**Министерство науки и высшего образования Российской Федерации**
**Федеральное государственное бюджетное образовательное учреждение высшего образования**
**«Московский государственный технический университет имени Н.Э. Баумана**
**(национальный исследовательский университет)»**
**(МГТУ им. Н.Э. Баумана)**

ФАКУЛЬТЕТ _____ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ_____

КАФЕДРА _____СИСТЕМЫ ОБРАБОТКИ ИНФОРМАЦИИ И УПРАВЛЕНИЯ_____

# РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

## *К КУРСОВОЙ РАБОТЕ*

## *НА ТЕМУ:*

## *Анализ производительности Cassandra при выполнения запросов на чтение*

Студент __ИУ5-34М_____ _____ **Н.В. Журавлев**
(Группа) (Подпись, дата) (И.О.Фамилия)

Руководитель курсовой работы _____ **Б.С. Горячкин**
(Подпись, дата) (И.О.Фамилия)

*2024 г.*

**Министерство науки и высшего образования Российской Федерации**
**Федеральное государственное бюджетное образовательное учреждение**
**высшего образования**
**«Московский государственный технический университет имени Н.Э. Баумана**
**(национальный исследовательский университет)»**
**(МГТУ им. Н.Э. Баумана)**

УТВЕРЖДАЮ
Заведующий кафедрой ___ИУ5____
(Индекс)
_____ ___В.И. Терехов__
(И.О.Фамилия)
«_04_» ____сентября_____ 2024 г.

# З А Д А Н И Е
## на выполнение курсовой работы

по дисциплине ____Эргономический анализ систем обработки и отображения информации _____

Студент группы _ИУ5-34М__

_____Журавлев Николай Вадимович _____
(Фамилия, имя, отчество)

Тема курсовой работы __Анализ производительности Cassandra при выполнения запросов на чтение__

Направленность КР (учебная, исследовательская, практическая, производственная, др.)
_____УЧЕБНАЯ_____
Источник тематики (кафедра, предприятие, НИР) _____КАФЕДРА_____

График выполнения работы: 25% к ___ нед., 50% к ___ нед., 75% к __ нед., 100% к ___ нед.

***Задание*** _Произвести теоретический анализ производительности СУБД Cassandra и через проведения эксперимента проверить его___ _правильность_____

***Оформление курсовой работы:***

Расчетно-пояснительная записка на __9__ листах формата А4.

Дата выдачи задания «_04_» ____сентября_____ 2024 г.

**Руководитель курсовой работы** _____ ____Б.С. Горячкин_____
(Подпись, дата) (И.О.Фамилия)

**Студент** _____ _____Н.В. Журавлев_____
(Подпись, дата) (И.О.Фамилия)

Примечание: Задание оформляется в двух экземплярах: один выдается студенту, второй хранится на кафедре.

# Содержание

# Введение

Базы данных NoSQL (Non SQL или Not Only SQL) стали стандартной платформой данных и основной промышленной технологией для работы с огромным ростом данных. В настоящее время они широко используются в различных рыночных нишах, включая социальные сети и другие крупномасштабные интернет-приложения, критические инфраструктуры, критически важные для бизнеса системы, IoT и промышленные приложения. Базы данных NoSQL, разработанные для обеспечения горизонтальной масштабируемости, часто предлагаются в качестве услуги поставщиками облачных услуг.

Концепция баз данных NoSQL была предложена для эффективного хранения и предоставления быстрого доступа к наборам больших данных, объем, скорость и изменчивость которых трудно обработать с помощью традиционных систем управления реляционными базами данных. Большинство хранилищ NoSQL жертвуют гарантиями ACID (атомарность, согласованность, изоляция и надёжность) в пользу свойств BASE (согласованность данных, доступность и устойчивость к разделению), что является платой за распределенную обработку данных и горизонтальную масштабируемость [1].

В данной работе рассмотрена NoSQL СУБД Cassandra, посчитано и экспериментально проверено время выполнения запросов на чтение в зависимости от количества строк и столбцов в таблице.

## Описание СУБД Cassandra

В терминологии кассандры приложение работает с пространством ключей (keyspace), что аналогично понятию схемы базы данных (database schema) в реляционной модели. В этом пространстве ключей могут находиться несколько колоночных семейств (column family), что аналогично понятию реляционной таблицы. В свою очередь, колоночные семейства содержат колонки (column),

которые объединяются при помощи ключа (row key) в записи (row). Колонка состоит из трех частей: имени (column name), метки времени (timestamp) и значения (value). Колонки в пределах записи упорядочены. В отличие от реляционной БД, никаких ограничений на то, чтобы записи (а в терминах реализационных БД – это строки) содержали колонки с такими же именами, как и в других записях – нет [2].

## Определение временных характеристик запросов к базе данных

Время выполнения запроса к БД ($t_{\text{запроса теор}}$) зависит от нескольких параметров, отраженных в формуле (1):

$$t_{\text{запроса теор}} = 2 * t_1 + t_2 + t_3, \quad (1)$$

где $t_1$ – время передачи запроса от приложения к БД (умножается на 2, так как учитывается и время передачи запроса от БД к приложению) равное 0,9мс; $t_2$ – время диспетчерской работы координатора по распределению нагрузки между серверами, равное 1мс; $t_3$ – время выполнения запроса сервером, которое зависит от особенностей выполнения запроса на конкретном сервере.

Расчет времени выполнения запроса на сервере ($t_3$) считается следующим образом [3]:

$$t_3 = t_{3.1} + t_{3.2} + t_{3.3}, \quad (2)$$

где $t_{3.1}$ – время чтения с диска в оперативную память, которое зависит от количества записей в таблице; $t_{3.2}$ – время выполнения запроса сервером, которое зависит от разных типов индексов; $t_{3.3}$ – время выполнения запроса сервером, которое зависит от категории СУБД и особенностей моделей данных.

Время выполнения запроса сервером, которое зависит от разных типов индексов ($t_{3.2}$) принимается равным нулю, так как заранее заготовленные индексированные таблицы не создавались. Так как в аналитической зависимости присутствует определенная погрешность, то для её уменьшения в параметре $t_{3.3}$ заложен поправочный коэффициент, уменьшающий расхождение $t_{\text{эксп}}$ и $t_{\text{запроса теор}}$ , в итоге $t_{3.3} = 2{,}8$мс.

Время $t_{3.1}$ рассчитывается по следующей формуле:

$$t_{3.1} = 0.0086(n * t\ + n * b)^{1,1382}, \quad (3)$$

Где n – количество строк, которые будут рассматриваться при выполнении запроса на чтение, t – количество столбцов, которые будут рассматриваться при выполнении запроса на чтение, b – смещение, которое позволяет учесть влияние количества строк и столбцов на запрос.

Смещение (b) влияние того, что строка более затратная для поиска, чем столбце из-за внутреннего устройства СУБД Cassandra. Если приравнять строку и столбцы, то для одинокого количества данных (количество строк, умноженное на количество столбцов) время выполнения будет различное. Для вычисления необходимо поделить два времени выполнения при одинаковом количестве данных, но разном количестве строк и столбцов. Смещение зависит от устройства, в данной работе оно вычислено и равно 1,8.

Формула (3) и смещение были получены с использованием метода регрессии для аппроксимации функции одной переменной [4, 5].

Для определения время выполнения запроса используется механизм трассировки в Cassandra, который сохраняет данные о выполнения запроса, включая время начала и завершения [6].

# Экспериментальное исследование производительности СУБД Cassandra

## Описание устройства для проведения эксперимента

Оценка времени выполнения запросов запроса зависит от производительности устройства, на котором проводятся эксперименты. Поэтому приведем характеристики устройств, на которых будут проводиться эксперименты.

Технические устройства

1) Процессор: 8 ядерный процессор с тактовой частотой 2,6 ГГц;
2) Оперативная память: DDR3 8192 МБ со скоростью 3000 Мгц;
3) Диск: Пустой SSD диск объемом 10 Гб со скоростью 1900 Мбит/с;

Программное обеспечение:

1) Операционная система Ubuntu 24.04.

Практический эксперимент проводится на операционной системе Ubuntu 24.04, работающей на базе Virtual Box на Windows 10. Во время проведения эксперимента отсутствуют фоновые процессы, помимо фоновых процессов операционной системы.

В качестве основы для эксперимента в базу данных были загружены все варианты объединения строк и столбцов следующего количества: 10000/100000/200000/300000 строк и 5/10/15 столбцов. Каждый такой вариант объединения является одной частью эксперимента.

## Методика проведения эксперимента

Для проведения исследования по оценке времени выполнения запросов на чтение с условиями требуется выполнить следующие задачи:

1. Создать базу данных и пространство ключей;

2. Создать таблицу с нужным для определённой части эксперимента количеством строк и столбцов и заполнить её тестовыми данными;
3. Выполнить запрос для получения экспериментального времени выполнения запроса. Данный пункт выполняется три раза подряд, после чего для всех результатов рассчитывается среднее значение, которое будет считаться экспериментальным временем выполнения запросов;
4. Пункты 1-3 повторяются для каждой части эксперимента.

**Результаты экспериментальные исследования производительности**

При проведении практического эксперимента соберем данные для формирования зависимости времени выполнения запроса от количества строк и столбцов.

В табл. 1 представлен результат выполнения эксперимента с последовательным увеличением количества строк и столбцов в таблицах.

Таблица 1. Результаты экспериментального выполнения запроса

| | | **Столбцы** | **Строки** | | | |
|---|---|---|---|---|---|---|
| | | | **10 000** | **100 000** | **200 000** | **300 000** |
| **t, мс** | **Теоретическое время выполнения** | **5** | 76,23372036 | 1047,965812 | 2306,638071 | 3659,371715 |
| | | **10** | 142,7462475 | 1962,296822 | 4319,137613 | 6852,106627 |
| | | **15** | 213,3932607 | 2933,463572 | 6456,736164 | 10243,30517 |
| | **Экспериментальное время выполнения** | **5** | 118,698 | 819,4866667 | 3835,273 | 5686,882667 |
| | | **10** | 117,846 | 1125,921333 | 4461,397 | 7657,586667 |
| | | **15** | 154,1826667 | 3172,168333 | 5985,99 | 8049,659333 |

## Оценка результатов

Обобщив теоретические выкладки и практические исследования, получим следующие результаты, представленные в табл. 2.

Таблица 2. Анализ времени выполнения запроса

| | | **Время выполнения, мс** | | | |
|---|---|---|---|---|---|
| **Строка** | **Столбец** | **Экспериментальное** | **Теоретическое** | **Относительная погрешность** | **Абсолютная погрешность, мс** |
| 10 000 | 5 | 118,698 | 76,23372036 | 36% | 42,46427964 |
| 10 000 | 10 | 117,846 | 142,7462475 | 21% | 24,9002475 |
| 10 000 | 15 | 154,1826667 | 213,3932607 | 38% | 59,210594 |
| 100 000 | 5 | 819,4866667 | 1047,965812 | 28% | 228,4791453 |
| 100 000 | 10 | 1125,921333 | 1962,296822 | 74% | 836,375489 |
| 100 000 | 15 | 3172,168333 | 2933,463572 | 8% | 238,704761 |
| 200 000 | 5 | 3835,273 | 2306,638071 | 40% | 1528,634929 |
| 200 000 | 10 | 4461,397 | 4319,137613 | 3% | 142,259387 |
| 200 000 | 15 | 5985,99 | 6456,736164 | 8% | 470,746164 |
| 300 000 | 5 | 5686,882667 | 3659,371715 | 36% | 2027,510952 |
| 300 000 | 10 | 7657,586667 | 6852,106627 | 11% | 805,48004 |
| 300 000 | 15 | 8049,659333 | 10243,30517 | 27% | 2193,645837 |

Данные из таблицы 2 свидетельствуют о том, что время при увеличении количества строк в 10 и более раз увеличивается примерно в 9 раз. Увеличение количества столбцов в меньшей степени, чем строки влияет на производительность СУБД, например, при увеличении в 2 раза, время выполнения увеличивается примерно на чуть меньше 50%. Зависимость времени выполнения от увеличения строк и столбцов показана на рис 1, 2 соответственно.

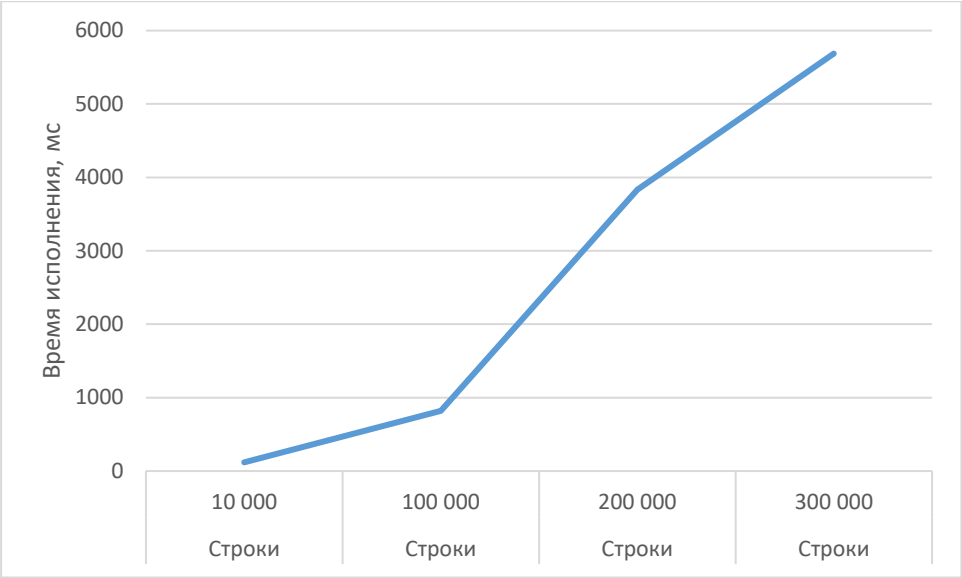


Рисунок 1. Зависимость времени выполнения от количества строк

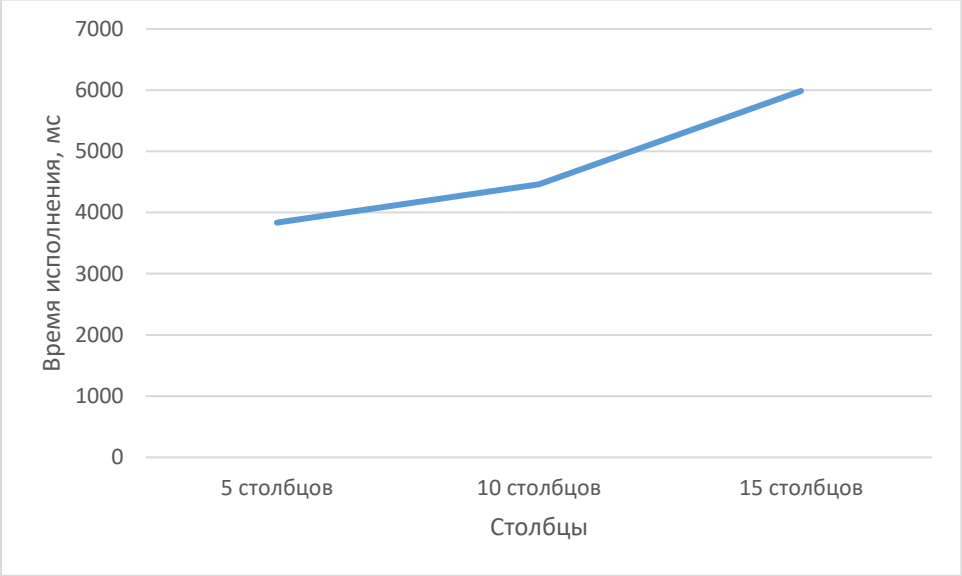


Рисунок 2. Зависимость времени выполнения от количества столбцов

Зависимость экспериментального и теоретического времени выполнения от количества строк и столбцов представлены на рис 3, 4 соответственно.

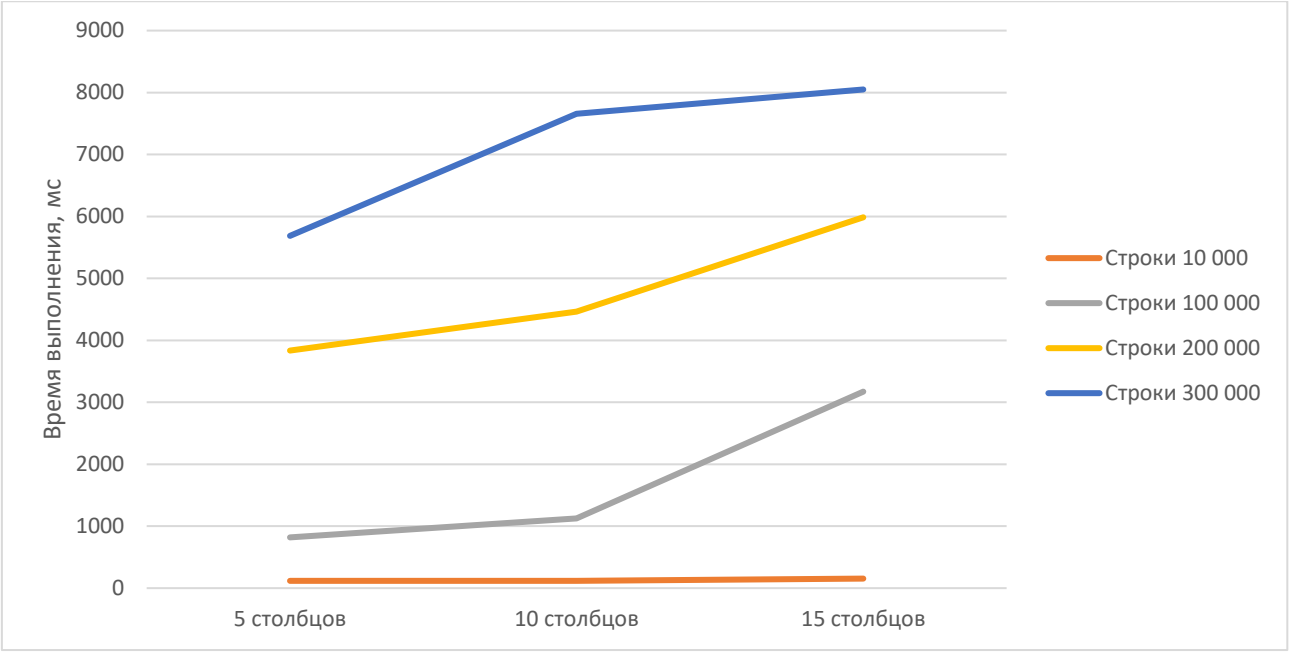


Рисунок 3. Зависимость экспериментального времени выполнения от количества строк и столбцов

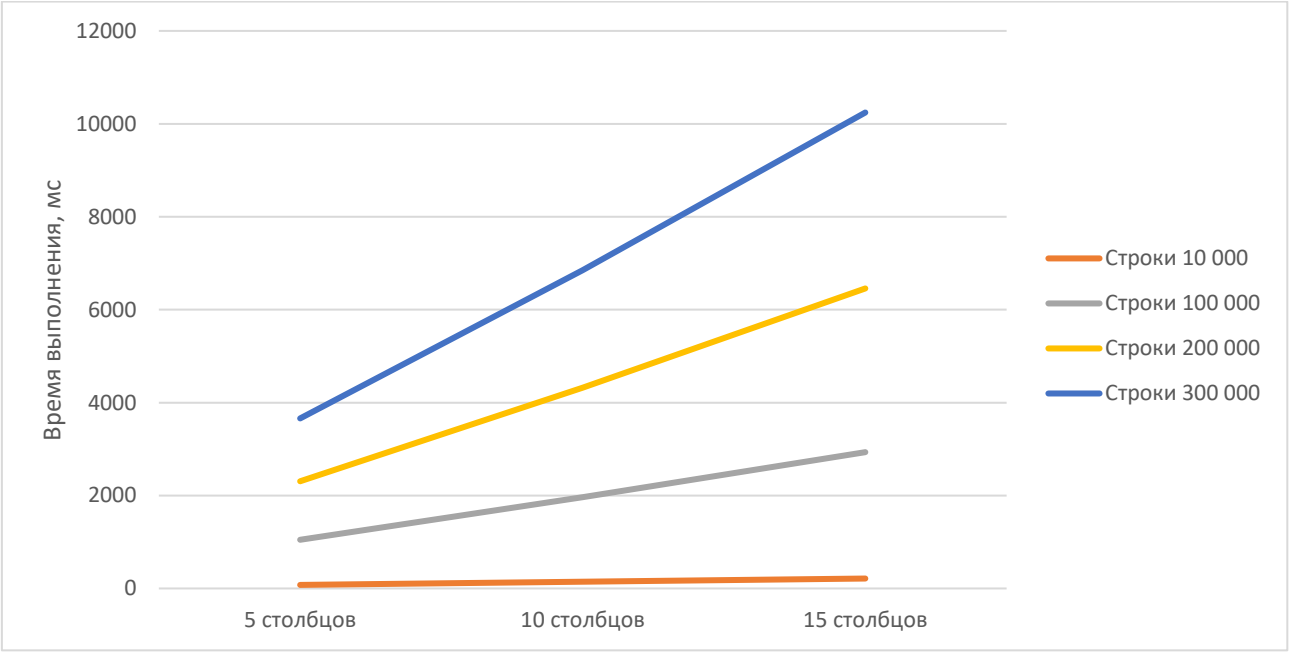


Рисунок 4. Зависимость теоретического времени выполнения от количества строк и столбцов

Средняя ошибка аппроксимации составляет 27,4%. В абсолютных же значениях это настолько незначительно, что в подавляющем числе случаев абсолютное значение меньше, чем время реакции человека (0,3 с), с учётом времени зрительного восприятия человека (0,9 – 0,95 с), то значение нас

устраивает [7]. График относительной и абсолютной погрешности от строк и столбцов показаны на рис. 5, 6 соответственно.

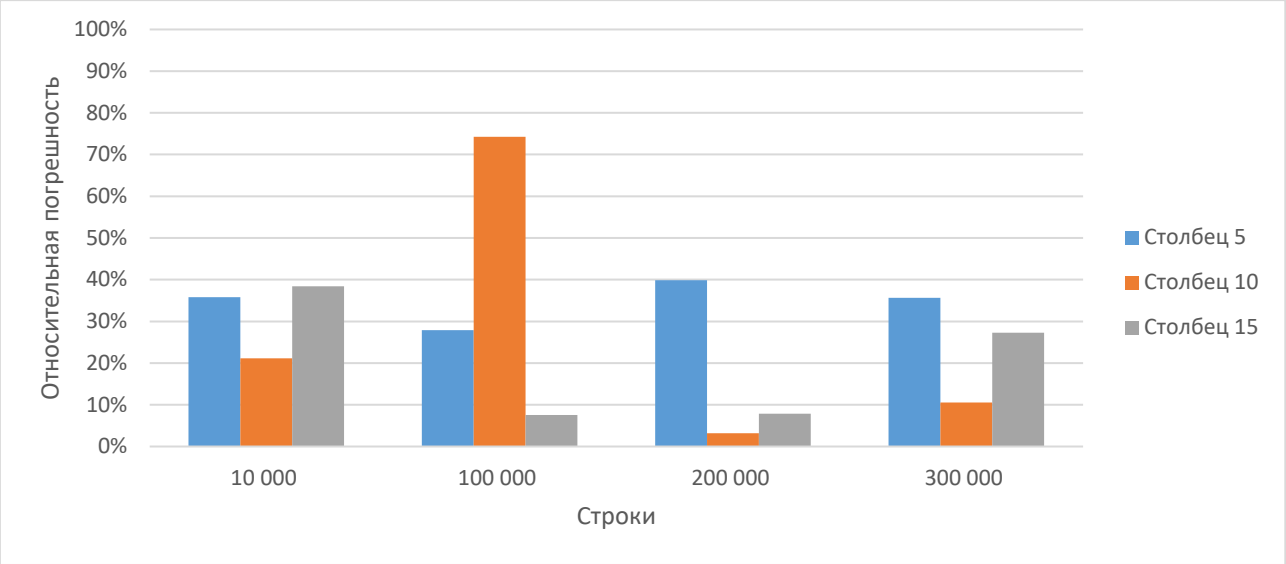


Рисунок 5. Зависимость относительной погрешности от количества строк и столбцов

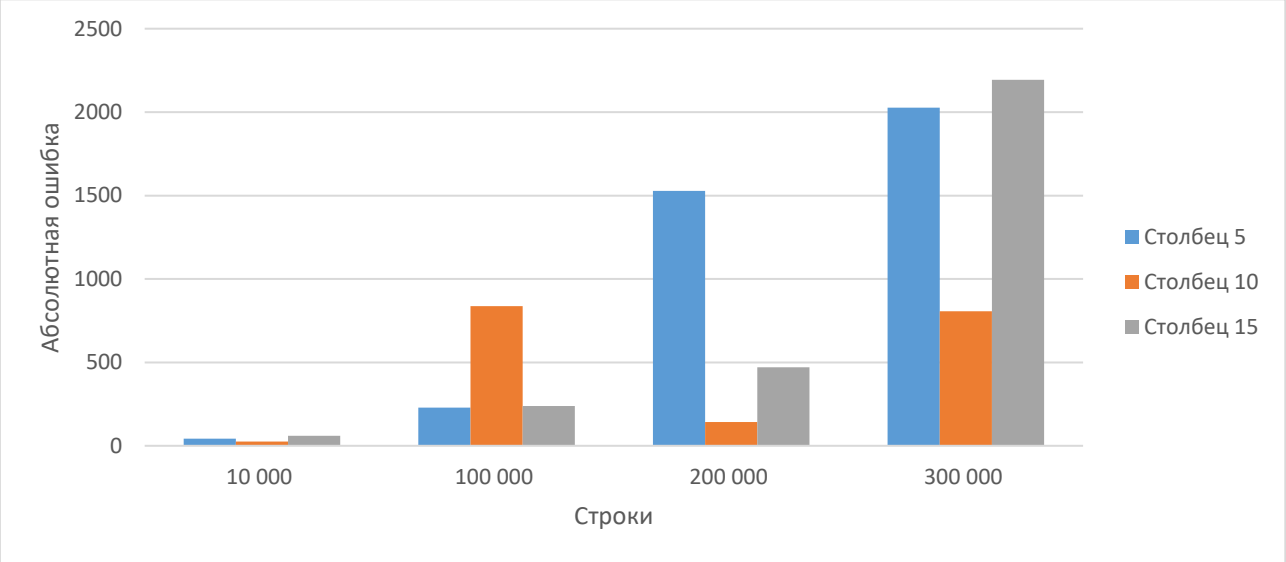


Рисунок 6. Зависимость абсолютной погрешности от количества строк и столбцов

Гистограмма сравнительного анализа приведена на рис. 7. Шкала слева показывает время выполнения запросов (t, мс), шкала снизу - количество строк. Цветовая гамма столбцов на гистограмме показывает количество столбцов для экспериментального и теоретического времени выполнения.

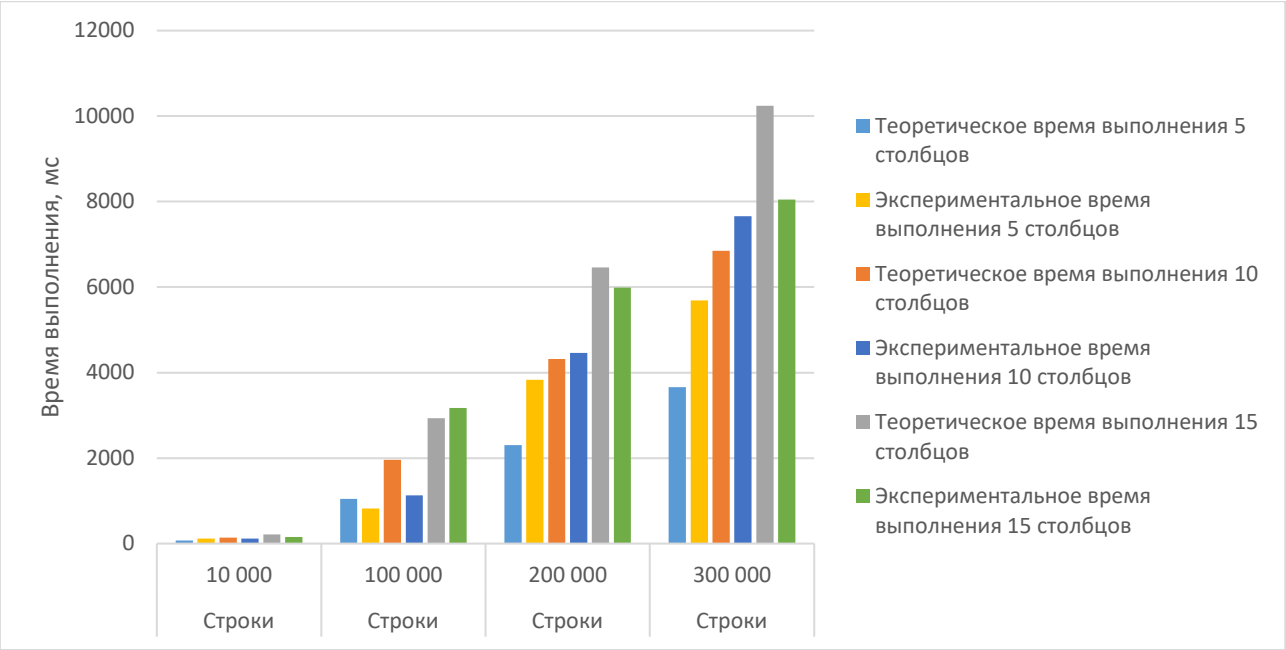


Рисунок 7. Гистограмма сравнительного анализа

## Заключение

В настоящей статье были проведены исследования по оценке времени выполнения запроса на чтение в СУБД Cassandra в зависимости от количества строк и столбцов. Проведён эксперимент по исследованию производительности и произведена оценка результатов, по которым можно позволяют сделать вывод, что изменение количества столбцов практически влияет на время выполнения запросов и на производительность СУБД в большой степени, чем изменения количества строк. При этом, стоит отметить, что на практике таблицы, которые имеют более 15 столбцов, используются крайне редко.

Также в работе определены формулы аналитических зависимостей времен выполнения запросов от числа строк и столбцов в таблицах. И показано на сколько результаты расчетов близки к результатам, полученным экспериментально.

## Список использованных источников

1. Gorbenko, A and Romanovsky, A and Tarasyuk, O. Interplaying Cassandra NoSQL Consistency and Performance: a Benchmarking Approach //

Communications in Computer and Information Science., 1279. pp. 168-184. ISSN 1865-0929 DOI: https://doi.org/10.1007/978-3-030-58462-7_14

2. Как устроена apache cassandra. URL: https://habr.com/ru/articles/155115/ (дата обращения: 11.09.2024).
3. Елисеева Е.А., Горячкин Б.С., Виноградова М.В, Черненький М.В. Оценка времени выполнения поисковых запросов в NoSQL и объектно-реляционной базах данных // Динамика сложных систем – XXI ВЕК. 2022. Т. 15. №2. С.44-51.
4. Данилов А.М., Гарькина И.А. Интерполяция, аппроксимация, оптимизация: анализ и синтез сложных систем. Пенза.: ПГУАС. 2014.
5. Онлайн калькуляторы PLANETCALC. URL: https://planetcalc.ru/5992 (дата обращения: 11.12.2024).
6. TRACING. URL: https://docs.datastax.com/en/cql-oss/3.3/cql/cql_reference/cqlshTracing.html (дата обращения: 6.11.2024).
7. Горячкин Б.С. Эргономический анализ систем обработки информации и управления // Вестник евразийской науки. 2017. Т. 9. №3. С. 72.