

Установка ClickHouse

ClickHouse установлен в докере:

- 1. Контейнер: [docker-compose.yaml](#)
- 2. Конфигурационные файлы ClickHouse: [configs](#)
- 3. Настроены volumes (конфигурационные файлы, данные, логи) для удобства отладки.

Скриншот запущенного контейнера

Загрузка датасета

[Скрипт загрузки датасета](#)

Количество записей

Тестирование производительности, изучение конфигурационных файлов

Тестирование производительности производилось запуском следующего запроса (`limit 1` установлен для уменьшения времени возвращения результатов):

```
select
    t.*,
    t2.*
from otus.trips t
left join otus.trips t2 on t.vendor_id = t2.vendor_id
limit 1

clickhouse-benchmark --query "select t.*, t2.* from otus.trips t left join otus.trips t2 on t.vendor_id = t2.vendor_id limit 1"
```

Запуск с параметрами по умолчанию

Запуск по умолчанию приводил к полной остановке ClickHouse (контейнер переходил в состояние `Exited`).

Для выяснения причины использовались логи сервера: `clickhouse-server.log`. После аудита логов выяснилось, что контейнер падал с `Out of Memory`. Установленного глобального лимита на ОЗУ в докере было недостаточного выполнения этого запроса.

Что бы контейнер не падал, был уставлено ограничение на максимальное использование ОЗУ (5 GB) для пользователя в [users.d/default.xml](#).

5368709120

После ограничения, контейнер падать перестал, однако теперь сам запрос падал по лимиту ОЗУ, из-за возможности уместить данные при джойне.

Для обхода ограничения, был принудительно уставлен алгоритм джойна, который сбрасывает на диск отсортированные результаты правой таблицы: [users.d/default.xml](#).

partial_merge

В результате запрос выполнялся и были получены результаты тестирования производительности.