

МОДУЛЬ УПРАВЛЕНИЯ ЗАПРОСАМИ К БАЗЕ ДАННЫХ

Рассматривается реализация модуля управления запросами к базе данных с применением репликации данных

ВВЕДЕНИЕ

Мы живем в мире Интернета, и везде нас окружают соцсети, форумы, интернет-магазины и другие сайты. Естественно, они должны работать круглосуточно и безотказно. Пользователю, который заходит на сайт, что-то смотрит, что-то покупает, главное чтобы все работало быстро и безопасно. Чем больше приложение (сайт), тем больше головной боли. Ведь чем больше приходит запросов, тем дольше пользователю придется ждать ответ, а долго ждать никто не любит. Поэтому одной из главных задач является разгрузка сервера базы данных. Целью проекта является разработка модуля управления запросами к базе данных. Он поможет разгрузить главный сервер базы данных и параллельно решит еще несколько сопутствующих задач. Репликация может решить и проблему резервного копирования. Если речь идет о корпоративной базе данных, то репликация прекрасно решает проблему архивации, когда база разбивается на текущую и архив.

I. ОСНОВНЫЕ ЭЛЕМЕНТЫ МОДУЛЯ

Рассмотрим средства, которые будут нужны для реализации модуля управления запросами к базе данных. Под средствами подразумеваются языки программирования, платформы, а также набор технологий, который позволит решить задачу дипломного проекта. Разработка модуля – достаточно сложный и трудоёмкий процесс, требующий решения сложных нетривиальных задач. Для их решения используются программные продукты, которые разрабатываются разными ИТ-компаниями, в частности продукты MsSql, AngularJS, ASP.NET MVC Framework. Главным элементом модуля управления запросами к базе данных является репликация данных, которая позволит поддерживать актуальные данные на нескольких базах. Реализация репликации достаточно непростой процесс, так как она бывает разных видов. Первый - когда любой из серверов может использоваться как для чтения, так и для записи. Второй - когда один сервер главный, на нем происходит как чтение, так и запись данных, а остальные - зависимые сер-

веры, на которых осуществляться только чтение данных. Также все эти виды репликаций могут работать как в синхронном, так и в асинхронном режимах. Вторая главная часть модуля - это организация распределения запросов. Третья - тестирование. Для этого разрабатывается клиентское тестирующее приложение, которое будет отправлять запросы к нашему модулю, и строить графики, показывающие результаты работы модуля в разных режимах.

II. ПРИНЦИП РАБОТЫ

Под репликацией подразумевается процесс, при котором данные с одного сервера базы данных постоянно копируются (реплицируются) на один или несколько других серверов (называемых репликами). Это дает возможность приложениям использовать не один сервер для обработки всех запросов, а несколько. Таким образом, можно перераспределить нагрузку с одного сервера на несколько. Стоит отметить, что распределение запросов эффективно применять в системах, где преобладает чтение данных, поскольку при чтении данные остаются неизменными. Если же данные изменяются, то нужно какое-то время, чтобы все базы также обновили свое содержимое. Далее главный сервер принимает запросы на запись и чтение, а зависимые – слейвы – обрабатывают запросы только на чтение. Базы при этом будут находиться в актуальном состоянии, далее, применяя модуль управления запросами, мы можем распараллелить запросы между серверами, тем самым равномерно распределяя общую нагрузку.

III. ВЫВОДЫ

Предлагаемый модуль позволяет компаниям затрачивать меньше времени на поиск данных. Модуль будет полезен тем, что он может применяться в системах, где стандартными средствами не реализована репликация данных. Кроме того, модуль позволяет резервировать данные, увеличивая производительность системы, а также использовать отложенные вычисления (тяжелые и медленные SQL-запросы можно выполнять на отдельном слейве, не боясь помешать нормальной работе всей системы).

Желдак Артем Олегович, студент 5-го курса Факультета информационных технологий и управления, artemzheldak1995@gmail.com

Научный руководитель: Герман Юлия Олеговна, ассистент кафедры информационных технологий автоматизированных систем БГУИР, jgerman@bsuir.by .