

Реализация алгоритма распределенной блокировки на основе БД Cassandra

Денис Мухаметьянов

9 июня 2015

Снятие денег с банковского счета

Узнать текущий баланс

Если баланс больше или равен запрашиваемой сумме:

 Уменьшить баланс на запрашиваемую сумму

 Сообщить об успешном проведении операции

Иначе:

 Сообщить о том, что средств недостаточно

Механизм блокировки

- Предоставление исключительного доступа к разделяемому ресурсу для множества потоков
- Реализован в большинстве современных языков программирования в рамках одного процесса

Механизм блокировки

- Предоставление исключительного доступа к разделяемому ресурсу для множества потоков
- Реализован в большинстве современных языков программирования в рамках одного процесса

Механизм блокировки

- Предоставление исключительного доступа к разделяемому ресурсу для множества потоков
- Реализован в большинстве современных языков программирования в рамках одного процесса

Распределенная система

- Существует множество серверов, на которых запущены компоненты системы
- Механизмы, реализованные в языках программирования, не подходят

Распределенная система

- Существует множество серверов, на которых запущены компоненты системы
- Механизмы, реализованные в языках программирования, не подходят

Распределенная система

- Существует множество серверов, на которых запущены компоненты системы
- Механизмы, реализованные в языках программирования, не подходят

Необходим механизм предоставления исключительного доступа
с разных серверов

Google Chubby

- Подробно описанная технология
- Реализация закрыта

Apache ZooKeeper

- Открытый исходный код
- Нет готового клиента для C#

Нужно собственное решение

Google Chubby

- Подробно описанная технология
- Реализация закрыта

Apache ZooKeeper

- Открытый исходный код
- Нет готового клиента для C#

Нужно собственное решение

Google Chubby

- Подробно описанная технология
- Реализация закрыта

Apache ZooKeeper

- Открытый исходный код
- Нет готового клиента для C#

Нужно собственное решение

Google Chubby

- Подробно описанная технология
- Реализация закрыта

Apache ZooKeeper

- Открытый исходный код
- Нет готового клиента для C#

Нужно собственное решение

Google Chubby

- Подробно описанная технология
- Реализация закрыта

Apache ZooKeeper

- Открытый исходный код
- Нет готового клиента для C#

Нужно собственное решение

Google Chubby

- Подробно описанная технология
- Реализация закрыта

Apache ZooKeeper

- Открытый исходный код
- Нет готового клиента для C#

Нужно собственное решение

Google Chubby

- Подробно описанная технология
- Реализация закрыта

Apache ZooKeeper

- Открытый исходный код
- Нет готового клиента для C#

Нужно собственное решение

Apache Cassandra

- Распределенная и отказоустойчивая
- Разреженная таблица

Apache Cassandra

- Распределенная и отказоустойчивая
- Разреженная таблица

Apache Cassandra

- Распределенная и отказоустойчивая
- Разреженная таблица

Существующий алгоритм

- Использует особенности хранения разреженной таблицы в Cassandra
- Корректно решает поставленную задачу
- Неравномерно распределяет блокировки между потоками

Существующий алгоритм

- Использует особенности хранения разреженной таблицы в Cassandra
- Корректно решает поставленную задачу
- Неравномерно распределяет блокировки между потоками

Существующий алгоритм

- Использует особенности хранения разреженной таблицы в Cassandra
- Корректно решает поставленную задачу
- Неравномерно распределяет блокировки между потоками

Существующий алгоритм

- Использует особенности хранения разреженной таблицы в Cassandra
- Корректно решает поставленную задачу
- Неравномерно распределяет блокировки между потоками

Новый алгоритм

- Использует особенности хранения разреженной таблицы в Cassandra
- Выстраивает потоки в очередь в строке
- Распределяет блокировку между потоками в порядке этой очереди
- Не ухудшил производительность
- Корректно решает поставленную задачу

Новый алгоритм

- Использует особенности хранения разреженной таблицы в Cassandra
- Выстраивает потоки в очередь в строке
- Распределяет блокировку между потоками в порядке этой очереди
- Не ухудшил производительность
- Корректно решает поставленную задачу

Новый алгоритм

- Использует особенности хранения разреженной таблицы в Cassandra
- Выстраивает потоки в очередь в строке
- Распределяет блокировку между потоками в порядке этой очереди
- Не ухудшил производительность
- Корректно решает поставленную задачу

Новый алгоритм

- Использует особенности хранения разреженной таблицы в Cassandra
- Выстраивает потоки в очередь в строке
- Распределяет блокировку между потоками в порядке этой очереди
- Не ухудшил производительность
- Корректно решает поставленную задачу

Новый алгоритм

- Использует особенности хранения разреженной таблицы в Cassandra
- Выстраивает потоки в очередь в строке
- Распределяет блокировку между потоками в порядке этой очереди
- Не ухудшил производительность
- Корректно решает поставленную задачу

Новый алгоритм

- Использует особенности хранения разреженной таблицы в Cassandra
- Выстраивает потоки в очередь в строке
- Распределяет блокировку между потоками в порядке этой очереди
- Не ухудшил производительность
- Корректно решает поставленную задачу

- Алгоритм реализован на C#
- Проведено модульное и нагрузочное тестирование
- Проведено сравнение производительности со старым алгоритмом

- Алгоритм реализован на C#
- Проведено модульное и нагрузочное тестирование
- Проведено сравнение производительности со старым алгоритмом

- Алгоритм реализован на C#
- Проведено модульное и нагрузочное тестирование
- Проведено сравнение производительности со старым алгоритмом

Результаты сравнения производительности

- Среднее время ожидания освобождения блокировки не ухудшилось
- Потоки захватывают блокировки равномерно

Результаты сравнения производительности

- Среднее время ожидания освобождения блокировки не ухудшилось
- Потоки захватывают блокировки равномерно

Результаты сравнения производительности

- Среднее время ожидания освобождения блокировки не ухудшилось
- Потоки захватывают блокировки равномерно

Рис.: Моменты взятия блокировок старым алгоритмом

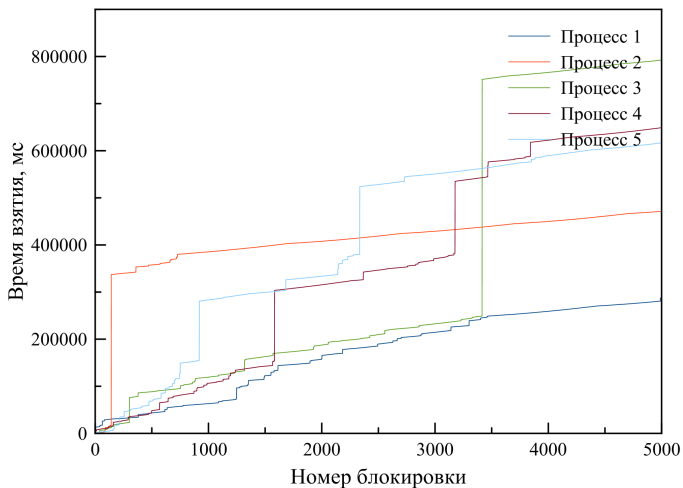
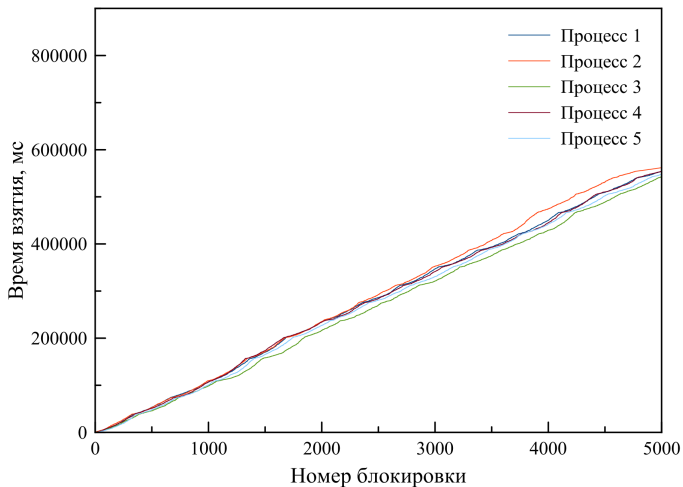


Рис.: Моменты взятия блокировок новым алгоритмом



- Внедрение нового алгоритма в проект
- Снижение среднего времени ожидания блокировки

- Внедрение нового алгоритма в проект
- Снижение среднего времени ожидания блокировки

- Внедрение нового алгоритма в проект
- Снижение среднего времени ожидания блокировки