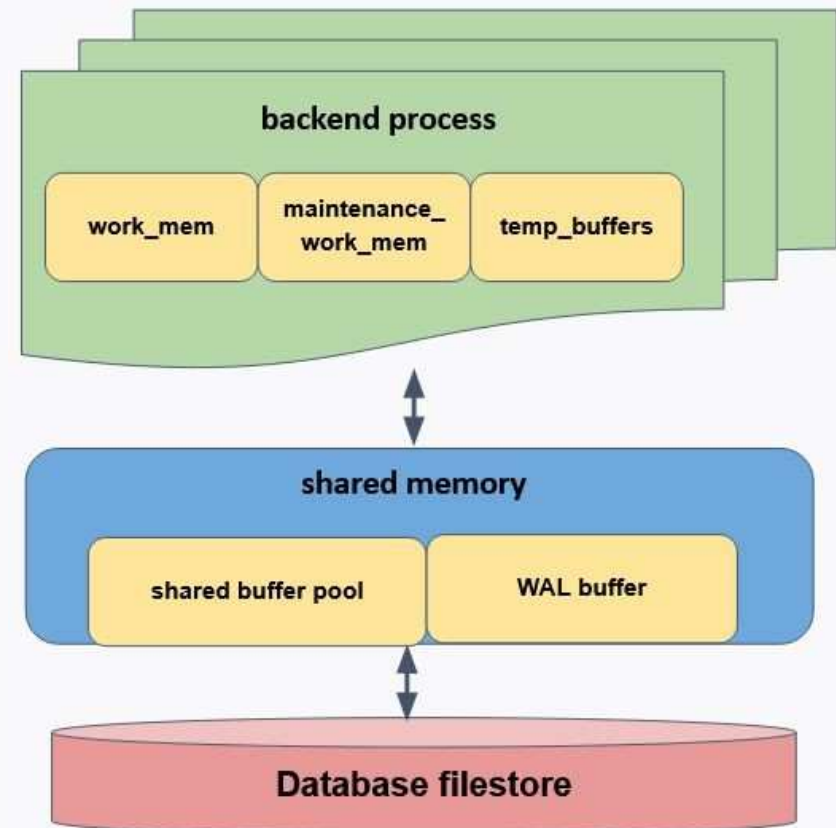
используется служебными операциями типа VACUUM и REINDEX

выделяется только при использовании команд обслуживания в сессии

- **temp_buffers (8 MB)**

используется на этапе выполнения для хранения временных таблиц

<http://www.interdb.jp/pg/pgsql02.html>



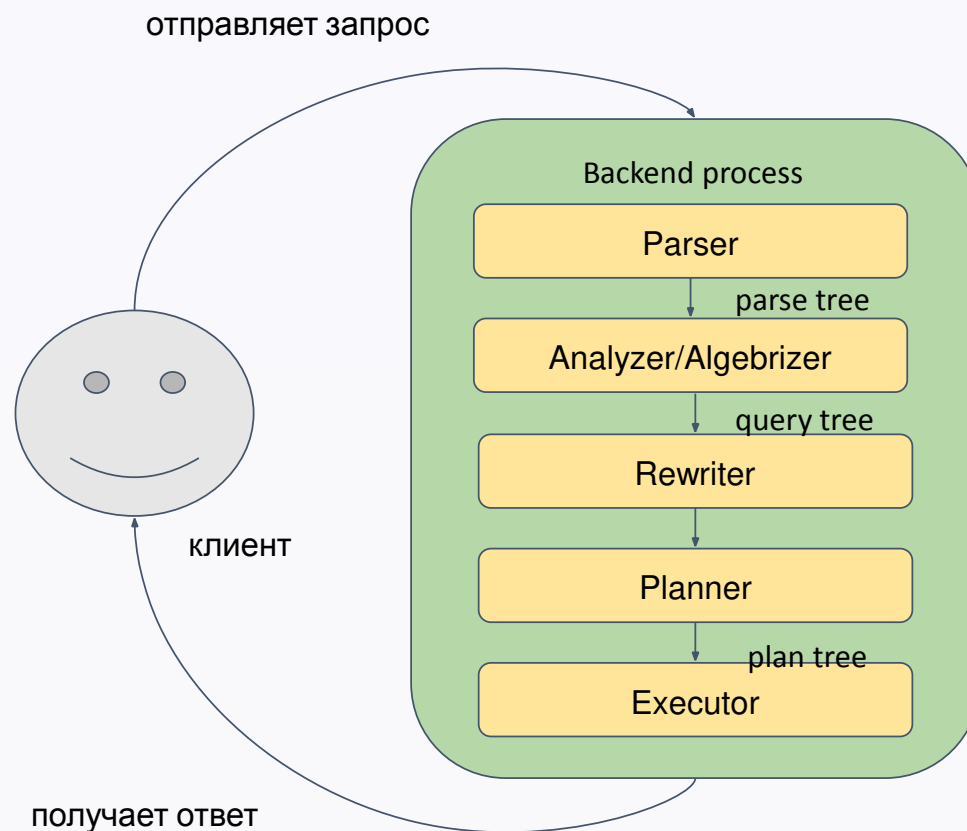
Что происходит внутри сессии

выполняет запрос:

- Parser
- Analyser
- Rewriter
- Planner
- Executor

[Documentation: 12: Chapter 40. The Rule System](#)

особо управлять не можем %)



Что не так с настройками?

Памяти у инстанса 4 Gb (периодически приходил OOM killer)

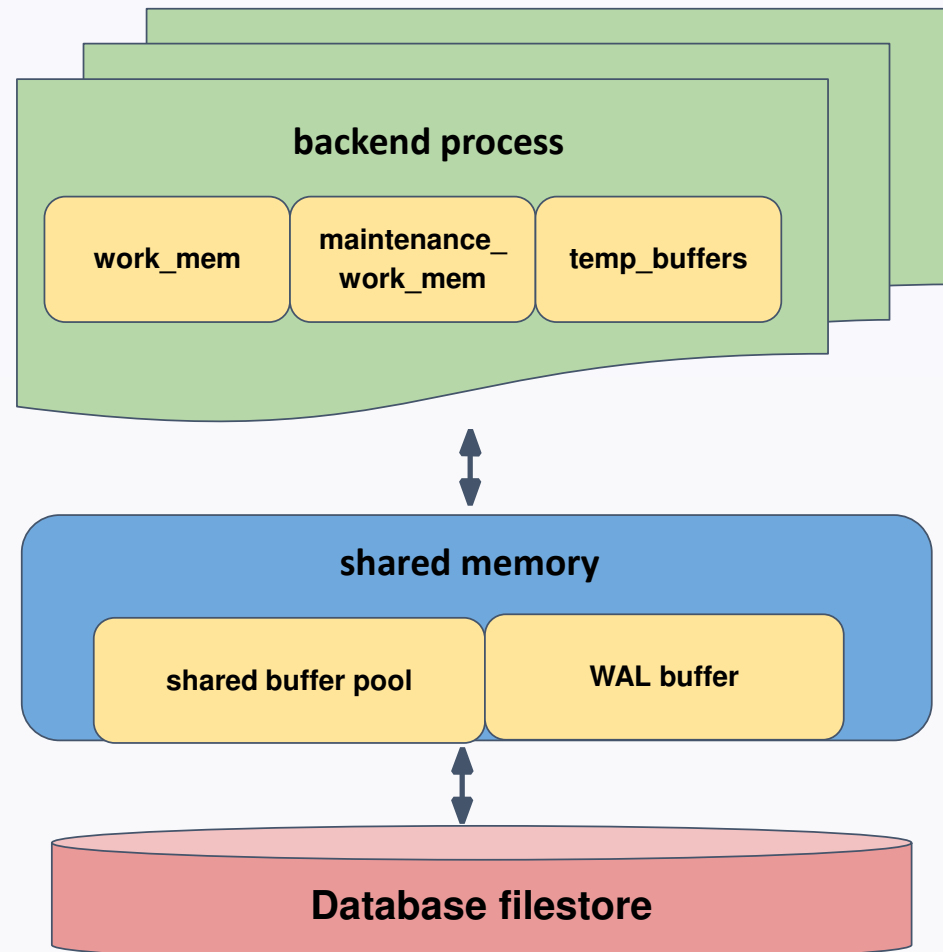
`max_connections = 1000` # (change requires restart)

`shared_buffers = 6GB` # min 128kB

`work_mem = 16MB` # min 64kB

`maintenance_work_mem = 256MB` # min 1MB

Что не так с настройками?



Практика

посмотрим под postgres

`ps -xf`

Сравниваем с запросом:

```
SELECT pid, backend_type, backend_start, state FROM pg_stat_activity;
```




Вопросы?

The background of the slide is a high-angle, blue-tinted aerial photograph of a city skyline, likely New York City, showing numerous skyscrapers and buildings. A semi-transparent blue band with a white geometric network pattern of dots and lines runs horizontally across the middle of the image, serving as a backdrop for the title.

Физическая структура данных

Физическая структура данных

- PostgreSQL работает с данными на дисках только через файловую систему
- EXT3/4 и XFS наиболее популярны
- Raw devices не поддерживаются
- Best practices:
 - не хранить данные в корневой файловой системе
 - отдельная файловая система для каждого табличного пространства
 - в случае внешнего файлового хранилища - отдельный каталог для каждого табличного пространства

Табличное пространство

табличное пространство

- отдельный каталог с точки зрения файловой системы
- лучше делать отдельную файловую систему
- одно табличное пространство может использоваться несколькими базами данных

по умолчанию:

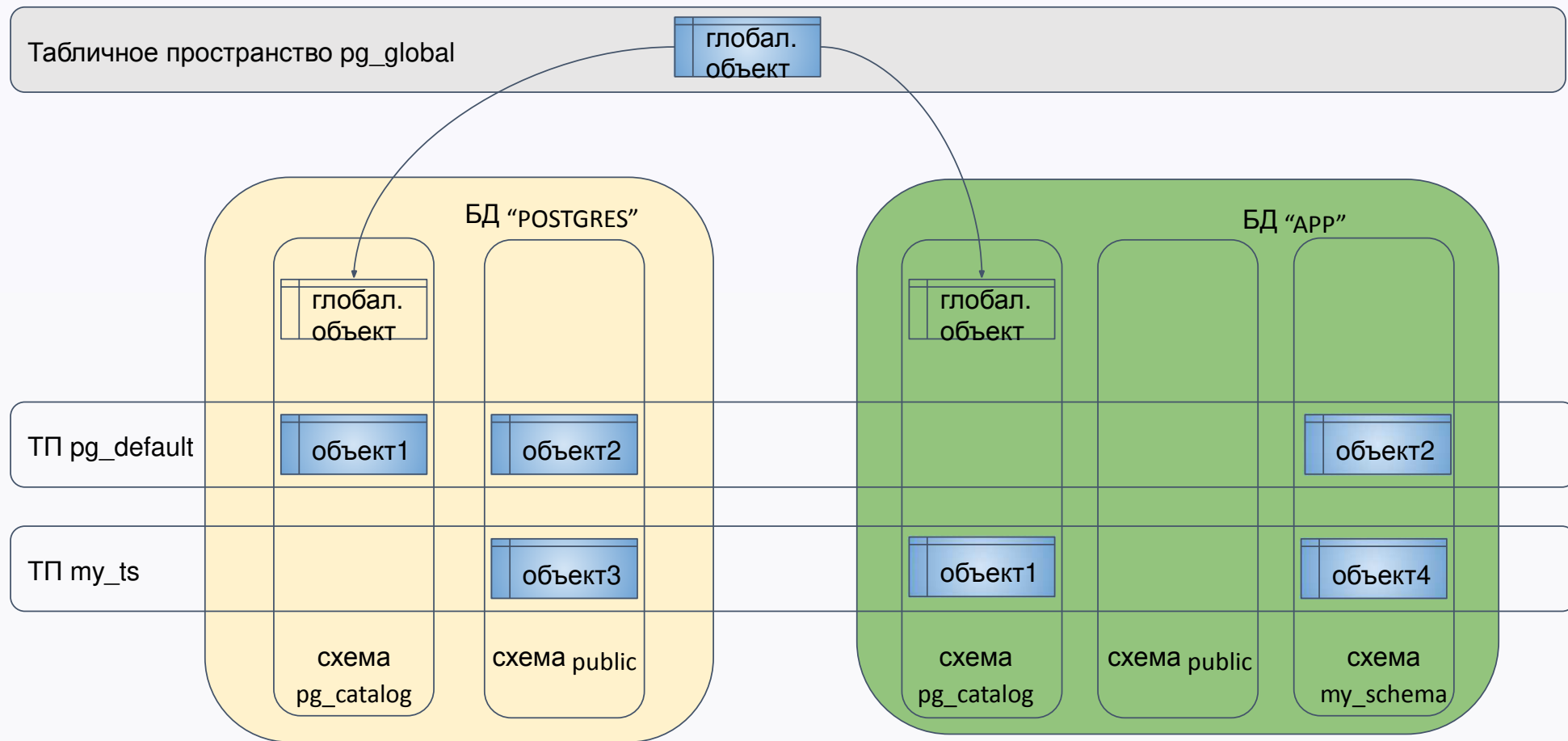
select * from pg_tablespace;

- pg_default - \$PGDATA/base
- pg_global - \$PGDATA/global

новые табличные пространства:

- создаются в \$PGDATA/pg_tblspc

Табличное пространство (ТП)



Табличное пространство. Где файлы?

Табличное пространство pg_global

глобал.
объект

\$PGDATA/global

БД "POSTGRES"

ТП pg_default

объект2

\$PGDATA/base/OID(db)

БД "APP"

объект2

ТП my_ts

объект3

\$PGDATA/pg_tblspc/OID(ts)/путь_к_каталогу/ver/OID(db)

объект4

Кластер PostgreSQL

- PostgreSQL cluster

несколько баз данных

под управлением одного сервера

по умолчанию:

- template0
- template1
- postgres

template0

- для восстановления из резервной копии
- по умолчанию даже нет прав на connect
- лучше всего не создавать в ней никаких объектов
- а еще лучше про нее забыть и не вспоминать ;)

template1

- используется как шаблон для создания новых баз данных
- в нем имеет смысл делать некие действия которые не
- хочется делать каждый раз при создании новых баз данных
- например `create extension` или `create schema`
- но (как мне кажется) лучше не создавать объектов

postgres

- первая база данных для регулярной работы
- создается по умолчанию
- хорошая практика - также не использовать
- но и не удалять - иногда нужна для различных утилит

*в качестве шаблона для создания БД можно использовать любую другую БД - при этом возникает соблазн использовать **create database with template** для копирования БД. Так делать не рекомендовано. Вот выдержка из документации*

***CREATE DATABASE will fail if any other connection exists when it starts; otherwise, new connections to the template database are locked out until CREATE DATABASE completes** используйте штатные методы бэкапов или репликации. обсудим на соответствующих темах*

Таблицы

для каждой таблицы создается до 3-х файлов, **каждый до 1 Гб**, если превышает, то создается файл NNN.1 NNN.2 и т.д. также для FSM и VM:

- файл с данными - OID таблицы
- файл со свободными блоками - OID_fsm

отмечает свободное пространство в страницах после очистки

используется при вставке новых версий строк

существует для всех объектов

- файл с таблицей видимости - OID_vm

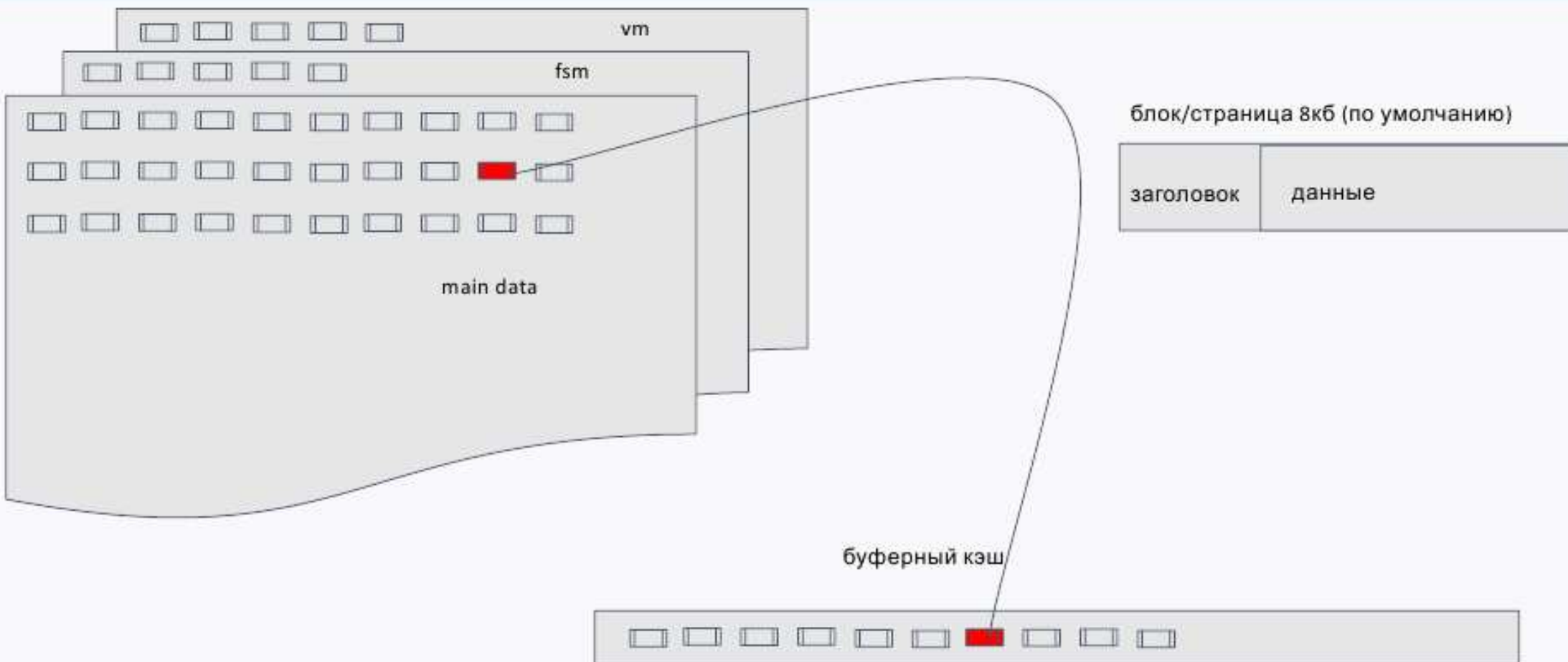
отмечает страницы, на которых все версии строк видны во всех снимках

используется для оптимизации работы процесса очистки и ускорения индексного доступа

существует только для таблиц

иными словами, это страницы, которые давно не изменялись и успели полностью очиститься от неактуальных версий

Таблицы



TOAST

Версия строки должна помещаться на одну страницу

- можно сжать часть атрибутов,
- или вынести в отдельную TOAST-таблицу,
- или сжать и вынести одновременно

TOAST-таблица

- схема pg_toast
- поддерживается собственным индексом
- «длинные» атрибуты разделены на части размером меньше страницы
- читается только при обращении к «длинному» атрибуту
- собственная версия (если при обновлении toast часть не меняется, то и не будет создана новая версия toast части)
- работает прозрачно для приложения
- *стоит задуматься, когда пишем select **

The image features a high-angle, aerial view of a city skyline, likely New York City, with numerous skyscrapers and buildings. The entire image is overlaid with a semi-transparent blue layer. A network of thin, light blue lines connects various points across the blue area, creating a digital or technological aesthetic. The word "Практика" is centered in the middle of the image in a white, sans-serif font.

Практика

The image features a high-angle, aerial view of a city skyline, likely New York City, with numerous skyscrapers and buildings. The entire image is overlaid with a semi-transparent blue layer. A network of thin, light blue lines connects various points across the blue area, creating a web-like pattern. The word "Вопросы?" is written in white, sans-serif font, centered horizontally and vertically within the blue area.

Вопросы?

ДЗ

1 вариант:

-создайте виртуальную машину с Ubuntu 20.04 LTS в GCE типа e2-medium в default VPC в любом регионе и зоне, например us-central1-a

- поставьте на нее PostgreSQL через `sudo apt`
- проверьте что кластер запущен через `sudo -u postgres pg_lsclusters`
- зайдите из под пользователя postgres в psql и сделайте произвольную таблицу с произвольным содержимым

```
postgres=# create table test(c1 text);
```

```
postgres=# insert into test values('1');
```

```
\q
```

- остановите postgres например через `sudo -u postgres pg_ctlcluster 13 main stop`
- создайте новый standard persistent диск GKE через Compute Engine -> Disks в том же регионе и зоне что

GCE инстанс размером например 10GB

- добавьте свежесозданный диск к виртуальной машине - надо зайти в режим ее редактирования и дальше выбрать пункт attach existing disk
- проинициализируйте диск согласно инструкции и подмонтировать файловую систему, только не забывайте менять имя диска на актуальное, в вашем случае это скорее всего будет `/dev/sdb` -

<https://www.digitalocean.com/community/tutorials/how-to-partition-and-format-storage-devices-in-linux>

- сделайте пользователя postgres владельцем `/mnt/data` - `chown -R postgres:postgres /mnt/data/`
- перенесите содержимое `/var/lib/postgres/13` в `/mnt/data` - `mv /var/lib/postgresql/13 /mnt/data`- попытайтесь запустить кластер - `sudo -u postgres pg_ctlcluster 13 main start`

ДЗ

- напишите получилось или нет и почему
 - задание: найти конфигурационный параметр в файлах расположенных в /etc/postgresql/13/main который надо поменять и поменяйте его
 - напишите что и почему меняли
 - попытайтесь запустить кластер - `sudo -u postgres pg_ctlcluster 13 main start`
 - напишите получилось или нет и почему
 - зайдите через `psql` и проверьте содержимое ранее созданной таблицы
 - задание со звездочкой *: не удаляя существующий GCE инстанс сделайте новый, поставьте на его PostgreSQL, удалите файлы с данными из /var/lib/postgres, перемонтируйте внешний диск который сделали ранее от первой виртуальной машины ко второй и запустите PostgreSQL на второй машине так чтобы он работал с данными на внешнем диске, расскажите как вы это сделали и что в итоге получилось.
- ДЗ оформите в markdown на github с описанием что делали и с какими проблемами столкнулись. Также приложите имя Гугл проекта, где пользователь ifti@yandex.ru добавлен в роли project editor.
- * не забываем останавливать инстансы. Сам запущу если нужно

ДЗ

Выполнение ДЗ: 10 баллов

+5 баллов за создание со*

+2 балла за красивое решение

-2 балла за рабочее решение и недостатки указанные преподавателем не устранены