

Предисловие

Эту небольшую книгу мы написали для тех, кто только начинает знакомиться с PostgreSQL. Из нее вы узнаете:

I	Что вообще такое этот PostgreSQL	3
II	Что нового появилось в версии PostgreSQL 15	17
III	Как установить PostgreSQL на Linux и Windows ...	25
IV	Как подключиться к серверу, начать писать SQL-запросы, и зачем нужны транзакции	35
V	Как продолжить самостоятельное изучение языка SQL с помощью демобазы	61
VI	Как использовать PostgreSQL в качестве базы данных для вашего приложения	89
VII	Без каких минимальных настроек сервера не обойтись, в том числе при работе с 1C	103
VIII	Про полезную программу pgAdmin	111
IX	Про дополнительные возможности: полнотекстовый поиск,	117
	формат JSON,	125
	доступ к внешним данным	137
X	Какие есть образовательные ресурсы, как стать сертифицированным специалистом ...	149
XI	Как быть в курсе происходящего	171
XII	И немного про компанию Postgres Professional ..	175

Мы надеемся, что наша книга сделает ваш первый опыт работы с PostgreSQL приятным и поможет влиться в сообщество пользователей этой СУБД. Желаем удачи!

I O PostgreSQL

PostgreSQL – наиболее полнофункциональная свободно распространяемая СУБД с открытым кодом. Разработанная в академической среде, за долгую историю сплотившая вокруг себя широкое сообщество разработчиков, эта СУБД обладает всеми возможностями, необходимыми большинству заказчиков. PostgreSQL активно применяется по всему миру для создания критичных бизнес-систем, работающих под большой нагрузкой.

Немного истории

Современный PostgreSQL ведет происхождение от проекта POSTGRES, который разрабатывался под руководством Майкла Стоунбрейкера (Michael Stonebraker), профессора Калифорнийского университета в Беркли. До этого Стоунбрейкер возглавлял разработку INGRES – одной из первых реляционных СУБД, – и POSTGRES возник как результат осмысления предыдущей работы и желания преодолеть ограниченность жесткой системы типов.

Работа над проектом началась в 1985 году, и до 1988 года был опубликован ряд научных статей, описывающих модель данных, язык запросов POSTQUEL (в то время SQL еще не был общепризнанным стандартом) и устройство хранилища данных.

4 POSTGRES иногда относят к так называемым постреляци-
i онным СУБД. Ограниченность реляционной модели всегда
была предметом критики, хотя и являлась обратной сто-
роной ее простоты и строгости. Однако проникновение
компьютерных технологий во все сферы жизни привело
к появлению новых классов приложений и потребовало от
баз данных поддержки нестандартных типов данных и та-
ких возможностей, как наследование, создание сложных
объектов и управление ими.

Первая версия СУБД была выпущена в 1989 году. База дан-
ных совершенствовалась на протяжении нескольких лет,
а в 1993 году, когда вышла версия 4.2, проект был закрыт.
Но, несмотря на официальное прекращение, открытый код
и BSD-лицензия позволили выпускникам Беркли Эндрю Ю
и Джоли Чену в 1994 году взяться за его дальнейшее раз-
витие. Они заменили язык запросов POSTQUEL на ставший
к тому времени общепринятым SQL, а проект нарекли Post-
gres95.

К 1996 году стало ясно, что название Postgres95 не выдер-
жит испытания временем, и было выбрано новое имя –
PostgreSQL, которое отражает связь и с оригинальным про-
ектом POSTGRES, и с переходом на SQL. Надо признать, что
название получилось сложновыговариваемым, но тем не
менее: PostgreSQL следует произносить как «постгрес-ку-
эль» или просто «постгрес», но только не «постгре».

Новая версия стартовала как 6.0, продолжая исходную ну-
мерацию. Проект вырос, и управление им взяла на себя
поначалу небольшая группа инициативных пользователей
и разработчиков, которая получила название Глобальной
группы разработки PostgreSQL (PostgreSQL Global Develop-
ment Group).

Все основные решения о планах развития и выпусках новых версий принимаются Управляющим комитетом (Core team), состоящим сейчас из семи человек.

Помимо обычных разработчиков, вносящих посильную лепту в развитие системы, выделяется группа основных разработчиков (major contributors), сделавших существенный вклад в развитие PostgreSQL, а также группа разработчиков, имеющих право записи в репозиторий исходного кода (committers). Состав групп со временем меняется, появляются новые члены, кто-то отходит от проекта. Актуальный список разработчиков публикуется на официальном сайте: postgresql.org/community/contributors.

Вклад российских разработчиков в PostgreSQL весьма значителен. Это, пожалуй, крупнейший глобальный проект с открытым исходным кодом из всех, в которых настолько широко представлена Россия.

Большую роль в становлении и развитии PostgreSQL сыграл программист из Красноярска Вадим Михеев, входивший в Управляющий комитет. Он является автором таких важнейших частей системы, как многоверсионное управление одновременным доступом (MVCC), система очистки (vacuum), журнал транзакций (WAL), вложенные запросы, триггеры. Сейчас Вадим уже не занимается проектом.

В 2015 году Олег Бартунов, астроном и научный сотрудник ГАИШ МГУ, совместно с Федором Сигаевым и Александром Коротковым основал компанию Postgres Professional как кузницу квалифицированных кадров в области разработки систем баз данных и место создания отечественной СУБД.

- 6 Среди направлений выполненных ими работ можно выделить
i локализацию PostgreSQL (поддержка национальных кодировок и Unicode), систему полнотекстового поиска, работу с массивами и слабоструктурированными данными (hstore, json, jsonb), новые методы индексации (GiST, SP-GiST, GIN и RUM, Bloom). Также они являются авторами большого числа популярных расширений.

Цикл работы над очередной версией PostgreSQL обычно занимает около года. За это время от всех желающих принимаются на рассмотрение патчи с исправлениями, изменениями и новым функционалом. Для обсуждения патчей по традиции используется список рассылки `pgsql-hackers`. Если сообщество признает идею полезной, ее реализацию – правильной, а код проходит обязательную проверку другими разработчиками, то патч включается в релиз.

В некоторый момент (обычно весной, примерно за полгода до релиза) объявляется этап стабилизации кода – новый функционал откладывается до следующей версии, а продолжают приниматься только исправления или улучшения уже включенных в релиз патчей. Несколько раз в течение релизного цикла выпускаются бета-версии, ближе к концу цикла появляется релиз-кандидат, а вскоре выходит и новая основная (major) версия PostgreSQL.

Раньше номер основной версии состоял из двух чисел, но начиная с 2017 года было решено оставить только одно. Таким образом, за 9.6 последовала 10, а последней актуальной версией PostgreSQL является версия 15, вышедшая в середине октября 2022 года.

При работе над новой версией СУБД могут обнаруживаться ошибки. Наиболее критические из них исправляются не только в текущей, но и в предыдущих версиях. Обычно раз в квартал выпускаются дополнительные (minor) версии,

включающие накопленные исправления. Например, версия 12.5 содержит только исправления ошибок, найденных в 12.4, а 15.1 – в версии 15.0.

7
i

Поддержка

Глобальная группа разработки PostgreSQL выполняет поддержку основных версий системы в течение пяти лет с момента выпуска. Эта поддержка, как и координация разработки, осуществляется через списки рассылки. Корректно оформленное сообщение об ошибке имеет все шансы на скорейшее решение: нередки случаи, когда исправления ошибок выпускаются в течение суток.

Помимо поддержки сообществом разработчиков, ряд компаний по всему миру осуществляет коммерческую поддержку PostgreSQL. В России такой компанией является Postgres Professional (postgrespro.ru), предоставляющая услуги поддержки в режиме 24x7.

Современное состояние

PostgreSQL – одна из самых популярных в мире систем баз данных. По итогам свыше двадцати лет развития на прочном академическом фундаменте она выросла в полноценную СУБД, пригодную для корпоративного использования, и составляет реальную альтернативу коммерческим системам. Чтобы убедиться в этом, достаточно посмотреть на важнейшие характеристики новейшей на сегодняшний день версии PostgreSQL 15.

8 Надежность и устойчивость

i

При работе с критически важными данными в корпоративных приложениях особенно важно обеспечить надежность. С этой целью PostgreSQL позволяет настраивать горячее резервирование, восстановление на заданный момент времени в прошлом, различные виды репликации (синхронную, асинхронную, каскадную).

Безопасность

PostgreSQL позволяет пользователям подключаться по защищенному SSL-соединению. Возможна аутентификация по паролю (включая SCRAM), использование клиентских сертификатов и аутентификация с помощью внешних сервисов (LDAP, RADIUS, PAM, Kerberos).

Для управления доступом к объектам баз данных предоставляются следующие возможности:

- создание и управление учетными записями пользователей и групповыми ролями;
- разграничение доступа к объектам БД на уровне как отдельных пользователей, так и групп;
- детальное управление доступом на уровне отдельных столбцов и строк;
- поддержка SELinux через встроенную функциональность SE-PostgreSQL (мандатное управление доступом).

Специальная версия PostgreSQL компании Postgres Professional, Postgres Pro Certified, сертифицирована ФСТЭК для использования в системах обработки конфиденциальной информации и персональных данных.

PostgreSQL обеспечивает соответствие новым требованиям стандарта ANSI SQL по мере их появления. Это относится ко всем версиям стандарта от SQL-92 до самой последней SQL:2016, стандартизовавшей поддержку работы с форматом JSON. Существенная часть этого функционала уже реализована в PostgreSQL 15.

В целом PostgreSQL обеспечивает высокий уровень соответствия стандарту и поддерживает 170 из 177 обязательных возможностей, а также большое количество необязательных.

Поддержка транзакционности

PostgreSQL обеспечивает полную поддержку свойств ACID и обеспечивает эффективную изоляцию транзакций. Для этого используется механизм многоверсионного управления одновременным доступом (MVCC), который позволяет обходиться без блокировок строк во всех случаях, кроме одновременного изменения одной и той же строки данных в нескольких процессах: чтение никогда не блокирует запись, а запись — чтение.

Это же относится и к самому строгому уровню изоляции serializable, который, используя инновационную систему Serializable Snapshot Isolation, обеспечивает полное отсутствие аномалий сериализации и гарантирует совпадение результатов параллельного и последовательного выполнения.

Для разработчиков приложений

Разработчики приложений получают в свое распоряжение богатый инструментарий, позволяющий реализовать приложения любого типа:

- всевозможные языки серверного программирования: встроенный PL/pgSQL (удобный своей тесной интеграцией с SQL), C для критичных по производительности задач, Perl, Python, Tcl, а также JavaScript, Java и другие;
- программные интерфейсы для обращения к СУБД из приложений на любом языке, включая стандартные интерфейсы ODBC и JDBC;
- набор объектов баз данных, позволяющий эффективно реализовать логику любой сложности на стороне сервера: таблицы и индексы, последовательности, ограничения целостности, представления и материализованные представления, секционирование, подзапросы и with-запросы (в том числе рекурсивные), агрегатные и оконные функции, хранимые функции, триггеры и т. д.;
- гибкая система полнотекстового поиска с поддержкой русского и всех европейских языков, дополненная эффективным индексным доступом;
- слабоструктурированные данные, характерные для NoSQL: hstore (хранилище пар «ключ-значение»), XML, json (как в текстовом, так и в более эффективном двоичном представлении jsonb);
- подключение источников данных, включая все основные СУБД, в качестве внешних таблиц по стандарту SQL/MED с возможностью их полноценного использования, в том числе для записи и распределенного выполнения запросов (Foreign Data Wrappers).

PostgreSQL эффективно использует современную архитектуру многоядерных процессоров — производительность СУБД растет практически линейно с увеличением количества ядер.

Предусмотрен параллельный режим выполнения запросов и некоторых служебных команд (таких как создание индексов и очистка). В таком режиме операции чтения данных и соединения выполняются несколькими одновременно работающими процессами. JIT-компиляция запросов повышает возможности ускорения операций аппаратными средствами. Новые возможности распараллеливания появляются в каждой новой версии PostgreSQL.

Для горизонтального масштабирования PostgreSQL предоставляет возможности репликации, как физической, так и логической. Это позволяет строить на базе PostgreSQL кластеры для обеспечения отказоустойчивости, высокой производительности и географической распределенности. Примерами таких систем могут служить Citus (Citusdata), Postgres-BDR (2ndQuadrant), Multimaster (Postgres Professional), Patroni (Zalando).

Планировщик запросов

В PostgreSQL используется стоимостной планировщик запросов. Сбор статистических данных и учет ресурсоемкости как дисковых, так и процессорных операций позволяет оптимизировать даже самые сложные запросы. В распоряжении планировщика находятся все методы доступа к данным и способы выполнения соединений, имеющиеся у передовых коммерческих СУБД.

Возможности индексирования

В PostgreSQL реализованы различные способы индексирования. Помимо традиционных B-деревьев, имеется ряд других методов доступа.

- Hash – индекс, основанный на хешировании. В отличие от B-деревьев, он работает только при проверке на равенство, но в ряде случаев оказывается компактнее и эффективнее.
- GiST – обобщенное сбалансированное дерево поиска, которое применяется для данных, не допускающих упорядочения. Примерами могут служить R-деревья для индексирования точек на плоскости с возможностью быстрого поиска ближайших соседей (k-NN search) и индексирование операции пересечения интервалов.
- SP-GiST – обобщенное несбалансированное дерево, основанное на разбиении области значений на непересекающиеся вложенные области. Примерами могут служить дерево квадрантов для пространственных данных и префиксное дерево для текстовых строк.
- GIN – обобщенный инвертированный индекс, который используется для сложных значений, состоящих из элементов. Основная область его применения – полнотекстовый поиск, то есть поиск документов, в которых встречаются указанные в поисковом запросе слова. Другим примером использования является поиск значений в массивах данных.
- RUM – дальнейшее развитие метода GIN для полнотекстового поиска. Этот индекс, доступный в виде расширения, ускоряет фразовый поиск и сортирует выдачу по релевантности без дополнительных вычислений.

- BRIN – компактная структура, позволяющая найти компромисс между размером индекса и скоростью поиска. Такой индекс эффективен на больших кластеризованных таблицах.
- Bloom – индекс, основанный на фильтре Блума. Благодаря очень компактному представлению позволяет быстро отсеять заведомо ненужные строки, но требует перепроверки оставшихся.

Многие типы индексов могут создаваться не только по одному, но и по нескольким столбцам таблицы. Независимо от типа можно строить индексы как по столбцам, так и по произвольным выражениям, а также создавать частичные индексы только для определенных строк. Покрывающие индексы позволяют ускорить выполнение запросов за счет того, что все необходимые данные извлекаются из самого индекса без обращения к таблице.

В арсенале планировщика имеется сканирование по битовой карте, которое позволяет объединять сразу несколько индексов для ускорения доступа.

Кроссплатформенность

PostgreSQL работает на операционных системах семейства Unix, включая серверные и клиентские разновидности Linux, FreeBSD, Solaris и macOS, а также на Windows.

За счет открытого и переносимого кода на языке C PostgreSQL можно собирать на самых разных платформах, даже если для них отсутствует поддерживаемая сообществом сборка.

Расширяемость — одно из фундаментальных преимуществ системы, лежащее в основе архитектуры PostgreSQL. Пользователи могут самостоятельно, не меняя базовый код системы, добавлять:

- типы данных;
- функции и операторы для работы с новыми типами;
- индексные и табличные методы доступа;
- языки серверного программирования;
- подключения к внешним источникам данных (Foreign Data Wrappers);
- загружаемые расширения.

Полноценная поддержка расширений позволяет реализовать функционал любой сложности не внося изменений в ядро PostgreSQL и допуская подключение по мере необходимости. Например, именно в виде расширений построены такие сложные системы, как:

- CitusDB — возможность распределения данных по разным экземплярам PostgreSQL (шардинг) и массивно-параллельного выполнения запросов;
- PostGIS — одна из наиболее известных и мощных систем обработки геоинформационных данных;
- TimescaleDB — работа с временными рядами, включая специальное секционирование и шардирование.

Только стандартный комплект, входящий в сборку PostgreSQL 15, содержит около полусотни расширений, доказавших свою надежность и полезность.

Доступность

15
i

Либеральная лицензия PostgreSQL, сходная с лицензиями BSD и MIT, разрешает неограниченное использование СУБД, модификацию кода, а также включение в состав других продуктов, в том числе закрытых и коммерческих.

Независимость

PostgreSQL не принадлежит ни одной компании и развивается международным сообществом, в том числе и российскими разработчиками. Это означает, что системы, использующие PostgreSQL, не зависят от какого-либо конкретного производителя, благодаря чему вложенные в них средства сохранятся в любой ситуации.