

POSTGRESQL

Анатолий ГАРБУЗОВ

Департамент Программной Инженерии и Автоматики, группа TI-196, Факультет Вычислительной Техники, Информатики и Микроэлектроники, Технический Университет Молдовы, Кишинев, Республика Молдова

Автор корреспондент: Анатолий ГАРБУЗОВ, e-mail: garbuzov.anatolie@isa.utm.md

Научный руководитель: Дориан САРАНЧУК, DISA, FCIM, UTM

Аннотация: В данной статье рассматривается СУБД PostgreSQL. Описываются структуры и типы данных, модель данных, масштабируемость, расширения и преимущества данной СУБД.

Ключевые слова: объектно-реляционная база данных, PostgreSQL, JSON, Postgre95

Введение

PostgreSQL создана на основе некоммерческой СУБД Postgres, разработанной как open-source проект в Калифорнийском Университете Беркли. Главным разработчиком данной СУБД является Майкл Стоунбрейкер. Начало разработки СУБД пришлось на 1986 год. PostgreSQL разрабатывалась Майклом Стоунбрейкером и его студентами долгие 8 лет, начиная с 1986 года и заканчивая 1994 годом.



Рисунок 1. Объектно-реляционная система управления базами данных, основанная на языке SQL [1]

За весь период разработки синтаксис СУБД пополнился различными процедурами, правилами и другими различными компонентами. В 95-ом году разработка СУБД снова разделилась. Главный разработчик, а именно Майкл Стоунбрейкер, использовал полученный опыт в создании коммерческой СУБД Illustra, а его студенты продолжили заниматься PostgreSQL, в следствии чего создали новую версию СУБД Postgre, получившую название — Postgres95, где POSTQUEL – язык запросов, изменили на SQL. Далее разработку Postgres95 передали одной команде энтузиастов, которые не имели отношение к Университету Беркли [2].

После чего название Postgre95 перестало существовать и ему на смену пришло новое, существующее и по сей день — PostgreSQL.

Особенности PostgreSQL

Надежность. Помимо безопасного хранения данных и предоставления пользователю возможности извлекать данные при обработке запроса, он поддерживается сообществом участников, которые регулярно находят ошибки и пытаются улучшить программное обеспечение, делая PostgreSQL надежным [2].

Производительность. Операции записи в PostgreSQL могут выполняться одновременно без необходимости блокировки чтения/записи. Индексы используются для ускорения запросов при работе с большими объемами данных, что позволяет базам данных находить определенную строку без необходимости циклического просмотра всех данных [2].

В PostgreSQL вы даже можете создать индекс выражения, который работает с результатом выражения или функции, а не только со значением столбца. Также поддерживается частичная индексация, при которой индексируется только часть таблицы. Он также поддерживает распараллеливание запросов на чтение, JIT-компиляцию выражений и вложенные транзакции (через точки сохранения), что обеспечивает высокую производительность и эффективность.

Соответствие стандартам SQL

Еще одна область, в которой PostgreSQL выделяется среди других систем реляционных баз данных, — это ее соответствие стандартам SQL. Стандарты SQL были разработаны группами ANSI и ISO для определения минимальных требований к функциональности и совместимости для реализаций SQL. Хотя спецификации, предоставляемые этими органами, предназначены для определения функций, которые должны предоставлять системы SQL, из-за сложности и длительного развития языка строгое соблюдение не всегда возможно. Согласно документации PostgreSQL, в настоящее время ни одна база данных не удовлетворяет всем требованиям, изложенным в спецификации. Учитывая это предостережение, PostgreSQL удовлетворяет большему количеству спецификаций SQL, чем другие варианты. По их собственным подсчетам, они соответствуют как минимум 160 из 179 основных требований, изложенных в SQL:2016.

Расширения. PostgreSQL может похвастаться несколькими надежными наборами функций, включая восстановление на определенный момент времени, управление параллельным доступом к нескольким версиям (MVCC), табличные пространства, детализированный контроль доступа, асинхронную репликацию, усовершенствованный планировщик/оптимизатор запросов и ведение журнала с опережающей записью. Multi-Version Concurrency Control позволяет одновременное чтение и запись таблиц, блокируя только одновременные обновления одной и той же строки. Таким образом удастся избежать столкновений [3].

Масштабируемость. PostgreSQL поддерживает Unicode, международные наборы символов, многобайтовые кодировки символов и учитывает локаль для сортировки, учета регистра и форматирования. PostgreSQL обладает высокой масштабируемостью — он может работать как с количеством одновременных пользователей, так и с объемом данных, которыми он может управлять. Кроме того, PostgreSQL является кроссплатформенным и может работать во многих операционных системах, включая Linux, Microsoft Windows, OS X, FreeBSD и Solaris [3].

Динамическая загрузка. Сервер PostgreSQL также может включать в себя пользовательский код посредством динамической загрузки. Пользователь может указать файл объектного кода; например, общая библиотека, которая реализует новую функцию или тип, и PostgreSQL загрузит ее по мере необходимости. Возможность изменять его работу «на лету» делает его уникально подходящим для быстрой реализации новых структур хранения и приложений.

Преимущества. Поддержка огромного количества типов данных, включая: [4]

- общие примитивы базы данных, такие как числовые, строковые, логические типы и типы даты и времени: они встречаются в большинстве баз данных и отлично поддерживаются в PostgreSQL.

- сетевые адреса: PostgreSQL поддерживает различные типы, связанные с сетью, такие как адреса CIDR, адреса с масками подсети и MAC-адреса, как для IPv4, так и для IPv6.
- геометрические типы: доступно множество типов, которые помогут вам определить двухмерные объекты. К ним относятся точки, линии и сегменты, а также полные фигуры, такие как прямоугольники, многоугольники и круги.
- денежные типы: тип PostgreSQL money хранит валюту с фиксированной дробной точностью. Тип учитывает локаль и включает автоматическое форматирование вывода.
- диапазоны: диапазоны позволяют вам работать с диапазонами значений, например, с диапазонами дат и времени для планирования.
- JSONB: встроенная поддержка PostgreSQL для хранения и работы с объектами JSON может помочь вам работать как с реляционными, так и с нереляционными данными в одной системе.
- hstore: этот hstore тип позволяет вам работать с парами ключ-значение изначально.
- многомерные массивы: массивы очень полезны для хранения значений, состоящих из нескольких частей, которые не имеют большого значения вне своего конкретного контекста.

Возможность определять свои собственные сложные типы. Определение собственных типов помогает вам более точно согласовать вашу базу данных с тем, как ваши данные представлены в ваших приложениях. 4

Полнотекстовый поиск. Полнотекстовый поиск предоставляет мощные методы поиска и работы с данными в полуструктурированном и неструктурированном тексте. Поиск можно настроить так, чтобы он соответствовал вашим ожиданиям в отношении релевантности и соответствия.

Надежные системы аутентификации, контроля доступа и управления привилегиями, подходящие для организаций любого размера: PostgreSQL имеет зрелые функции аутентификации и авторизации пользователей, позволяющие определять, кто может использовать систему и что каждому пользователю разрешено видеть или делать.

Оболочки внешних данных: Обертки внешних данных позволяют представлять и получать доступ к таблицам и данным на удаленных серверах. Представления и материализованные представления.

Поддержка представлений и материализованных представлений обеспечивает удобный и упрощенный доступ к данным за счет абстрагирования исходных структур таблиц для информации, которая часто запрашивается вместе.

Комментарии к объектам базы данных. Возможность прикреплять комментарии к таблицам, базам данных, столбцам и другим отдельным объектам базы данных позволяет документировать решения или детали реализации.

Ведение журнала с опережающей записью для обеспечения восстановления на определенный момент времени, отработки отказа и потоковой репликации. Эти технологии помогают гарантировать, что ваша база данных остается согласованной даже в случае сбоя программного обеспечения, а также помогают копировать данные между системами для масштабирования и резервного копирования данных.

Поддержка поведения, подобного NoSQL, такого как хранение документов с использованием JSONB и пар ключ-значение с помощью hstore. Возможность использования нескольких парадигм базы данных в одной системе может помочь минимизировать ваши административные издержки и улучшить совместимость между различными представлениями.

Заключение

PostgreSQL — это бесплатная система управления реляционными базами данных с открытым исходным кодом, ориентированная на совместимость и расширяемость SQL. Благодаря более чем 30-летнему активному развитию PostgreSQL является одним из наиболее широко используемых инструментов баз данных с открытым исходным кодом во всем мире.

В этой статье мы рассмотрели некоторые важные особенности PostgreSQL, архитектуру PostgreSQL, варианты ее использования, преимущества, операционные проблемы и ключевые альтернативы. Мы завершили его несколькими рекомендуемыми практиками, чтобы поддерживать вашу базу данных WordPress в оптимальном состоянии, пока вы продолжаете масштабирование.

Библиография

1. PostgreSQL — объектно-реляционная система управления базами данных [Электронный ресурс]. [дата обращения 02.03.2023], Доступно:– <https://web-creator.ru/articles/postgresql>
2. PostgreSQL Википедия[Электронный ресурс]. [дата обращения 02.03.2023], Доступно:–<https://ru.wikipedia.org/wiki/PostgreSQL>
3. Чем PostgreSQL лучше других SQL баз данных с открытым исходным кодом. [Электронный ресурс]. [дата обращения 02.03.2023], Доступно:– <https://habr.com/ru/post/282764/>
4. Основные преимущества PostgreSQL для разработчиков [Электронный ресурс]. [дата обращения 02.03.2023], Доступно:– <https://www.cischool.ru/osnovnye-preimushhestva-postgresql-dlya-razrabotchikov/>