

ИСПЫТАНИЕ ПРОИЗВОДИТЕЛЬНОСТИ HYPERLEDGER FABRIC В СРАВНЕНИИ С MYSQL И COUCHDB

Аннотация: данная статья описывает эксперимент, проведённый с Hyperledger Fabric и двумя конкурирующими решениями для хранения данных – MySQL и CouchDB. В результате эксперимента установлено, что пропускная способность Hyperledger Fabric при отправке транзакций в 3-4 раза ниже, чем могут предложить MySQL и CouchDB, однако скорость выборки оказалась такой же, как у CouchDB, и более высокой, чем у MySQL.

Ключевые слова: hyperledger fabric, база данных, блокчейн, MySQL, CouchDB, производительность

Annotation: this article describes an experiment with Hyperledger Fabric and two other technologies for structured data storage – MySQL and CouchDB. The results of experiment showed the throughput of Hyperledger Fabric being 3-4 time worse comparing to results of MySQL and CouchDB, but the performance of queueing appeared to be on par with CouchDB and faster than MySQL.

Keywords: hyperledger fabric, database, blockchain, consensus mechanism, smart-contracts.

ВВЕДЕНИЕ

Hyperledger Fabric является платформой для закрытого частного блокчейна с возможностью распределения ролей между узлами и выполнения смарт-контрактов, называемых Chaincode в терминологии Hyperledger Fabric

и позволяющих самостоятельно указывать структуру данных в блокчейне и правила их добавления [1]. Платформа входит в открытый зонтичный проект Hyperledger, созданный некоммерческим консорциумом Linux Foundation в декабре 2015 года [2]. Блокчейн – вид базы данных, представляющей собой цепь связанных блоков данных. Механизм его действия основывается на фундаментальных свойствах – одном из видов консенсусов (примеры: Proof-of-Work и Proof-of-Stake) и открытой истории транзакции [3].

Одной из проблем, которые часто связывают с блокчейном, является производительность, а именно – пропускная способность, выражаемая в транзакциях в секунду. Таким образом, цель данной работы – провести эксперимент, который позволит сравнить различные аспекты производительности последней версии Hyperledger Fabric с несколькими конкурентными решениями для хранения данных.

МЕТОДОЛОГИЯ

Шесть физических компьютеров были объединены в локально-вычислительную сеть с использованием коммутатора и использовали одну и ту же подсеть. Основные характеристики пяти компьютеров, на которых планировалось провести:

- Процессор: Intel Celeron G3930;
- Оперативная память: 4 Гбайт;
- Операционная система: Ubuntu Minimal 18.04 x86-64;
- Пространство на диске: не менее 20 Гбайт, одинаковая производительность и состояние на всех ПК;
- Скорость сети: 1 Гбит/с.

Таким образом системные требования Hyperledger Fabric были выполнены, а следовательно, и проблем с работоспособностью системы не должно было возникнуть. Шестой компьютер использовался для контроля за остальными через ssh.

Для испытаний кроме Hyperledger Fabric версии 1.1 были также выбраны платформы MySQL версии 5.7 и CouchDB версии 2.3.0. Обе были испытаны в режиме репликации.

Для всех трёх систем была использована простая структура данных:

- id – число;
- name – строка;
- value – строка.

Для генерации значений, которые необходимо было добавлять в базы данных, а также для автоматизации их отправки был написан скрипт на Python. Скрипт генерировал 1000 значений, которые в последствии добавлялись в MySQL, CouchDB и Hyperledger Fabric. Общее время добавления выводилось после выполнения скрипта. Было выполнено в общей сложности 10 таких тестов, среди которых были выведены средние результаты.

Также был написан аналогичный скрипт с небольшими изменениями, который генерировал запросы выборки к базам данных на основании случайного выбора параметра name. Принцип аналогичен – 1000 запросов, вывод времени выполнения цикла, 10 тестов, выводение средних значений результатов.

РЕЗУЛЬТАТЫ

Подробнее с результатами можно ознакомиться на рисунке 1.

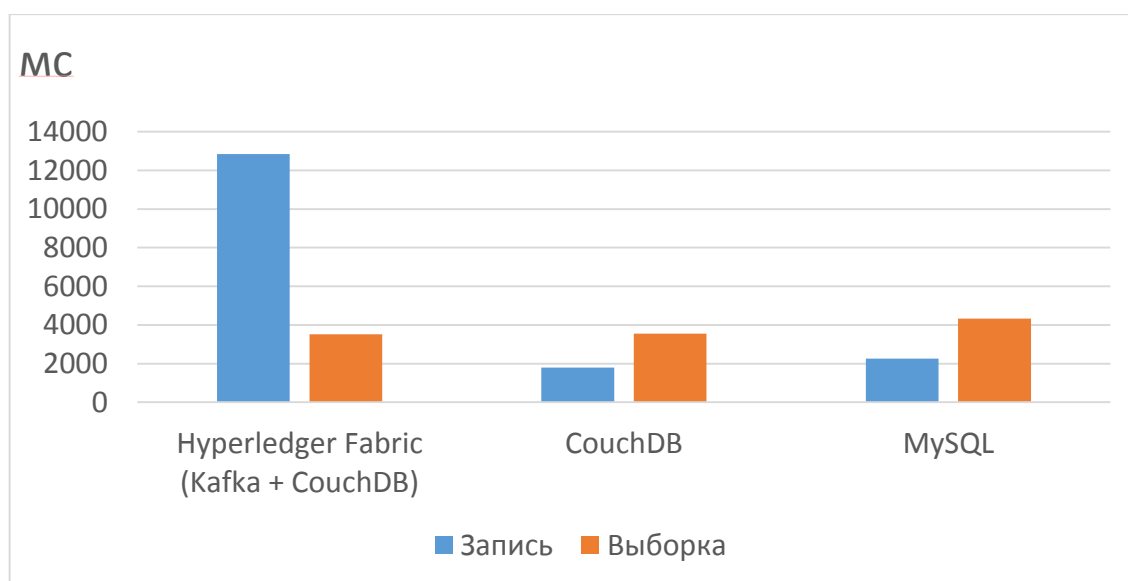


Рисунок 1 – Среднее время выполнения 1000 операций записи и выборки (пять компьютеров, репликация)

Подробное описание результатов испытаний на производительность записи:

- Hyperledger Fabric (Kafka + CouchDB): 12845 мс на 1000 операций записи, что равно 77 операциям в секунду;
- CouchDB: 1790 мс на 1000 операций записи, что равно 558 операциям в секунду;
- MySQL: 2264 мс на 1000 операций записи, что равно 442 операциям в секунду.

Таким образом, Hyperledger Fabric оказался самым медленным в скорости выполнения операций записи данных в блокчейн, что и неудивительно, т.к. если другие СУБД работают исключительно с репликацией, которая может быть отложенной, то в случае с Hyperledger Fabric используется хоть и простой, но алгоритм консенсуса.

Подробное описание результатов испытаний на производительность выборки:

- Hyperledger Fabric (Kafka + CouchDB): 3515 мс на 1000 операций выборки, что равно 284 операциям чтения в секунду;
- CouchDB: 3546 мс на 1000 операций выборки, что равно 282 операциям в секунду;
- MySQL: 4320 мс на 1000 операций выборки, что равно 231 операциям в секунду.

Интересным результатом данных испытаний стало то, что производительность при операциях выборки оказалась одинаковой между Hyperledger Fabric и CouchDB. Таким образом, можно сделать вывод, что архитектура платформы Hyperledger Fabric не влияет на скорость чтения данных из состояния блокчейна и упирается в возможности CouchDB. Оба программных продукта оказались несущественно, но быстрее MySQL.

Также было выполнено дополнительное испытание на зависимость скорости записи в базу данных от числа компьютеров в кластере. Результаты указаны на рисунке 2.

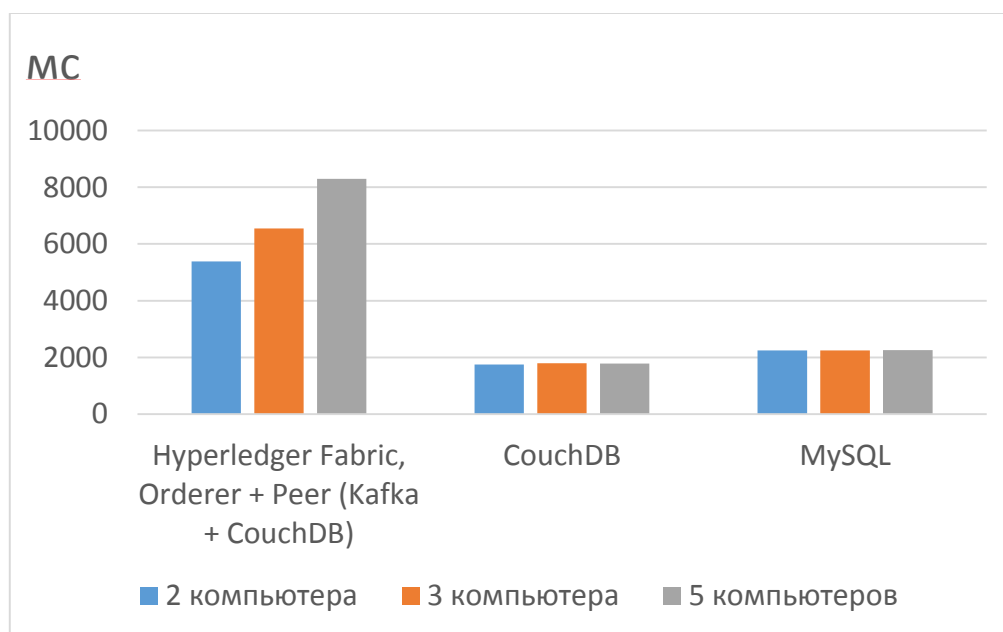


Рисунок 2 – Зависимость производительности от количества компьютеров в кластере при выполнении 1000 операций записи

Производительность MySQL и CouchDB не менялась при изменении количества компьютеров в кластере – следствие того, что в этих системах используется отложенное реплицирование. Производительность кластера Hyperledger Fabric снижалась при увеличении количества сервисных узлов (Orderers) в нём. Это можно объяснить тем, что чем больше используется сервисных узлов, тем сложнее им найти общий порядок добавления транзакций в блоки. Таким образом, можно сделать вывод о том, что для высокой производительности сети количество сервисных узлов не должно быть большим, в то время как количество обычных узлов может варьироваться в широких пределах.

ИСПОЛЬЗОВАННЫЕ ИСТОЧНИКИ

1. Architecture explained // hyperledger-fabric.readthedocs. URL: <http://hyperledger-fabric.readthedocs.io/en/release/arch-deep-dive.html> (дата обращения: 20.12.2018)
2. Хултквист Х. Что такое Hyperledger? Как Linux Foundation создает открытую платформу вокруг блокчейн проектов Intel и IBM. // Голос. Beta. 21.06.2017. URL: <https://golos.io/ru--blokchejn/@hultqvist/что-такое-hyperledger-kak-linux-foundation-sozdaet-otkrytuyu-platformu-vokrug-blokchein-proektov-intel-i-ibm> (дата обращения: 08.12.2018).
3. Бубель А.И. Возможности использования блокчейна и виртуальных токенов в таможенных операциях // Таможенная политика России на Дальнем Востоке. №3 (76). 2016. – с. 14-22.;