

Лабораторная работа №01: Архитектура СУБД и конфигурация

Цель работы: Изучить базовые компоненты архитектуры PostgreSQL (процессы, память) и получить практические навыки управления конфигурационными параметрами сервера на разных уровнях (экземпляр, сеанс). Освоить работу с основными и дополнительными файлами конфигурации, а также с представлениями `pg_settings` и `pg_file_settings`.

Стек технологий:

- **ОС:** Xubuntu 24.04 64-bit (предустановлена в виртуальной машине)
- **СУБД:** PostgreSQL 16
- **Утилиты:** `psql`, `sudo`, текстовые редакторы (`nano`, `vim` и др.)
- **Ключевые понятия:** `postgresql.conf`, `postgresql.auto.conf`, `pg_settings`, `pg_file_settings`, `ALTER SYSTEM`, `SET`, `SHOW`

Теоретическая часть (краткое содержание из лекции): Архитектура PostgreSQL основана на модели «процесс на подключение». Основной процесс (`postmaster`) управляет кластером, принимает подключения и порождает для каждого клиента свой обслуживающий процесс (`backend`). Процессы взаимодействуют через структуры в **общей памяти** (разделяемый буферный кеш), у каждого процесса также есть своя **локальная память**. Поведение сервера управляется **параметрами конфигурации**. Основные источники их значений:

1. **Файлы конфигурации:** `postgresql.conf` (основной), `postgresql.auto.conf` (управляемый командой `ALTER SYSTEM`), файлы в каталогах, подключаемых через `include_dir`.
2. **Команды уровня сеанса:** `SET` (для текущего сеанса или транзакции). Параметры имеют разный **контекст** (`internal`, `postmaster`, `sighup`, `user` и др.), определяющий необходимые для их применения действия (перезагрузка сервера, перечитывание конфигурации или немедленное применение).

Задание на практическую реализацию:

Часть 1: Исследование параметров и файлов конфигурации

1. **Текущая конфигурация:** Подключитесь к серверу с помощью `psql`. Определите расположение основного файла конфигурации (`postgresql.conf`) с помощью команды `SHOW config_file;`.
2. **Анализ параметров:** Изучите представление `pg_settings`. Найдите параметры, для изменения которых требуется перезагрузка сервера (`context = 'postmaster'`). Найдите 2-3 параметра с контекстом `sighup` и `user`.
3. **Анализ файлов:** Изучите представление `pg_file_settings`. Определите, из каких файлов и с какими значениями были считаны текущие настройки параметров `shared_buffers` и `work_mem`.

Часть 2: Управление параметрами на уровне экземпляра

1. **Изменение через `ALTER SYSTEM`:** Используя команду `ALTER SYSTEM`, установите для параметра `work_mem` новое значение. Убедитесь, что изменение записалось в файл `postgresql.auto.conf` (используйте функцию `pg_read_file`). Примените изменение, перечитав конфигурацию (`SELECT pg_reload_conf();`). Проверьте новое значение параметра и его источник в `pg_settings`.
2. **Изменение через дополнительный файл:** Создайте файл в каталоге, указанном в директиве `include_dir` основного конфигурационного файла. Установите в этом файле значение для

параметра `log_min_duration_statement`. Примените изменение и проверьте его.

3. **Ошибка в конфигурации:** Намеренно внесите синтаксическую ошибку в один из конфигурационных файлов (например, `invalid_value` вместо числового значения). Попытайтесь перечитать конфигурацию. Изучите представление `pg_file_settings`, чтобы найти запись об ошибке. Исправьте ошибку и перечитайте конфигурацию.

Часть 3: Управление параметрами на уровне сеанса

1. **Команда SET:** В рамках сеанса измените значение параметра `work_mem` с помощью `SET`. Проверьте новое значение. Завершите транзакцию с помощью `ROLLBACK` и проверьте значение параметра again. Объясните результат.
2. **Команда SET LOCAL:** Откройте транзакцию (`BEGIN`). Inside the transaction, use `SET LOCAL` to change the `work_mem` parameter. Verify the change. After committing the transaction (`COMMIT`), check the parameter value again. Explain the result.
3. **Пользовательский параметр:** Создайте и установите значение для пользовательского параметра (имя должно содержать точку, например, `app.my_setting`). Прочитайте его значение с помощью `current_setting`.

Требования к оформлению и отчету:

1. **Скрипты:** Предоставьте последовательность всех выполненных SQL-команд и команд ОС.
2. **Проверка:** Для каждого действия приложите вывод команд, подтверждающих успешность выполнения (вывод `SHOW`, `pg_settings`, `pg_file_settings`).
3. **Отчет:** Краткий отчет должен содержать:
 - Объяснение разницы между контекстами параметров `postmaster`, `sighup` и `user`.
 - Объяснение разницы между применением изменений через `ALTER SYSTEM` и через `SET/SET LOCAL`.
 - Описание процедуры поиска и исправления ошибки в конфигурационном файле.

Критерии оценки:

- **Удовлетворительно:** Выполнены Часть 1 и задание 1 из Части 2. Предоставлены основные проверочные выводы.
- **Хорошо:** Дополнительно выполнены задания 2 и 3 из Части 2, а также задание 1 из Части 3. Отчет содержит объяснения по контекстам параметров.
- **Отлично:** Полностью выполнены все части работы. В отчете содержатся четкие объяснения различий между способами управления параметрами и механизма их применения.

Рекомендуемая литература:

1. Курс "Администрирование PostgreSQL 16. Базовый курс":
<https://postgrespro.ru/education/courses/DBA1>
2. Курс "Администрирование PostgreSQL 16. Настройка и мониторинг":
<https://postgrespro.ru/education/courses/DBA2>
3. Курс "Администрирование PostgreSQL 16. Резервное копирование и репликация":
<https://postgrespro.ru/education/courses/DBA3>
4. Книги по PostgreSQL: <https://postgrespro.ru/education/books>