

SQLite – КОМПАКТНАЯ ВСТРАИВАЕМАЯ СИСТЕМА УПРАВЛЕНИЯ БАЗАМИ ДАННЫХ

Светлана СОЛТЫК

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

Автор корреспонденции: Светлана СОЛТЫК, e-mail: soltic.svetlana@isa.utm.md

Научный руководитель: Дориан САРАНЧУК, DISA, FCIM, UTM

Аннотация: Данная работа посвящена СУБД SQLite. SQLite - это программное обеспечение, которое позволяет пользователям взаимодействовать с реляционной базой данных. В SQLite база данных хранится в одном файле — черта, которая отличает ее от других движков баз данных. Этот факт обеспечивает большую доступность: копирование базы данных не сложнее, чем копирование файла, в котором хранятся данные, совместное использование базы данных может означать отправку вложения электронной почты. В статье рассказывается о преимуществах, особенностях и принципе работы. Так же в статье будет рассмотрено использование SQLite, и сделан вывод на основе рассмотренного материала.

Ключевые слова: SQLite, СУБД, SQL, библиотека, таблица, один файл.

1. Введение

SQLite - это встроенный механизм базы данных SQL. Отличается от многих других баз данных SQL тем, что SQLite не имеет отдельного серверного процесса. SQLite считывает и записывает непосредственно в обычные дисковые файлы. Полная база данных SQL с несколькими таблицами, индексами, триггерами и представлениями содержится в одном файле на диске. Формат файла базы данных является кроссплатформенным - вы можете свободно копировать базу данных между 32-разрядными и 64-разрядными системами или между архитектурой big-end и little-end. Эти функции делают SQLite популярным выбором в качестве формата файла приложения. Файлы базы данных SQLite являются рекомендуемым форматом хранения Библиотекой Конгресса США. Следует думать о SQLite не как о замене Oracle, а как о замене fopen().

SQLite - это компактная библиотека. При включенных всех функциях размер библиотеки может составлять менее 750 КБ, в зависимости от целевой платформы и настроек оптимизации компилятора. Существует компромисс между использованием памяти и скоростью. SQLite обычно работает быстрее, чем больше памяти вы ему предоставляете. Тем не менее, производительность обычно довольно высока даже в средах с низким объемом памяти. Зависит от использования SQLite, SQLite может быть быстрее, чем прямой ввод-вывод файловой системы [1].

SQLite очень внимательно тщательно тестируется перед каждым выпуском и имеет репутацию очень надежного механизма. Большая часть исходного кода SQLite посвящена исключительно тестированию и проверке. Автоматизированный набор тестов запускает миллионы и миллионы тестовых примеров, включающих сотни миллионов отдельных операторов SQL, и обеспечивает 100%-ный охват тестированием отрасли. SQLite изящно реагирует на сбои в распределении памяти и ошибки дискового ввода-вывода. Транзакции выполняются автоматически, даже если они прерваны системными сбоями или сбоями питания. Все это проверяется с помощью автоматизированных тестов с использованием специальных тестовых жгутов, которые имитируют системные сбои. Конечно, даже при всем этом тестировании все еще есть ошибки. Но в отличие от некоторых подобных проектов (особенно коммерческих конкурентов) SQLite открыт и честен в отношении всех ошибок и предоставляет списки ошибок и поминутную хронологию изменений кода.

База кода SQLite поддерживается международной командой разработчиков, которые работают над SQLite полный рабочий день. Разработчики продолжают расширять возможности SQLite и повышать его надежность и производительность, сохраняя при этом обратную совместимость с опубликованными спецификациями интерфейса, синтаксисом SQL и форматом файла базы данных. Исходный код абсолютно бесплатен для всех, кто этого хочет, но также доступна профессиональная поддержка.

Проект SQLite был начат 9 сентября 2000 года. Будущее всегда трудно предсказать, но разработчики намерены поддерживать SQLite до 2050 года. Проектные решения принимаются с учетом этой цели.

2. Особенности SQLite

SQLite обладает следующими заметными особенностями: *бессерверный, автономный, с нулевой конфигурацией, транзакционный* [2].

Бессерверный

Обычно для работы СУБД, такой как MySQL, PostgreSQL и т.д., требуется отдельный серверный процесс. Приложения, которые хотят получить доступ к серверу базы данных, используют протокол TCP/IP для отправки и получения запросов. Это называется архитектурой клиент/сервер.

Следующая диаграмма, показанная на рисунке 1, иллюстрирует архитектуру клиент/сервер СУБД:

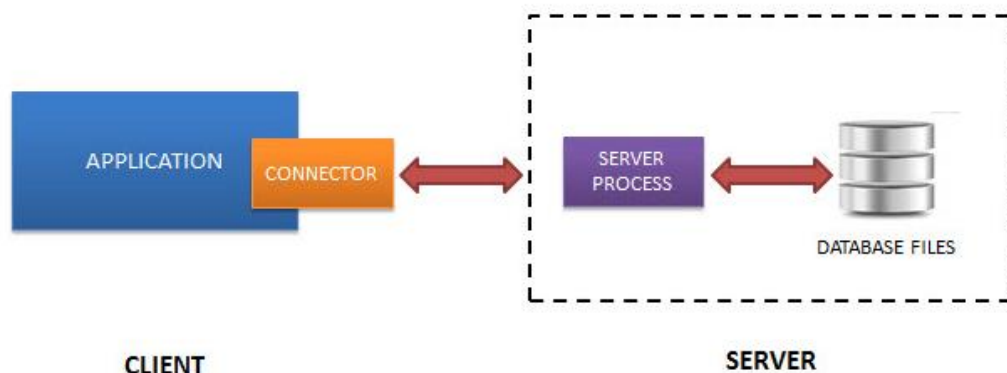


Рисунок 1. Архитектура клиент/сервер СУБД.

Источник: <https://tableplus.com/blog/2018/08/sqlite-vs-mysql-which-database-should-you-use.html>

SQLite не работает таким образом. Для запуска SQLite НЕ требуется сервер.

База данных SQLite интегрирована с приложением, которое обращается к базе данных. Приложения взаимодействуют с базой данных SQLite для чтения и записи непосредственно из файлов базы данных, хранящихся на диске.

Следующая диаграмма, показанная на рисунке 2, иллюстрирует архитектуру без сервера SQLite:



Рисунок 2. Архитектура без сервера SQLite.

Источник: <https://tableplus.com/blog/2018/08/sqlite-vs-mysql-which-database-should-you-use.html>

Автономный

SQLite является автономным, что означает, что он требует минимальной поддержки со стороны операционной системы или внешней библиотеки. Это делает SQLite пригодным для использования в любой среде, особенно во встроенных устройствах, таких как iPhone, телефоны Android, игровые консоли, портативные медиаплееры и т. д.

SQLite разработан с использованием ANSI-C. Исходный код доступен в виде большого файла `sqlite3.c` и его заголовочного файла `sqlite3.h`. Если вы хотите разработать приложение, использующее SQLite, вам просто нужно добавить эти файлы в свой проект и скомпилировать его со своим кодом.

Нулевая конфигурация

Так как SQLite имеет бессерверную архитектуру нет необходимости «устанавливать» SQLite перед его использованием. Нет серверного процесса, который нужно настраивать, запускать и останавливать.

Кроме того, SQLite не использует никаких файлов конфигурации.

Транзакционный

Все транзакции в SQLite полностью совместимы с ACID. Это значит, что запросы и изменения являются последовательными, атомарными, изолированными и надежными.

Иными словами, внутри транзакции все изменения происходят полностью или не происходят вообще, и даже если возникает непредвиденная ситуация, такая как сбой приложения, сбой питания или сбой операционной системы.

3. Недостатки SQLite

Переносимость сигнатур SQLite, к сожалению, делает его плохим выбором, когда множество разных пользователей обновляют таблицу одновременно (для сохранения целостности данных только один пользователь может одновременно записывать в файл). Также может потребоваться дополнительная работа по обеспечению безопасности личных данных благодаря тем же функциям, которые делают SQLite доступным. Кроме того, SQLite не проверяет типы данных. Там, где многие другие программы для баз данных отвергают данные, которые не соответствуют схеме таблицы, SQLite позволяет пользователям хранить данные любого типа в любом столбце [3].

SQLite создает схемы, которые ограничивают тип данных в каждом столбце, но не обеспечивают их соблюдение. В приведенном ниже примере на рисунке 3 показано, что в столбце `id` предполагается хранить целые числа, в столбце `имя` предполагается хранить текст, а в столбце `возраст` предполагается хранить целые числа [4]:

```
CREATE TABLE celebs (  
  id INTEGER,  
  name TEXT,  
  age INTEGER  
);
```

Рисунок 3. Создание таблицы и переменных

Однако SQLite не будет отклонять значения неправильного типа. Мы могли случайно вставить неправильные типы данных в столбцы. Хранение разных типов данных в одном столбце - плохая привычка, которая может привести к ошибкам, которые трудно исправить, поэтому важно строго относиться к вашей схеме, даже если SQLite не будет ее применять.

4. Использование SQLite

Даже учитывая недостатки, преимущества возможности доступа к базе данных и управления ею без использования серверного приложения огромны. SQLite используется во всем мире для тестирования, разработки и в любых других сценариях, когда имеет смысл, чтобы база данных находилась на том же диске, что и код приложения. Разработчики SQLite считают его одним из наиболее тиражируемых программных продуктов в мире.

Поскольку SQLite широко используется в каждом смартфоне, и активно используется более 4,0 миллиарда (4,0e9) смартфонов, каждый из которых содержит сотни файлов базы данных SQLite, представляется вероятным, что в активном использовании находится более одного триллиона (1e12) баз данных SQLite.

Заключение

В заключение, следует указать важный плюс в использовании SQLite.

SQLite использует динамические типы для таблиц. Это означает, что вы можете хранить любое значение в любом столбце, независимо от типа данных.

SQLite позволяет использовать одно подключение к базе данных для одновременного доступа к нескольким файлам базы данных. Это обеспечивает множество приятных функций, таких как объединение таблиц в разных базах данных или копирование данных между базами данных одной командой.

SQLite способен создавать базы данных в памяти, с которыми очень быстро работать.

Библиотека SQLite написана на C. существует большое количество привязок на других языках программирования, включая Apple Swift, Delphi, C++, Java, C#, VB.NET, Python, Perl, узел.js, PHP и многие другие [5].

Простота и удобство интеграции SQLite привели к тому, что библиотека используется в браузерах, музыкальных проигрывателях и многих других программах.

SQLite используется в:

- Adobe Integrated Runtime — среда для запуска приложений (частично);
- Gears;
- Autoit;
- Фреймворк Qt;
- Платформа XUL на движке Gecko 1.9+, XULRunner 1.9+;
- Mozilla Firefox (начиная с версии 3.0);
- Mozilla Thunderbird (начиная с версии 3.0);
- Songbird;
- SQLite Manager;
- Skype;
- Viber;
- Некоторые модели GPS-навигаторов Garmin.

Библиография

1. SQLite homepage. What is SQLite? - [online]. [21.12.2022] – Доступно: <https://www.sqlite.org/index.html>
2. SQLite. SQLite features. - [online]. [22.12.2022] – Доступно: <https://www.sqlite.org/features.html>
3. Codecademy Team. What is SQLite? - [online]. [04.01.2022] – Доступно: <https://www.codecademy.com/article/what-is-sqlite>
4. SQLite Tutorial. Basic SQLite Tutorial. - [online]. [25.12.2022] – Доступно: <https://www.sqlitetutorial.net/>
5. Sibsankar Haldar. - SQLite Database System Design and Implementation (Second Edition, Version 1) – 2015.