

ПРОИЗВОДИТЕЛЬНОСТЬ ОБРАБОТКИ ДАННЫХ ТИПА JSON

В.А. Савельев

(Украина, ДВНЗ «Национальный горный университет», Днепр)

Постановка проблемы. Целью работы является изучение зависимости производительности обработки данных типа JSON сервером реляционных баз данных MySQL.

Сервер MySQL, начиная с выпуска 5.7.8, поддерживает тип данных в формате JSON (JavaScript Object Notation). Документы JSON преобразуются в двоичный внутренний формат. Двоичный формат дает возможность поиска подчиненных объектов или вложенных значений непосредственно по ключу или массиву индексов, не читая все значения до или после них в документе [1].

Серверы реляционных баз данных часто используют для оперативной обработки транзакций (online transaction processing, OLTP). Обобщенным параметром производительности работы сервера баз данных является количество транзакций, которые сервер может обработать за единицу времени. При измерении следует учитывать то, что производительность является относительным параметром и может изменяться в зависимости от различных факторов [2].

Сервер реляционных баз данных MySQL является программным продуктом с открытым исходным кодом. Это дает возможность сравнить работу структур данных, которые используют для хранения и индексирования [3].

Результаты исследования. Для измерений производительности был использован компьютер с процессором Intel Core i3-3110M 2,4 GHz, RAM 4 GB, HDD 500 GB. Операционная система Ubuntu 16.04 Server 64 bit, MySQL 5.7.16 из репозитория Ubuntu. Производительность обработки данных типа JSON сравнивалась с производительностью обработки типов данных `int` и `char`. Была создана тестовая база данных с таблицами, содержащими от 10 000 до 50 000 строк. Данные созданы искусственно. Три таблицы реализовывают связь типа «многие ко многим» с использованием внешних ключей. Дополнительные индексы не создавались. Для имитации нагрузки на сервер использовалась программа `mysqlslap`, входящая в состав MySQL. Исследованы нагрузки в диапазоне от 10 до 640 одновременных подключений.

Были получены следующие результаты. Время выполнения инструкций `insert`, `update`, `delete` с данными типа JSON существенно не изменяется. Время выполнения инструкций `select` с данными типа JSON без использования встроенных функций увеличивается в 1,5 раза, чем для данных типов `int` и `char`.

Выводы. В результате проведенных исследований установлено, что использование данных типа JSON может приводить к увеличению времени обработки транзакций в 1,5 раза. Поэтому данные типа JSON следует использовать при необходимости. Для выяснения влияния индексов необходимы дополнительные исследования.

ПЕРЕЧЕНЬ ССЫЛОК

1. <http://dev.mysql.com/doc/refman/5.7/en/>
2. Schwarts B., Zaitsev P., Tkachenko V. High performance MySQL, Thirded. – USA: O’ReillyMedia, 2012. – 826 p.
3. Bell C. ExpertMySQL, Seconded. – USA: Apress, 2012. – 668 p.

УДК 681.3.06 : 534.4:

ИНТЕГРАЦИЯ DELPHI-ПРИЛОЖЕНИЙ С МАТЕМАТИЧЕСКИМИ БИБЛИОТЕКАМИ СРЕДЫ МАТЛАВ ДЛЯ РЕШЕНИЯ ЗАДАЧ ЦИФРОВОЙ ОБРАБОТКИ СИГНАЛОВ

Ю.Д. Шмаков, А.А. Козаченко

(Украина, ДВНЗ «Национальный горный университет», Днепр)

Постановка проблемы. Для разработки приложений с профессиональным графическим интерфейсом широкое применение находят системы визуального программирования, например, Delphi, Visual C++ и др. Имея возможность подключения API-функций, обеспечивающих работу с портами ввода-вывода, эти системы позволяют легко разрабатывать интерфейсы внешних устройств, существенно облегчая задачу построения систем сбора информации. Однако эти системы программирования располагают весьма ограниченным набором процедур математического характера, что усложняет задачу обработки полученной информации. Для сложных математических задач лучше подходят специализированные программные средства - системы компьютерной математики (СКМ), такие Matlab, MathCAD и некоторые другие. Они обладают огромными библиотеками математических алгоритмов, которые при необходимости легко могут быть расширены пользователем, но, в то же время, они имеют ограниченные возможности по созданию графического интерфейса пользователя, ограниченные возможности взаимодействия с разнообразными внешними устройствами, что ограничивает возможность обработки этой информации в реальном масштабе времени. Напрашивается вполне очевидное решение – совместное использование систем визуального программирования с СКМ, позволяющее создавать эффективные системы сбора и обработки информации.

Одним из чрезвычайно мощных инструментов, позволяющий разработчикам создавать приложения с графическим интерфейсом, предназначенные для выполнения математических вычислений является СКМ Matlab. Для нее реализованы и документированы средства организации связи с Matlab для приложений, написанных на языках Fortran, C/C++, Java, C#[1]. Однако для связи с проектами на Delphi разработчиками системы Matlab никаких средств не предусматривается. В то же время, Delphi, в силу простоты и наглядности языка программирования, достаточно часто используется для создания систем сбора и обработки измерительной информации, и их интеграция обеспечивает достаточно очевидные преимущества (рис.1).