

Postgres Pro Enterprise 13

Редакции и возможности



Авторские права

© Postgres Professional, 2023 год.

Авторы: Алексей Береснев, Илья Баштанов, Павел Толмачев

Использование материалов курса

Некоммерческое использование материалов курса (презентации, демонстрации) разрешается без ограничений. Коммерческое использование возможно только с письменного разрешения компании Postgres Professional. Запрещается внесение изменений в материалы курса.

Обратная связь

Отзывы, замечания и предложения направляйте по адресу:

edu@postgrespro.ru

Отказ от ответственности

Компания Postgres Professional не несет никакой ответственности за любые повреждения и убытки, включая потерю дохода, нанесенные прямым или косвенным, специальным или случайным использованием материалов курса. Компания Postgres Professional не предоставляет каких-либо гарантий на материалы курса. Материалы курса предоставляются на основе принципа «как есть» и компания Postgres Professional не обязана предоставлять сопровождение, поддержку, обновления, расширения и изменения.

Продукты Postgres Professional и редакции Postgres Pro
Возможности Postgres Pro Enterprise

Редакции Postgres Pro

Лицензионная политика

Другие продукты Postgres Professional

Standard
Certified
Enterprise
Enterprise Certified

Postgres Pro поставляется в различных редакциях.

Standard – доработанная версия базового PostgreSQL с некоторыми дополнительными расширениями. Это основной выбор для клиентов, которых устраивают базовые возможности, но которым необходима техподдержка:

<https://postgrespro.ru/products/postgrespro/standard>

Enterprise – существенно доработанная версия PostgreSQL с функциональностью, предназначенной для высоконагруженных систем в больших компаниях:

<https://postgrespro.ru/products/postgrespro/enterprise>

Обе редакции имеют сертифицированные ФСТЭК версии, которые могут применяться для хранения персональных данных и конфиденциальной информации:

<https://postgrespro.ru/products/postgrespro/certified>

<https://postgrespro.ru/products/postgrespro/enterprisecertified>

Начиная с версии 15, ОС Windows не поддерживается Postgres Pro, но компания собирает обычный PostgreSQL под Windows. Сборка содержит программу-установщик.

<https://postgrespro.ru/windows>

В этом курсе, если не указано иное, мы рассматриваем возможности Postgres Pro Enterprise.

Лицензирование по количеству ядер

в стоимость лицензии входит годовая техническая поддержка
редакция Standard бесплатна для образовательных учреждений

Для 1С возможно лицензирование по количеству серверов
и пользователей

Postgres Pro лицензируется по количеству ядер. В стоимость лицензии входит один год технической поддержки.

Редакция Postgres Pro Standard может бесплатно использоваться образовательными учреждениями.

Все редакции Postgres Pro поддерживают работу с 1С: в состав входят все необходимые расширения (fasttrun, fulleq, mchar, online_analyze), а ядро включает оптимизации производительности (например, временные таблицы не требуют ввода-вывода, пока помещаются в отведенной оперативной памяти).

При использовании Postgres Pro для работы с 1С можно выбрать другую модель лицензирования – по количеству серверов и пользователей.

Дополнительно в компании можно заказать услуги технической поддержки, миграции, аудита информационных систем:

<https://postgrespro.ru/services>

Подробную информацию по стоимости, срокам лицензии и по вопросам предоставления услуг можно узнать, написав на sales@postgrespro.ru

Postgres Pro Enterprise Manager

интегрированная панель управления
централизованный интерфейс мониторинга и администрирования

Postgres Pro Backup Enterprise

утилита резервного копирования и восстановления pg_probackup
(рассматривается в этом курсе)

Postgres Pro Shardman

распределенная отказоустойчивая СУБД с ACID-гарантиями
горизонтальное масштабирование по чтению и записи

Кроме СУБД Postgres Pro, компания Postgres Professional предлагает ряд других продуктов.

Postgres Pro Enterprise Manager – интегрированная административная панель управления для Postgres Pro Enterprise. Предоставляет централизованный интерфейс мониторинга и управления СУБД.

<https://postgrespro.ru/products/PPEM>

Postgres Pro Backup Enterprise – утилита pg_probackup для управления резервным копированием и восстановлением кластеров баз данных PostgreSQL. Утилита работает с обычным PostgreSQL и со всеми редакциями Postgres Pro, но в сочетании с Postgres Pro Enterprise поддерживает CFS и хранение резервных копий в S3-хранилище.

Утилита входит в состав Postgres Pro Enterprise и рассматривается в этом курсе.

https://postgrespro.ru/products/pg_probackup

Postgres Pro Shardman – распределенная отказоустойчивая СУБД на базе PostgreSQL. Shardman не основан на Postgres Pro, но использует многие ее возможности (сбор статистики, аудит, сжатая файловая система).

<https://postgrespro.ru/products/shardman>

Поддержка локалей ICU на уровне кластера и БД

Замена нулевого байта в COPY FROM

nul_byte_replacement_on_import

oracle_fdw

обертка сторонних данных для Oracle

orafce

популярные функции из пакетов Oracle

Тема 2. Установка,
настройка, обновление

Postgres Pro Enterprise позволяет указывать локаль провайдера ICU при инициализации кластера и при создании базы данных. PostgreSQL поддерживает эту возможность начиная с версии 15.

Команда COPY FROM может обрабатывать начальные нулевые байты, которые могут появиться при выгрузке данных из других СУБД.

Поведение регулируется параметром *nul_byte_replacement_on_import*.

<https://postgrespro.ru/docs/postgrespro/13/runtime-config-compatible#GUC-NUL-BYTE-REPLACEMENT-ON-IMPORT>

Для облегчения миграции с Oracle в Postgres Pro Enterprise добавлены два сторонних расширения:

- `oracle_fdw` — обертка сторонних данных для Oracle;
- `orafce` — распространенные функции из пакетов Oracle.

Управление транзакциями и сеансами

64-разрядный счетчик транзакций

Автономные транзакции

Встроенный пул соединений

сохраняет контекст сеанса

Тема 3. Управление
транзакциями

32-битные идентификаторы транзакций в PostgreSQL создают проблемы с обслуживанием переполнения счетчика транзакций. в нагруженных системах. В Postgres Pro Enterprise используются 64-битные идентификаторы, упрощающие администрирование.

Автономные транзакции позволяют зафиксировать или откатить часть операторов независимо от статуса основной транзакции.

В обычном режиме работы сервера каждое подключение обслуживается отдельным процессом, что уменьшает эффективность совместной работы. В Postgres Pro Enterprise реализован встроенный пул соединений, сохраняющий окружения сеансов.

`pg_variables`

переменные в памяти сеанса

`in_memory`

небольшие таблицы в общей памяти

Расширение `pg_variables` позволяет работать с переменными различных типов в памяти сеанса. Расширение несовместимо с пулом соединений, включая встроенный пул Postgres Pro.

<https://postgrespro.ru/docs/enterprise/13/pg-variables>

С помощью расширения `in_memory` можно размещать небольшие таблицы в общей памяти сервера. Работа с таблицей происходит с помощью обертки сторонних данных.

<https://postgrespro.ru/docs/enterprise/13/in-memory>

`pgbouncer` — сторонний пул соединений

Параметры `libpq` для подключения к разным серверам

`hostorder`, `failover_timeout`

Отключение сеансов и переподключение

`idle_session_timeout`, `reusepass`

Функции для изменения параметров других сеансов

`pg_backend_set_config`, `pg_backend_load_library`

Помимо встроенного пула соединений, в Postgres Pro включен сторонний пул соединений `pgbouncer`. Работа с `pgbouncer` с точки зрения разработчика приложений рассматривается в курсе DEV2.

Доработанный протокол `libpq` позволяет распределять сеансы по нескольким серверам:

- `hostorder` — порядок подключения к серверам (`sequential/random`);
- `failover_timeout` — максимальное время для циклического переподключения к узлам, указанным в строке соединения.

<https://postgrespro.ru/docs/enterprise/13/libpq-connect>

Два новых конфигурационных параметра:

- `idle_session_timeout` — завершает любые сеансы, простаивающие дольше заданного времени вне открытой транзакции;
- `reusepass` — разрешается ли использовать запрошенный пароль при повторном установлении подключения (запрет автоматического переподключения).

Добавлены функции `pg_backend_set_config` и `pg_backend_load_library`, позволяющие менять параметры других сеансов и загружать в них библиотеки.

<https://postgrespro.ru/docs/enterprise/13/functions-admin#FUNCTIONS-ADMIN-SET>

Приоритеты сеансов в отношении ресурсов

<i>session_cpu_weight</i>	— использование процессора
<i>session_ioread_weight</i>	— объем чтения
<i>session_iowrite_weight</i>	— объем записи

Фоновый процесс собирает статистику использования ресурсов сеансами

usage_tracking_interval = 0 — интервал сбора

В высоконагруженных системах возникает потребность неравномерного распределения ограниченных ресурсов между сеансами. В обычной версии PostgreSQL такого механизма нет.

В Postgres Pro Enterprise можно назначить сеансам приоритеты в отношении трех видов ресурсов:

- использование процессора;
- объем чтения (страницы локального и общего кешей);
- объем записи (страницы локального и общего кешей).

Использование ресурсов процессами определяется на основе статистики, собираемой отдельным фоновым рабочим процессом.

Интервал сбора статистики задается параметром *usage_tracking_interval*. По умолчанию его значение равно 0, при этом статистика не собирается и приоритизация отключена.

<https://postgrespro.ru/docs/enterprise/13/kernel-resources#RESOURCE-PRIORITIZATION>

Возможности, связанные с хранением данных в файлах

CFS — Compressed File System

включается для отдельных табличных пространств
позволяет сжать данные в несколько раз
сокращает объем дискового ввода-вывода
измененные страницы записываются последовательно

Тема 4. CFS – сжатая
файловая система

Использование сжатой файловой системы (Compressed File System, CFS) позволяет сократить объем дискового ввода-вывода за счет хранения сжатых блоков данных и улучшить кучность хранения за счет последовательной записи сжатых блоков в конец файла на диске.

Механизм CFS включается на уровне отдельных табличных пространств.

Как правило, сжатие позволяет уменьшить объем хранимых данных в несколько раз.

<https://postgrespro.ru/docs/enterprise/13/cfs>

TOAST для системного каталога

для столбцов со списками управления доступом

`pg_transfer`

быстрое перемещение таблиц между серверами

`pg_repack`

устранение раздувания таблиц и индексов

Таблицы системного каталога, содержащие потенциально длинные списки управления доступом (Access Control List), такие как `pg_class` и `pg_attribute`, могут использовать TOAST.

С помощью расширения `pg_transfer` можно специальным образом подготовить таблицу, содержащую много данных, и затем быстро переместить ее на другой экземпляр Postgres Pro простым копированием файлов данных. Расширение выполняет необходимые для этого корректировки системного каталога.

<https://postgrespro.ru/docs/enterprise/13/pgtransfer>

В Postgres Pro Enterprise включена сторонняя утилита `pg_repack`, которая позволяет избавиться от раздувания таблиц и индексов без установки исключительной блокировки на все время работы, как того требует команда `VACUUM FULL`.

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

`amcheck`

контроль целостности
добавлена проверка уникальности индексов

`pg_filedump`

низкоуровневый просмотр файлов данных

Просмотр всех версий строк

функция `pg_snapshot_any` для отключения правил видимости

Postgres Pro Enterprise включает дополнительные средства для контроля целостности, отладки и исследования поврежденных данных.

Штатное расширение `amcheck`, позволяющее проверять целостность данных, доработано для проверки ограничений уникальности в В-деревьях.

<https://postgrespro.ru/docs/enterprise/13/amcheck>

Утилита `pg_filedump` выводит содержимое файлов таблиц и индексов, а также управляющего файла в отформатированном виде. Это может быть полезно для исправления поврежденных данных.

<https://postgrespro.ru/docs/enterprise/13/pg-filedump>

Функция `pg_snapshot_any` выводит все версии строк таблицы, отключая правила видимости.

<https://postgrespro.ru/docs/enterprise/13/functions-admin#FUNCTIONS-DEBUGGING-TABLE>

Выполнение и оптимизация запросов

dump_stat — перенос статистики планировщика

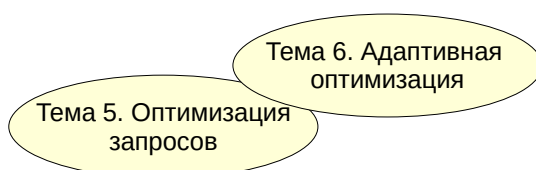
pg_query_state — состояние выполняющегося запроса

pg_hint_plan — указания оптимизатору

plantuner — запрет определенных индексов

AQO — адаптивная оптимизация запросов

накопление статистики выполнений и коррекция прогноза
кардинальности



Расширение dump_stat позволяет после восстановления из логической копии обойтись без выполнения команды ANALYZE для всего кластера, что для больших БД может значительно сократить время простоя.

<https://postgrespro.ru/docs/enterprise/13/dump-stat>

Расширение pg_query_state позволяет узнать текущее состояние выполнения запросов.

<https://postgrespro.ru/docs/enterprise/13/pg-query-state>

Расширение pg_hint_plan позволяет управлять планом выполнения запроса с помощью специальных указаний.

<https://postgrespro.ru/docs/enterprise/13/pg-hint-plan>

Расширение plantuner позволяет скрывать определенные индексы от планировщика, не удаляя их.

<https://postgrespro.ru/docs/enterprise/13/plantuner>

Расширение AQO (Adaptive Query Optimization) используется для адаптивной оптимизации запросов за счет коррекции прогноза кардинальности с учетом накопленной статистики выполнения запросов.

<https://postgrespro.ru/docs/enterprise/13/aqo>

Удаление замкнутых соединений из планов

enable_self_join_removal

Оптимизированы планы с пустыми узлами Materialize

Уточнение стоимости последовательного сканирования

seq_scan_startup_cost_first_row = off

Уточнение селективности соединений по нескольким условиям при наличии расширенной статистики

enable_compound_index_stats = on

Уменьшено потребление памяти планировщиком в запросах с большими массивами

Добавлена возможность удаления замкнутых соединений из планов запросов, что может увеличить скорость выполнения (параметр *enable_self_join_removal*). Полезно для систем с автоматически генерируемыми запросами.

<https://postgrespro.ru/docs/enterprise/13/runtime-config-query#GUC-ENABLE-SELF-JOIN-REMOVAL>

Улучшена производительность в случаях, когда в плане возникает много пустых узлов Materialize, которые ничего не возвращают.

Параметр *seq_scan_startup_cost_first_row* увеличивает начальную стоимость последовательного сканирования. В обычном PostgreSQL эта величина всегда равную нулю, за счет чего планировщик может иногда предпочесть последовательное сканирование доступа по индексу для получения одной строки по первичному ключу.

<https://postgrespro.ru/docs/enterprise/13/runtime-config-query#GUC-SEQ-SCAN-STARTUP-COST-FIRST-ROW>

При включенном параметре *enable_compound_index_stats* оптимизатор использует составные индексы для уточнения оценки селективности соединения по нескольким атрибутам, не составляющим внешний ключ (в PostgreSQL в этом случае перемножаются селективности по отдельным колонкам).

Добавлена оптимизация, которая уменьшает потребление памяти планировщиком на запросах, содержащих большие массивы (например, IN для проверки вхождения значения в массив).

Управление объемом кеша для подготовленных операторов

plan_cache_lru_memsizes, plan_cache_lru_size

Режим автоподготовки операторов

autoprepares_threshold, autoprepares_limit, autoprepares_memory_limit

Поиск ближайших соседей для B-tree, GiST и SP-GiST

Ускорено сканирование только индекса для больших таблиц

vops — векторные операции для OLAP-запросов

Параметрами *plan_cache_lru_memsizes* и *plan_cache_lru_size* можно ограничить объем памяти, занимаемый подготовленными операторами. Это может помочь, если приложение создает слишком много подготовленных операторов.

<https://postgrespro.ru/docs/enterprise/13/runtime-config-resource#GUC-PLAN-CACHE-LRU-MEMSIZE>

Добавлен режим автоподготовки операторов (отключен по умолчанию), в котором часто используемые операторы неявно подготавливаются. Может давать преимущество в тех случаях, когда явные команды PREPARE не работают, например, когда применяется pgbouncer.

<https://postgrespro.ru/docs/enterprise/13/autoprepares>

В индексах B-tree, GiST и SP-GiST с сортировкой по расстоянию реализована поддержка алгоритма k-NN — поиска ближайших соседей.

<https://postgrespro.ru/docs/enterprise/13/k-nn-search>

Увеличен размер кеша карты видимости, что позволяет сканированию только индекса быстрее работать с таблицами больше 256 Мбайт.

Экспериментальное расширение vops (vectorized operations) позволяет ускорять запросы OLAP с фильтрацией и агрегированием благодаря колоночному хранению внутри страниц («паркет») и использованию векторных операций.

<https://postgrespro.ru/docs/enterprise/13/vops>

Оптимизация механизма проверки взаимоблокировок

Очередь для разделяемых легких блокировок

`lwlock_shared_limit = 0`

Разделение таблицы блокировок на секции

`log2_num_lock_partitions = 4`

Уменьшена конкуренция при расширении файлов B-tree

Одновременная проверка ситуации взаимоблокировки многими процессами (что может произойти на нагруженных системах с большим количеством ядер) вызывает замедление работы сервера вплоть до полной остановки. В Postgres Pro проверка ограничена одним процессом в один момент времени.

Для улучшения работы с легкими блокировками имеется параметр `lwlock_shared_limit`. Он задает количество разделяемых блокировок, по достижении которого запускается «справедливый» режим ожидания — последующие блокировки добавляются в очередь. В этом случае устраняется возможность бесконечного откладывания исключительных блокировок, что важно при большой конкуренции за ресурс.

<https://postgrespro.ru/docs/enterprise/13/runtime-config-locks#GUC-LWLOCK-SHARED-LIMIT>

Для повышения параллелизма таблица блокировок разделяется на несколько секций. Количество секций всегда является степенью двойки, поэтому задается показатель степени. В PostgreSQL он всегда равен 4, то есть таблица делится на 16 секций. В Postgres Pro Enterprise показатель можно задать параметром `log2_num_lock_partitions`, если при большой конкуренции значения по умолчанию не хватает.

<https://postgrespro.ru/docs/enterprise/13/runtime-config-locks#GUC-LOG2-NUM-LOCK-PARTITIONS>

Уменьшена конкуренция при расширении файлов B-tree: к файлу добавляется сразу несколько страниц, как при расширении таблицы.

Более эффективная работа с горячими страницами индексов

Оптимизация для NUMA-архитектур

Уменьшена конкуренция за хеш-таблицу

1024 отдельно блокируемых секций

Улучшен менеджер буферного кеша:

- устранено узкое место, возникающее при обращении к горячим страницам индексов при высокой конкуренции;
- изменена внутренняя реализация хеш-таблицы буферного кеша для более эффективной работы, в том числе на NUMA-архитектурах;
- сама хеш-таблица буферного кеша разделена на 1024 секции (в PostgreSQL — 128) для снижения конкуренции на многоядерных системах.

Метод доступа RUM

фразовый поиск и ранжирование

pg_tparser — альтернативный анализатор

Hunspell — словари для ряда языков

shared_ispell — размещение словарей в общей памяти

JsQuery — язык запросов к данным jsonb

поиск во вложенных объектах и массивах

операторы сравнения с индексной поддержкой

Для улучшения полнотекстового поиска добавлен метод доступа RUM, который похож на GIN, но поддерживает фразовый поиск и выдачу ранжированных результатов.

<https://postgrespro.ru/docs/enterprise/13/rum>

Расширение pg_tparser изменяет стандартную стратегию разбора текста для слов, включающих подчеркивания, цифры и буквы, а также для слов, разделенных знаком минуса. Оно удобно, в частности, для поиска в коде.

<https://postgrespro.ru/docs/enterprise/13/pg-tparser>

Расширение Hunspell предоставляет возможность использовать популярные словари Hunspell для разных языков.

<https://postgrespro.ru/docs/enterprise/13/hunspell-dict>

В отличие от стандартной реализации словарей ispell, расширение shared_ispell позволяет разместить словарь в общей памяти.

<https://postgrespro.ru/docs/enterprise/13/shared-ispell>

Расширение JsQuery предоставляет язык запросов для поиска в значениях типа jsonb. С добавлением в PostgreSQL поддержки стандарта SQL/JSON потеряло актуальность, но сохраняется для совместимости.

<https://postgrespro.ru/docs/enterprise/13/jsquery>

Анализ нагрузки и мониторинг

pgpro_stats — статистика выполнения

развитие pg_stat_statements

pgpro_pwr — выявление ресурсоемких операций: выборки и отчеты

развитие pg_profile

Тема 7. Анализ
производительности

Расширение pgpro_stats позволяет собрать статистику планирования и выполнения SQL-операторов. Основано на pg_stat_statements.

<https://postgrespro.ru/docs/enterprise/13/pgpro-stats>

Pgpro_pwr накапливает статистические выборки (полученные на основе других расширений и результатов работы сборщика статистики), анализирует их и строит отчеты о нагрузке.

<https://postgrespro.ru/docs/enterprise/13/pgpro-pwr>

`pg_wait_sampling` — семплирование событий ожидания

`auto_explain` — добавлено время планирования

`pg_waldump` — добавлен вывод времени

Определение объема сгенерированных WAL-записей

`pgBadger` — поддержка формата журнала Postgres Pro

`Mamonsu` — агент мониторинга для Zabbix

`pgbench` — учет ошибок сериализации и взаимоблокировок

Расширение `pg_wait_sampling` семплирует события ожидания. Расширение подробно рассматривается в курсе DBA2.

<https://postgrespro.ru/docs/enterprise/13/pg-wait-sampling>

В `auto_explain` добавлено отображение длительности планирования.

<https://postgrespro.ru/docs/enterprise/13/auto-explain>

В `pg_waldump` добавлен вывод времени для различных типов журнальных записей.

<https://postgrespro.ru/docs/enterprise/13/pgwaldump>

Представление `pgpro_stat_wal_activity` отображает объем сгенерированных каждым процессом записей WAL.

<https://postgrespro.ru/docs/enterprise/13/monitoring-stats#MONITORING-PGPRO-STAT-WAL-ACTIVITY-VIEW>

Добавлено стороннее расширение `pgBadger`, адаптированное для корректной работы с Postgres Pro.

Добавлен `Mamonsu` — агент мониторинга Postgres Pro для Zabbix:

<https://postgrespro.ru/docs/enterprise/13/mamonsu>

В `pgbench` добавлен корректный учет ошибок сериализации и взаимоблокировок.

<https://postgrespro.ru/docs/enterprise/13/pgbench>

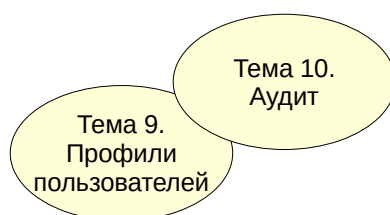
Профили и аудит, сертификация

passwordcheck — запрет простых паролей

Профили пользователя

парольные политики и ограничения

pg_proaudit — аудит действий пользователей



Расширение passwordcheck запрещает использование простых паролей.

<https://postgrespro.ru/docs/enterprise/13/passwordcheck>

Профили пользователей ограничивают использование системы, в частности, реализуют парольные политики.

<https://postgrespro.ru/docs/enterprise/13/sql-createprofile>

Расширение pg_proaudit позволяет проводить аудит действий пользователей.

<https://postgrespro.ru/docs/enterprise/13/pg-proaudit>

Очистка удаленных данных

на диске и в оперативной памяти

Контроль целостности файлов

Работа с персональными данными и конфиденциальной информацией требует использования сертифицированной редакции Postgres Pro Enterprise Certified. Для этой редакции сделано несколько изменений, которые присутствуют и в несертифицированной Postgres Pro Enterprise, но по умолчанию выключены.

Чтобы не допустить возможности восстановления удаленных данных, освобождаемые области памяти автоматически заполняются нулевыми байтами — как на диске, так и в оперативной памяти.

<https://postgrespro.ru/docs/enterprise/13/memory-purge-cert>

Утилита `pg_integrity_check` проверяет контрольные суммы отслеживаемых файлов: исполняемых, конфигурационных, таблиц системного каталога.

<https://postgrespro.ru/docs/enterprise/13/integrity-checks>

`pgpro_scheduler` — встроенный планировщик заданий

разовые и планируемые задания

позволяет обойтись без средств операционной системы

Тема 11. Планировщик
заданий

Расширение `pgpro_scheduler` — интерпретированный в Postgres Pro Enterprise планировщик заданий. С его помощью можно задавать сложные расписания, динамически вычислять время следующего запуска для повторяющихся заданий, выполнять SQL-команды задания в одной или в нескольких последовательных транзакциях, назначать задания для немедленного или отложенного однократного выполнения одновременно с обычными планируемыми заданиями.

<https://postgrespro.ru/docs/enterprise/13/pgpro-scheduler>

pg_probackup — система резервного копирования и восстановления

- централизованный каталог копий
- политики хранения
- инкрементальное копирование
- многопоточность

ptrack

механизм для получения измененных страниц данных

Темы 12, 13 и 14.
Резервное копирование

pg_probackup — система резервного копирования и восстановления кластеров баз данных PostgreSQL и Postgres Pro.

Утилита поддерживает каталог резервных копий, позволяя настраивать политики хранения.

Реализованы различные виды инкрементального копирования, многопоточность и другие возможности.

Механизм PTRACK позволяет в определенных случаях значительно ускорить инкрементальное копирование, сообщая утилите номера блоков данных, измененных после предыдущего копирования.

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

Доработки репликации и кластер multimaster

Смягченная синхронная репликация

ограничение отставания реплики

`synchronous_standby_gap = 0`

Автоматическое исправление поврежденных страниц

страницы запрашиваются с резервного сервера

`page_repair = off`

Восстановление поврежденных записей WAL

walsender может прочитать запись из буфера в оперативной памяти

`wal_sender_check_crc = off`

Смягченная синхронная репликация позволяет временно перевести синхронную реплику в асинхронный режим при значительном отставании.

<https://postgrespro.ru/docs/enterprise/13/runtime-config-replication#GUC-SYNCHRONOUS-STANDBY-GAP>

При чтении страницы с неверной контрольной суммой основной сервер может запросить эту страницу с резервного сервера и автоматически восстановить ее.

<https://postgrespro.ru/docs/enterprise/13/warm-standby#REPAIR-PAGE-FROM-STANDBY>

Процесс walsender может восстановить поврежденную запись WAL из буферов оперативной памяти.

<https://postgrespro.ru/docs/enterprise/13/wal-restoration>

multimaster — синхронный кластер без разделения ресурсов

все узлы ведущие

распределенные транзакции

отказоустойчивость и высокая доступность

масштабирование читающей нагрузки

Тема 15.
Синхронный кластер
multimaster

Расширение multimaster позволяет настроить синхронный кластер, в котором любой узел может принимать запросы как на чтение, так и на запись.

Встроенный менеджер глобальных транзакций позволяет работать с кластером как с монолитной базой, гарантируя обычные свойства ACID. Протокол фиксации на основе кворума обеспечивает отказоустойчивость и высокую доступность кластера.

Обеспечивается масштабируемость читающих транзакций.

<https://postgrespro.ru/docs/enterprise/13/multimaster>

Postgres Pro Enterprise содержит разнообразные дополнительные возможности

- изменения в коде ядра PostgreSQL

- дополнительные расширения

- улучшения сторонних расширений

Основные возможности детально рассматриваются в курсе