

# **Отчёт по нагрузочному тестированию сервиса “book”” и базы данных PostgreSQL**

# Оглавление

|  |    |
|--|----|
| Цели, задачи и результаты тестирования .....                                     | 3  |
| Цели и задачи тестирования .....   | 3  |
| Профиль нагрузки .....   | 3  |
| Результаты тестирования (до оптимизации БД).....                                 | 4  |
| Результаты тестирования (после оптимизации БД) .....                             | 5  |
| Проведенные тесты.....   | 6  |
| Максимальная производительность (до оптимизации БД).....                         | 6  |
| Подтверждение максимальной производительности (до оптимизации БД) .....          | 7  |
| Анализ производительности БД.....  | 8  |
| Оптимизация скорости выполнения запросов на БД.....                              | 9  |
| Максимальная производительность (после оптимизации БД).....                      | 10 |
| Подтверждение максимальной производительности (после оптимизации БД) .....       | 11 |
| Тест надёжности (после оптимизации БД) .....                                     | 12 |
| Количество коннектов от сервиса к БД.....  | 13 |
| Изменение количества потоков и коннектов (Threads-5   Maximum pool size-3) ..... | 14 |
| Threads-10    Maximum pool size-3 tps-N19.....                                   | 14 |
| Threads-5     Maximum pool size-6 tps-N19 .....                                  | 15 |
| Threads-20    Maximum pool size-12 tps-N19.....                                  | 16 |
| Threads-40    Maximum pool size-24 tps-N19.....                                  | 17 |
| Threads-80    Maximum pool size-48 tps-N19.....                                  | 18 |
| Threads-160   Maximum pool size-96 tps-N19.....                                  | 19 |
| Threads-80    Maximum pool size-3 tps-N19.....                                   | 20 |
| Threads-5     Maximum pool size-48 tps-N19 .....                                 | 21 |

# Цели, задачи и результаты тестирования

## Цели и задачи тестирования

- 1) Определить максимальную производительность.
- 2) Определить надёжность на 100% профиля.
- 3) Выполнить анализ производительности БД.
- 4) Оптимизировать скорость выполнения запросов на БД.
- 5) Оптимизировать скорость выполнения запросов на БД.
- 6) Повторить тест после оптимизаций
- 7) Определить количество коннектов от сервиса к БД.
- 8) Подобрать оптимальное количество потоков и коннектов и сравнить производительность.

## Профиль нагрузки

| № | Операция                               | Интенсивность | %      |
|---|--|---------------|--------|
| 1 | Информация о книге (по автору)         | 10 в секунду. | ≈29,41 |
| 2 | Информация о книге (по имени)          | 10 в секунду. | ≈29,41 |
| 3 | Изменить цену                          | 6 в секунду   | ≈17,65 |
| 4 | Информация о книге (по имени и автору) | 5 в секунду.  | ≈14,71 |
| 5 | Список книг с одной страницы           | 1 в секунду   | ≈2,94  |
| 6 | Удалить книгу                          | 1 в секунду   | ≈2,94  |
| 7 | Добавить книгу                         | 1 в секунду   | ≈2,94  |

## Результаты тестирования (до оптимизации БД)

1) Максимальная производительность составляет, около **23.8 запросов в секунду**.

| № | Операция                               | Интенсивность  |
|---|--|----------------|
| 1 | Информация о книге (по автору)         | 7 в секунду.   |
| 2 | Информация о книге (по имени)          | 7 в секунду.   |
| 3 | Изменить цену                          | 4.2 в секунду  |
| 4 | Информация о книге (по имени и автору) | 3.5 в секунду. |
| 5 | Список книг с одной страницы           | 0.7 в секунду  |
| 6 | Удалить книгу                          | 0.7 в секунду  |
| 7 | Добавить книгу                         | 0.7 в секунду  |

2) Среднее время выполнения базой данных запросов, с поиском книг по имени или автору около **100мс**.

| 123 calls | 123 total_exec_time | 123 mean_exec_time | ABC query  |
|-----------|---------------------|--------------------|--|
| 4 985     | 510 749,4615999996  | 102,4572641123     | select bookv2x0_id as id1_0_ bookv2x0_author as author2_0_ bo            |
| 5 029     | 502 209,16399999999 | 99,8626295486      | select bookv2x0_id as id1_0_ bookv2x0_author as author2_0_ bo            |
| 3 105     | 357 309,6287000003  | 115,0755647987     | update bookv2 set price=\$1 where author=\$2 and name=\$3                |
| 2 615     | 264 353,16009999999 | 101,091074608      | select bookv2x0_id as id1_0_ bookv2x0_author as author2_0_ bo            |
| 547       | 62 879,7653         | 114,9538670932     | delete from bookv2 where author=\$1 and name=\$2                         |
| 547       | 51 033,5837         | 93,2972279707      | select count(bookv2x0_id) as col_0_0_ from bookv2 bookv2x0_              |
| 547       | 5 769,9628          | 10,5483780622      | select bookv2x0_id as id1_0_ bookv2x0_author as author2_0_ bo            |
| 547       | 33,8886             | 0,0619535649       | insert into bookv2 (author, name, price, id) values (\$1, \$2, \$3, \$4) |

## Результаты тестирования (после оптимизации БД)

1) Максимальная производительность составляет, около 646 запросов в секунду.

| № | Операция                               | Интенсивность  |
|---|--|----------------|
| 1 | Информация о книге (по автору)         | 190 в секунду. |
| 2 | Информация о книге (по имени)          | 190 в секунду. |
| 3 | Изменить цену                          | 114 в секунду  |
| 4 | Информация о книге (по имени и автору) | 95 в секунду.  |
| 5 | Список книг с одной страницы           | 19 в секунду   |
| 6 | Удалить книгу                          | 19 в секунду   |
| 7 | Добавить книгу                         | 19 в секунду   |

2) Среднее время выполнения запросов базой данных, с поиском книг по имени или автору около 0,05мс.

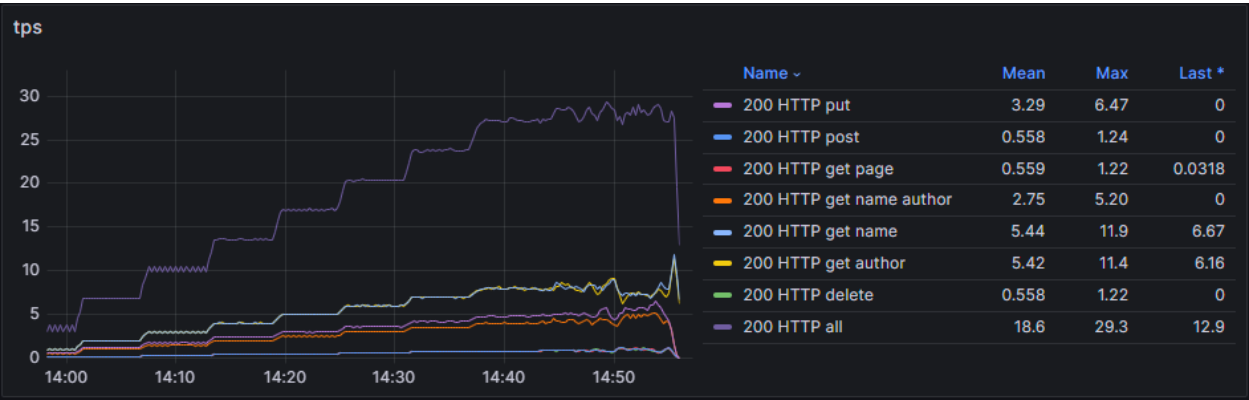
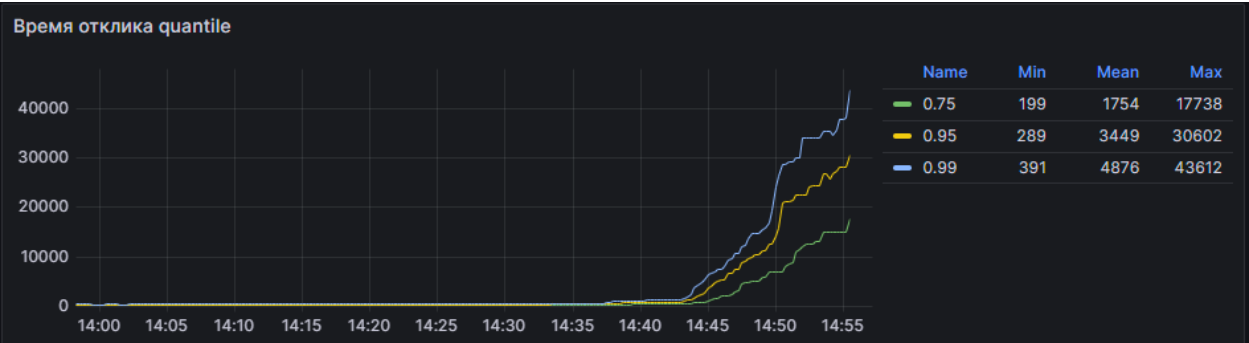
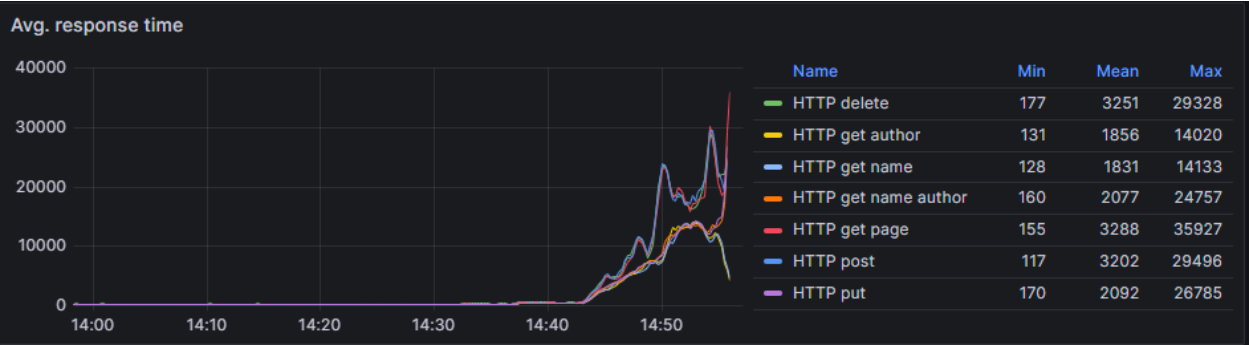
|   | 123 calls | 123 total_exec_time | 123 mean_exec_time | ABC query  |
|---|-----------|---------------------|--------------------|--|
| 1 | 3 467     | 132,2776            | 0,0381533314       | select bookv2x0_id as id1_0_, bookv2x0_.author as author2_0_, boo        |
| 2 | 3 455     | 159,9176            | 0,0462858466       | select bookv2x0_id as id1_0_, bookv2x0_.author as author2_0_, boo        |
| 3 | 2 072     | 113,6916            | 0,0548704633       | update bookv2 set price=\$1 where author=\$2 and name=\$3                |
| 4 | 1 729     | 66,1347             | 0,0382502603       | select bookv2x0_id as id1_0_, bookv2x0_.author as author2_0_, boo        |
| 5 | 348       | 4 012,589           | 11,5304281609      | select bookv2x0_id as id1_0_, bookv2x0_.author as author2_0_, boo        |
| 6 | 347       | 36,6957             | 0,1057512968       | insert into bookv2 (author, name, price, id) values (\$1, \$2, \$3, \$4) |
| 7 | 347       | 15,66               | 0,045129683        | delete from bookv2 where author=\$1 and name=\$2                         |
| 8 | 347       | 25 908,6662         | 74,6647440922      | select count(bookv2x0_id) as col_0_0_ from bookv2 bookv2x0_              |

# Проведенные тесты

## Максимальная производительность (до оптимизации БД)

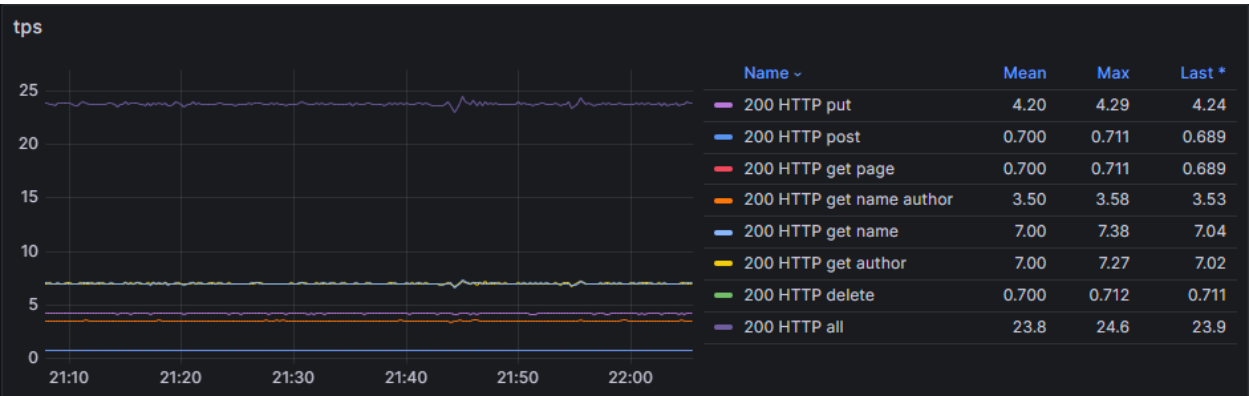
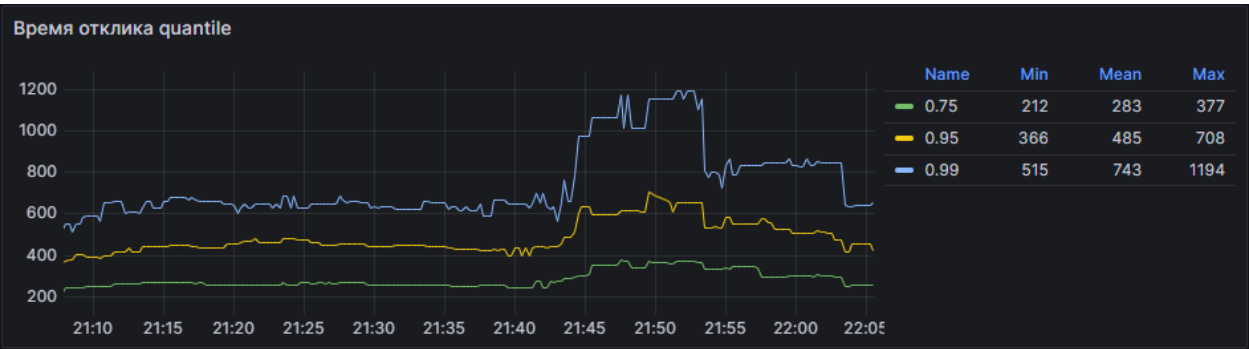
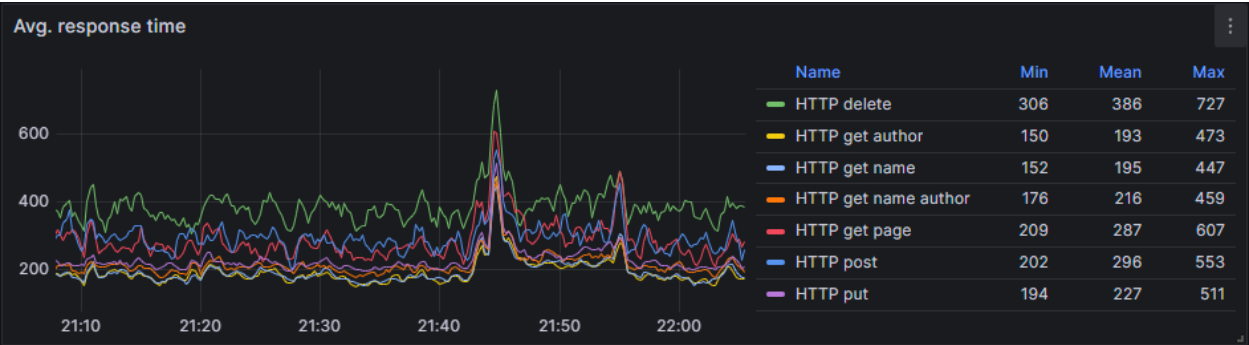
При проведении серии тестов с автоматическим управлением количества потоков и интенсивностью операций эквивалентной профилю нагрузки, умноженной на коэффициент N, установлено начало деградации системы при коэффициенте **N=0,7(23,8 запросов в секунду)**

Деградация выражалась в резком увеличении времени ответа сервиса.



# Подтверждение максимальной производительности (до оптимизации БД)

Плавное повышение нагрузки до максимальной N=0.7 (23,8 запросов в секунду) и фиксация на 1 час.



При выполнении теста с нагрузкой N=0,7 профиля(23,8 запросов в секунду), деградации сервиса не наблюдалось.

# Анализ производительности БД.

С помощью расширения «pg\_stat\_statements» была собрана статистика запросов к БД.

|   | 123 calls | 123 total_exec_time  | 123 mean_exec_time | ABC query  |
|---|-----------|----------------------|--------------------|--|
| 1 | 116 034   | 10 493 442,822699957 | 90,4342074108      | select bookv2x0_id as id1_0_, bookv2x0_.author as author2_0_, bookv2x0_.name as name3_0_ |
| 2 | 116 187   | 10 313 719,809699861 | 88,7682770852      | select bookv2x0_id as id1_0_, bookv2x0_.author as author2_0_, bookv2x0_.name as name3_0_ |
| 3 | 69 924    | 7 974 114,197600036  | 114,0397316744     | update bookv2 set price=\$1 where author=\$2 and name=\$3                                |
| 4 | 58 341    | 5 526 098,814599991  | 94,7206735332      | select bookv2x0_id as id1_0_, bookv2x0_.author as author2_0_, bookv2x0_.name as name3_0_ |
| 5 | 16 687    | 1 514 610,0555000158 | 90,7658689699      | select count(bookv2x0_id) as col_0_0_ from bookv2 bookv2x0_                              |
| 6 | 11 734    | 1 335 370,434000001  | 113,8035140617     | delete from bookv2 where author=\$1 and name=\$2   |
| 7 | 16 682    | 133 053,2750000004   | 7,97585871         | select bookv2x0_id as id1_0_, bookv2x0_.author as author2_0_, bookv2x0_.name as name3_0_ |

По статистике видно, среднее время выполнения запросов, с поиском книг по имени или автору около 100мс.

Были рассмотрены планы запросов с поиском книг по имени или автору

|   | ABC QUERY PLAN   |
|---|--|
| 1 | Gather (cost=1000.00..20889.43 rows=1 width=84) (actual time=48.855..111.821 rows=1 loops=1)                               |
| 2 | Workers Planned: 2   |
| 3 | Workers Launched: 2  |
| 4 | -> Parallel Seq Scan on bookv2 bookv2x0_ (cost=0.00..19889.33 rows=1 width=84) (actual time=19.949..39.086 rows=0 loops=3) |
| 5 | Filter: (author = 'Пахомий Савонов':text)  |
| 6 | Rows Removed by Filter: 333334   |
| 7 | Planning Time: 0.074 ms  |
| 8 | Execution Time: 111.845 ms   |

|   | ABC QUERY PLAN   |
|---|--|
| 1 | Gather (cost=1000.00..21933.52 rows=1 width=84) (actual time=44.270..95.746 rows=1 loops=1)                                |
| 2 | Workers Planned: 2   |
| 3 | Workers Launched: 2  |
| 4 | -> Parallel Seq Scan on bookv2 bookv2x0_ (cost=0.00..20933.42 rows=1 width=84) (actual time=22.478..37.991 rows=0 loops=3) |
| 5 | Filter: ((name = 'Ассоциация и Благодарность':text) AND (author = 'Евстолия Побегайлова':text))                            |
| 6 | Rows Removed by Filter: 333334   |
| 7 | Planning Time: 0.078 ms  |
| 8 | Execution Time: 95.771 ms  |

|   | ABC QUERY PLAN   |
|---|--|
| 1 | Update on bookv2 (cost=0.00..29694.42 rows=0 width=0) (actual time=119.451..119.451 rows=0 loops=1)      |
| 2 | -> Seq Scan on bookv2 (cost=0.00..29694.42 rows=1 width=10) (actual time=61.610..119.053 rows=1 loops=1) |
| 3 | Filter: ((author = 'qfvhu':text) AND (name = 'pozifwvfon':text))   |
| 4 | Rows Removed by Filter: 1000002  |
| 5 | Planning Time: 0.064 ms  |
| 6 | Execution Time: 119.541 ms   |

Поиск записи производится с помощью последовательного сканирования (Seq Scan), а отношение найденных записей к их общему количеству  $\approx 1/1000000$ .

Сделан вывод: у значений в столбцах «name» и «author» отсутствует индекс.



# Оптимизация скорости выполнения запросов на БД.

Были добавлены индексы «b-tree» значениям «name», «author».

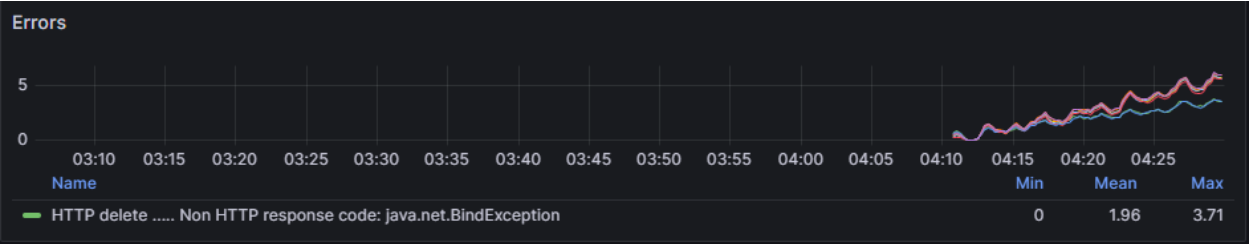
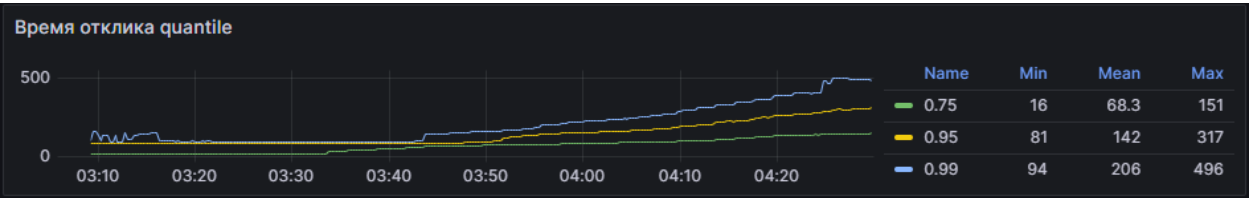
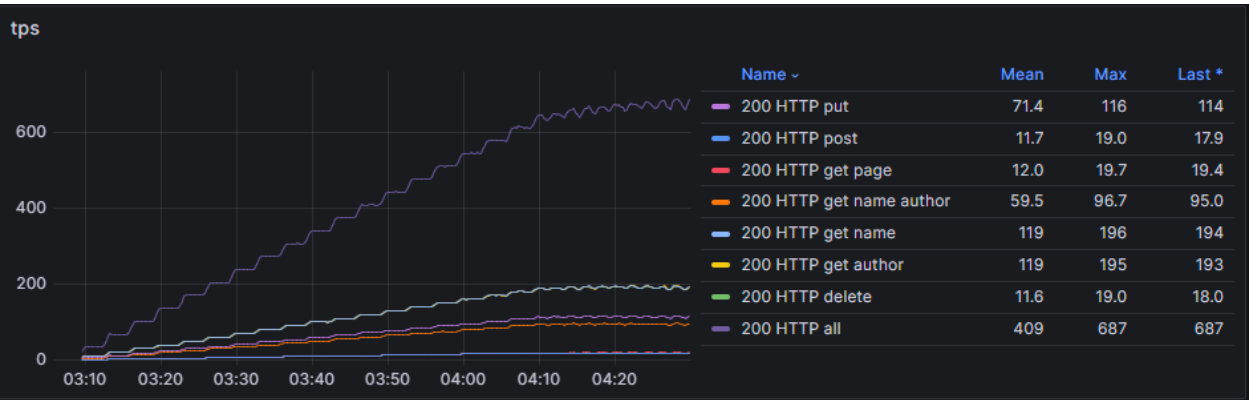
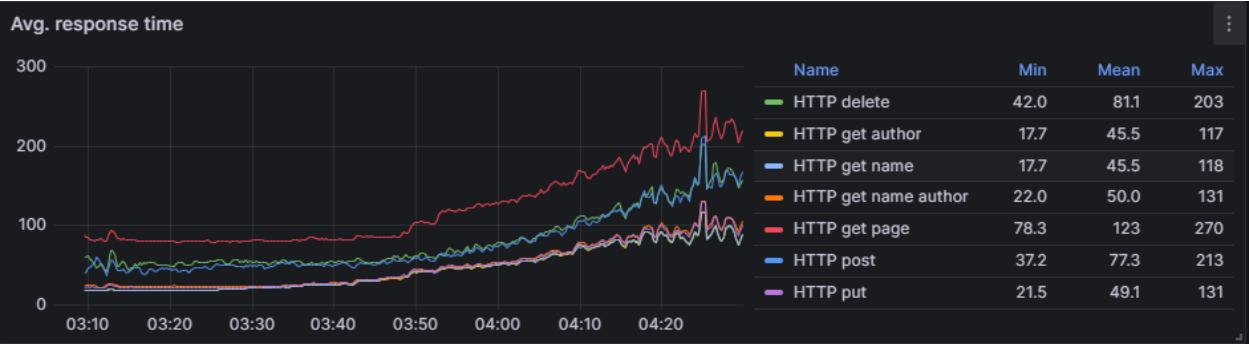
|   | 123 calls | 123 total_exec_time | 123 mean_exec_time | ABC query  |
|---|-----------|---------------------|--------------------|--|
| 1 | 3 470     | 231,1424            | 0,0666116427       | select bookv2x0_id as id1_0_, bookv2x0_.author as author2_0_, bookv2x0_.name as name3_0_ |
| 2 | 3 470     | 218,2882            | 0,0629072622       | select bookv2x0_id as id1_0_, bookv2x0_.author as author2_0_, bookv2x0_.name as name3_0_ |
| 3 | 2 082     | 207,7151            | 0,0997670989       | update bookv2 set price=\$1 where author=\$2 and name=\$3                                |
| 4 | 1 735     | 104,2557            | 0,0600897406       | select bookv2x0_id as id1_0_, bookv2x0_.author as author2_0_, bookv2x0_.name as name3_0_ |
| 5 | 347       | 34,5172             | 0,0994731988       | insert into bookv2 (author, name, price, id) values (\$1, \$2, \$3, \$4)                 |
| 6 | 347       | 24,5657             | 0,0707945245       | delete from bookv2 where author=\$1 and name=\$2   |
| 7 | 347       | 3 806,7998          | 10,970604611       | select bookv2x0_id as id1_0_, bookv2x0_.author as author2_0_, bookv2x0_.name as name3_0_ |
| 8 | 347       | 23 433,3688         | 67,5313221902      | select count(bookv2x0_id) as col_0_0_ from bookv2 bookv2x0_                              |

Время выполнения запросов значительно уменьшилось.

# Максимальная производительность (после оптимизации БД)

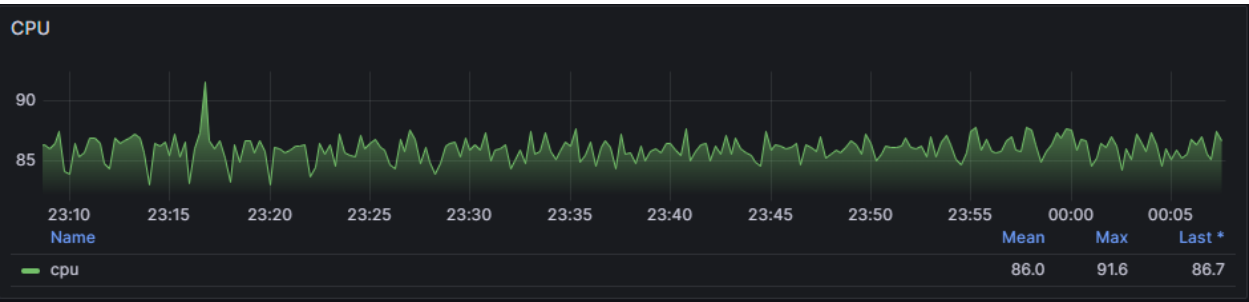
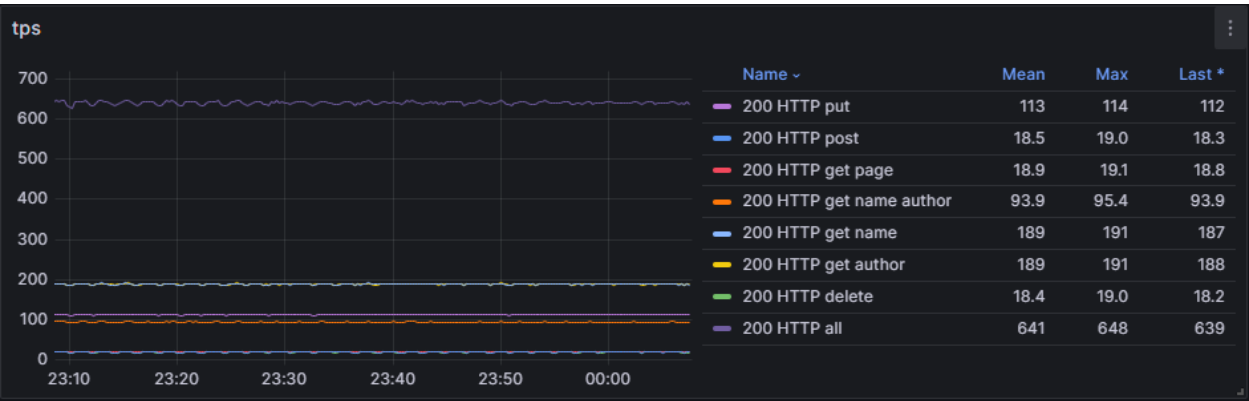
При проведении серии тестов с автоматическим управлением количества потоков и интенсивностью операций эквивалентной профилю нагрузки, умноженной на коэффициент N, установлено начало деградации системы при коэффициенте N=19(646 запросов в секунду)

Деградация выражалась в увеличении времени ответа сервиса и появлении ошибок.



# Подтверждение максимальной производительности (после оптимизации БД)

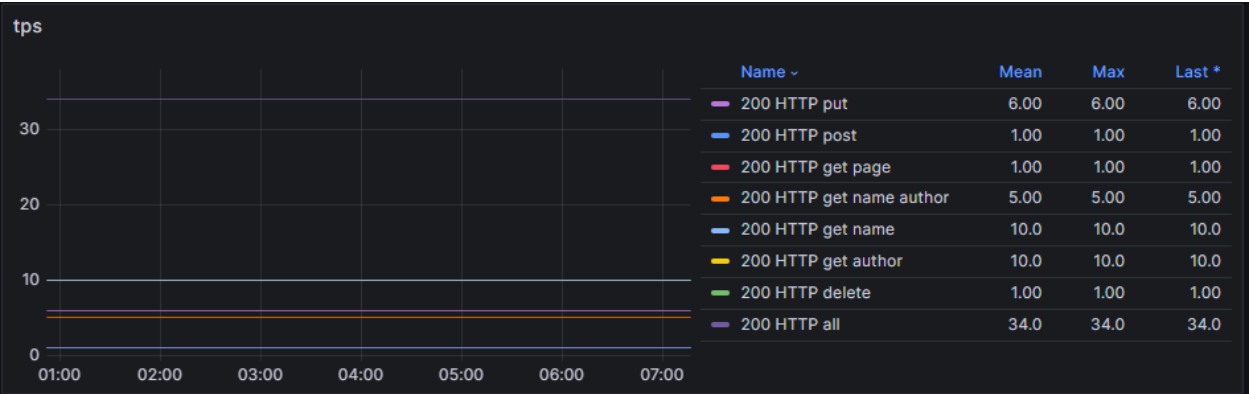
Плавное повышение нагрузки до максимальной N=19 (646 запросов в секунду) и фиксация на 1 час.



При выполнении теста с нагрузкой N=19 профиля (646 запросов в секунду), деградации сервиса не наблюдалось.

# Тест надёжности (после оптимизации БД)

Тест проводился при нагрузке 100% профиля (34 запросов в секунду) 6 часов.



Метрики стабильны. Утечки памяти не выявлено.

# Количество коннектов от сервиса к БД

Были найдены все текущие коннекты к БД

Три коннекта из текущих от сервиса.

pg\_stat\_activity 1 X

select \* from pg\_catalog.pg\_stat\_activity

Введите SQL выражение чтобы отфильтровать результаты

Таблица

Текст

с1

|    | 123 datid | abc datname | 123 pid | 123 leader_pid | 123 usesysid | abc username | abc application_name                      | client_addr |
|----|-----------|-------------|---------|----------------|--------------|--------------|---|-------------|
| 1  | [NULL]    | [NULL]      | 5 180   | [NULL]         | [NULL]       | [NULL]       |   | [NULL]      |
| 2  | [NULL]    | [NULL]      | 5 188   | [NULL]         | 10           | postgres     |   | [NULL]      |
| 3  | 5         | postgres    | 15 064  | [NULL]         | 10           | postgres     | DBeaver 23.3.4 - Main <postgres>          | 127.0.0.1   |
| 4  | 5         | postgres    | 12 928  | [NULL]         | 10           | postgres     | DBeaver 23.3.4 - Metadata <postgres>      | 127.0.0.1   |
| 5  | 5         | postgres    | 10 736  | [NULL]         | 10           | postgres     | DBeaver 23.3.4 - SQLEditor <Script-2.sql> | 127.0.0.1   |
| 6  | 5         | postgres    | 5 952   | [NULL]         | 10           | postgres     | PostgreSQL JDBC Driver                    | 127.0.0.1   |
| 7  | 5         | postgres    | 14 460  | [NULL]         | 10           | postgres     | PostgreSQL JDBC Driver                    | 127.0.0.1   |
| 8  | 5         | postgres    | 13 400  | [NULL]         | 10           | postgres     | PostgreSQL JDBC Driver                    | 127.0.0.1   |
| 9  | [NULL]    | [NULL]      | 4 964   | [NULL]         | [NULL]       | [NULL]       |   | [NULL]      |
| 10 | [NULL]    | [NULL]      | 4 508   | [NULL]         | [NULL]       | [NULL]       |   | [NULL]      |
| 11 | [NULL]    | [NULL]      | 5 172   | [NULL]         | [NULL]       | [NULL]       |   | [NULL]      |

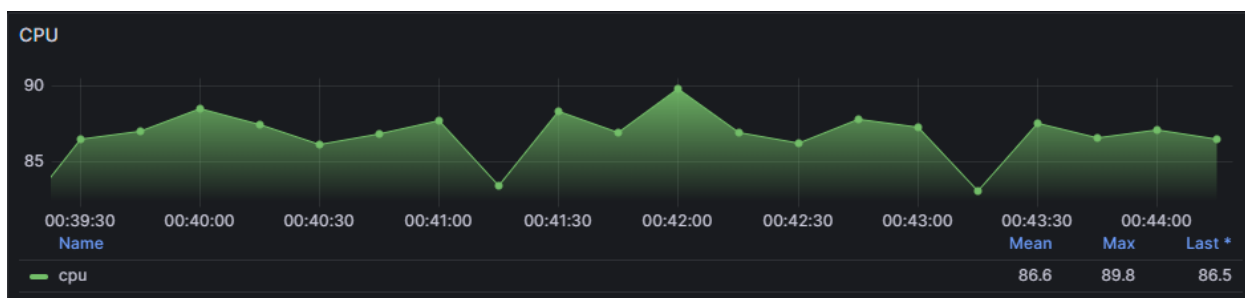
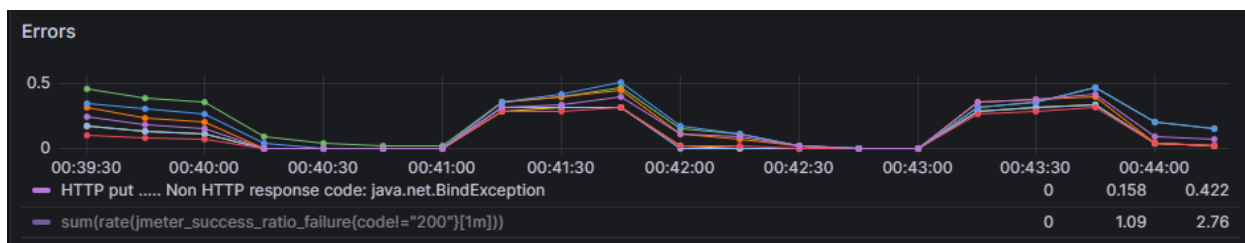
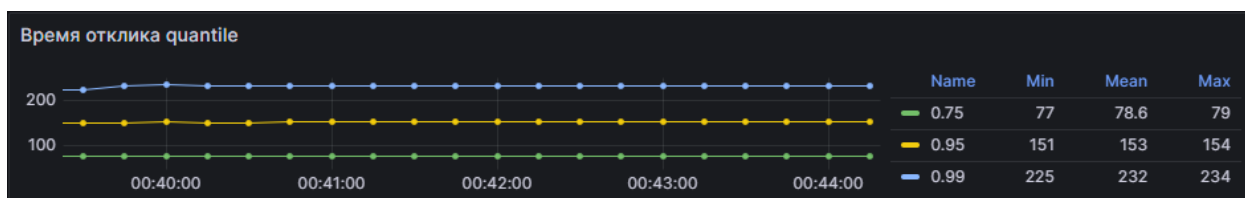
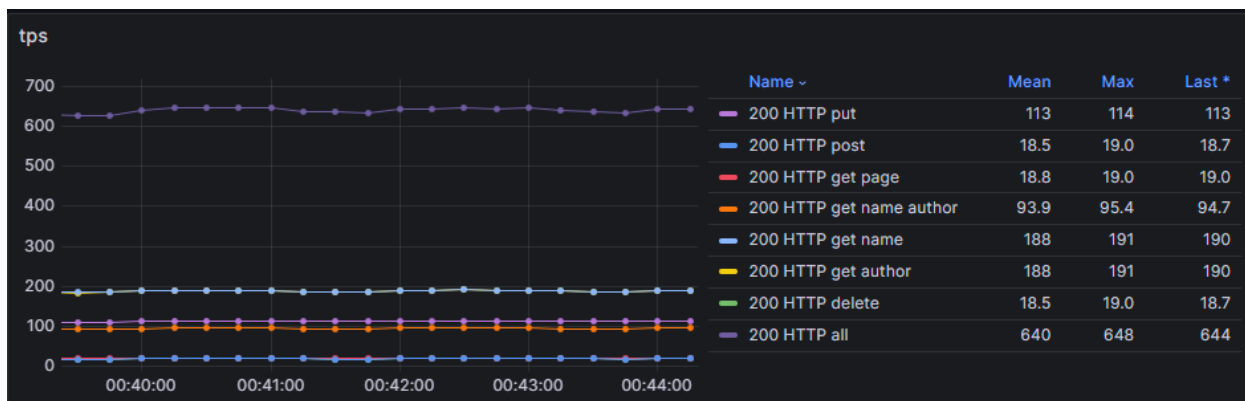
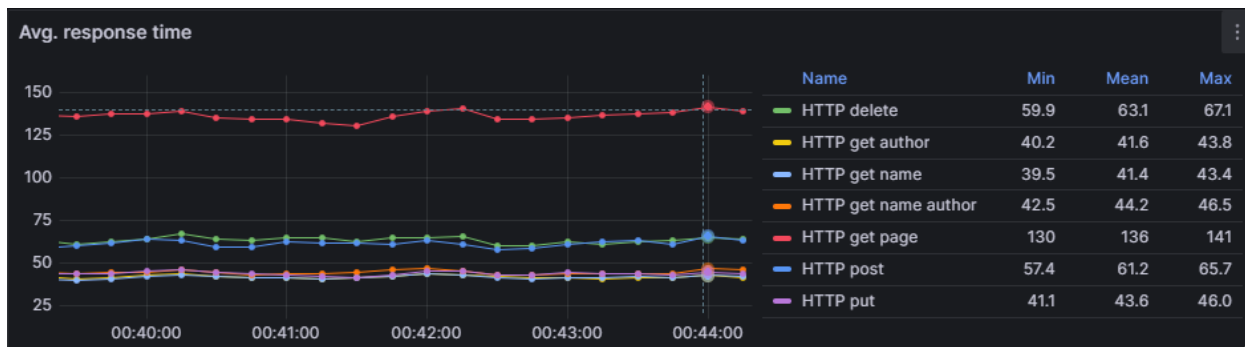
# Изменение количества потоков и коннектов (Threads-5 Maximum pool size-3)

Threads-10

Maximum pool size-3

tps-N19

При увеличении количества потоков, уменьшается время ответа сервиса.

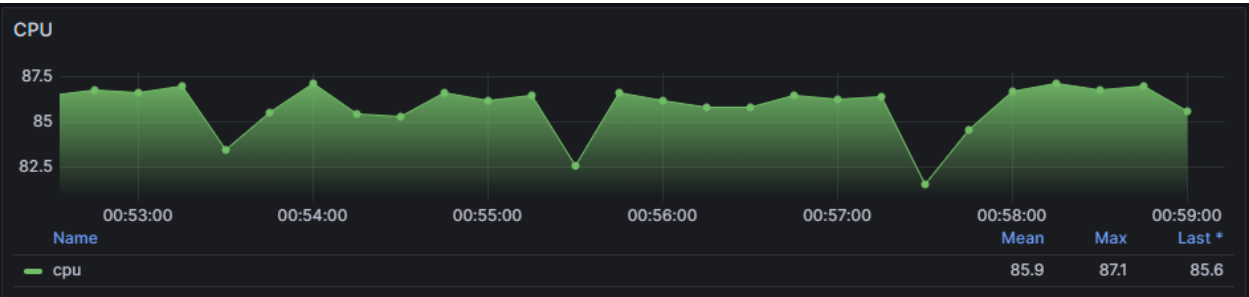
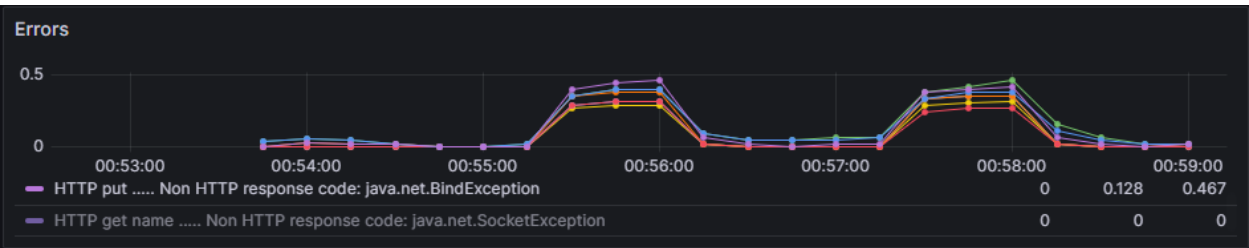
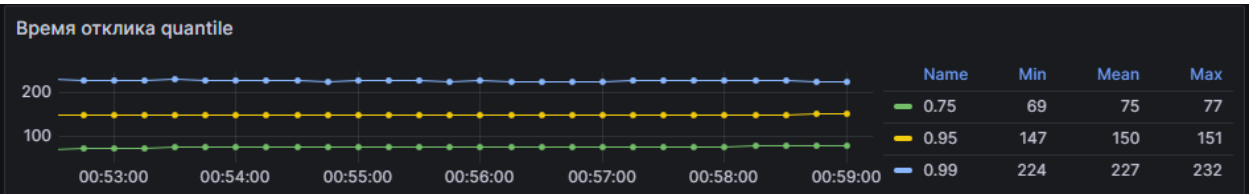
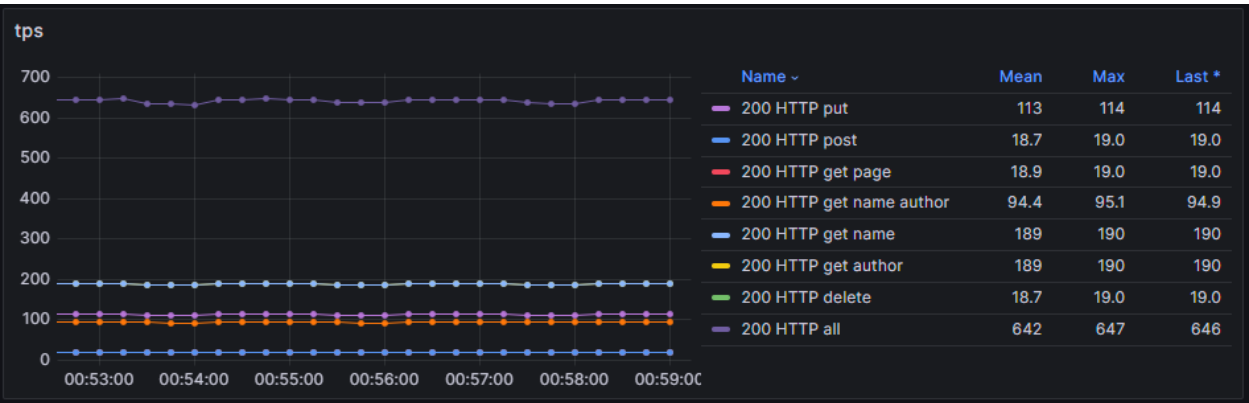
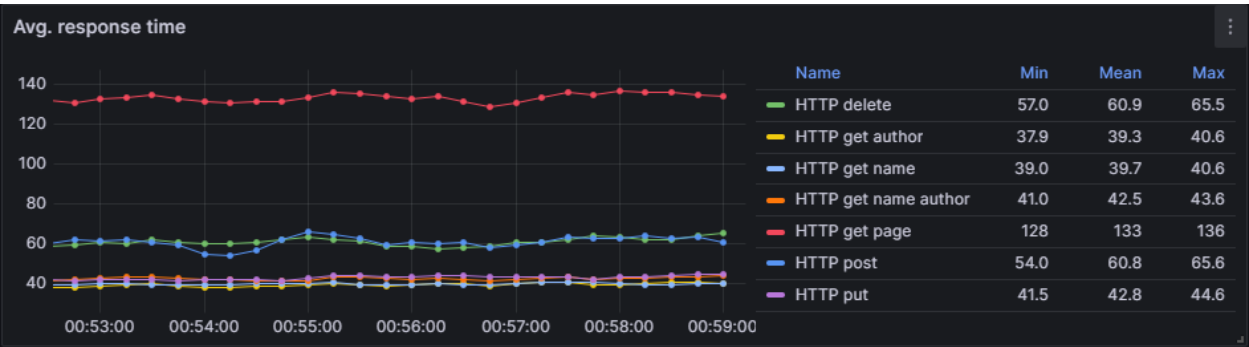


Threads-5

Maximum pool size-6

tps-N19

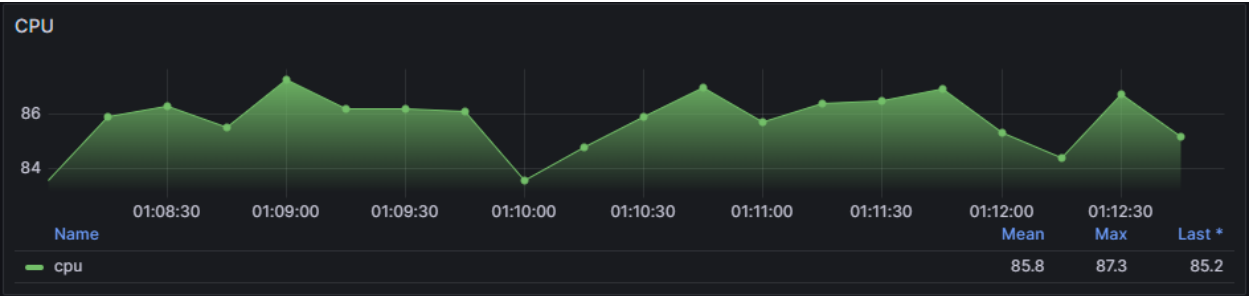
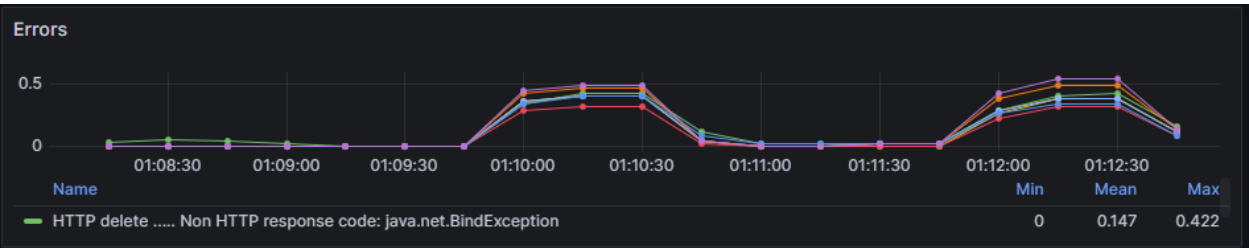
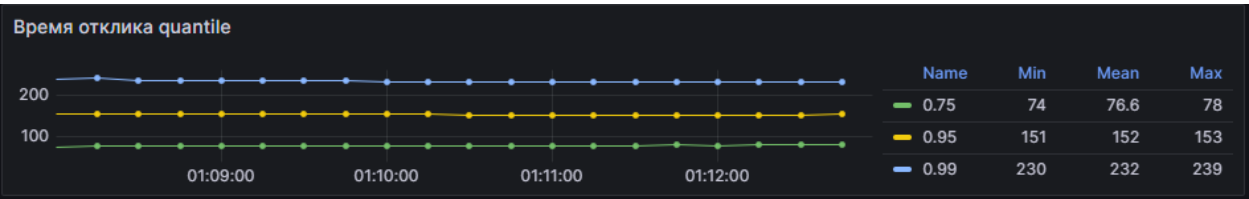
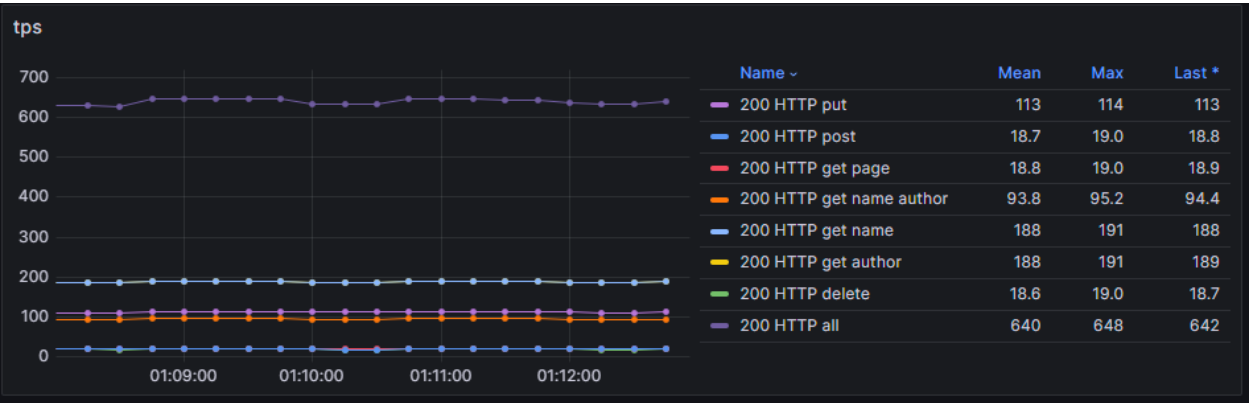
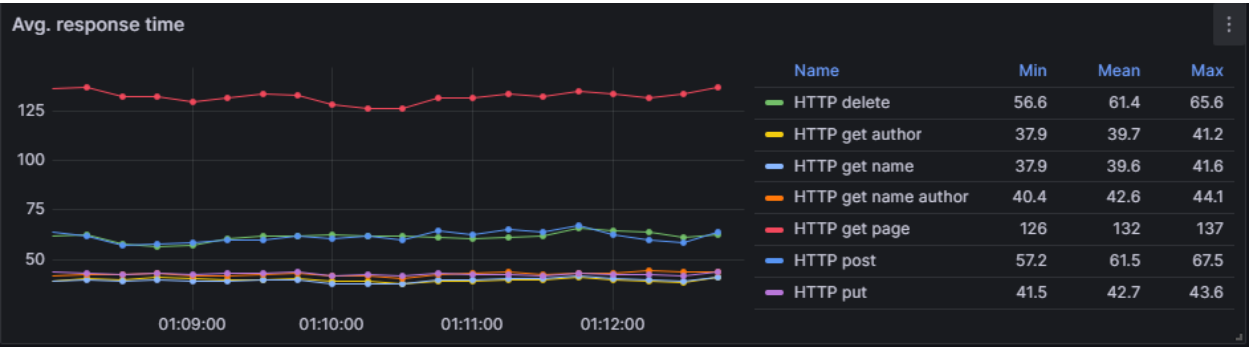
При увеличении количества коннектов, уменьшается время ответа сервиса.



Threads-20

Maximum pool size-12

tps-N19

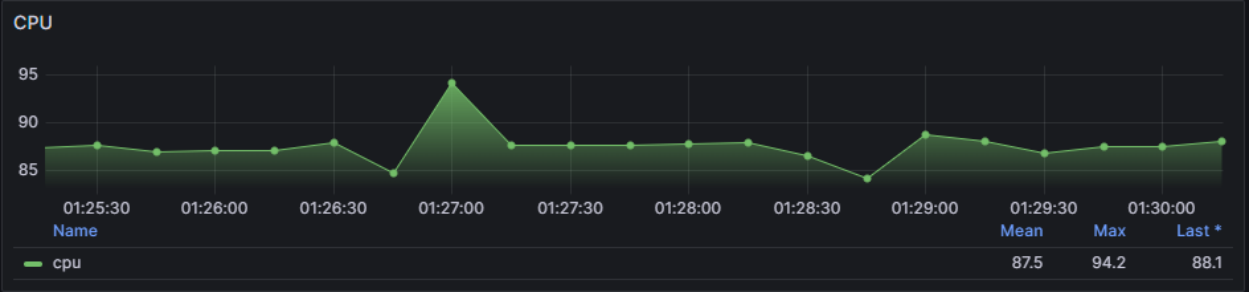
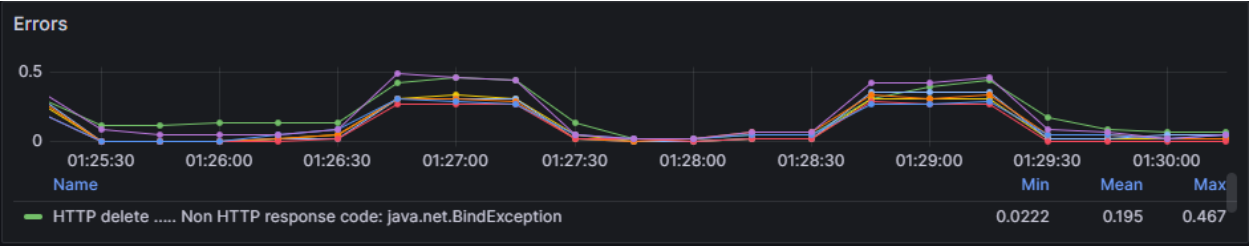
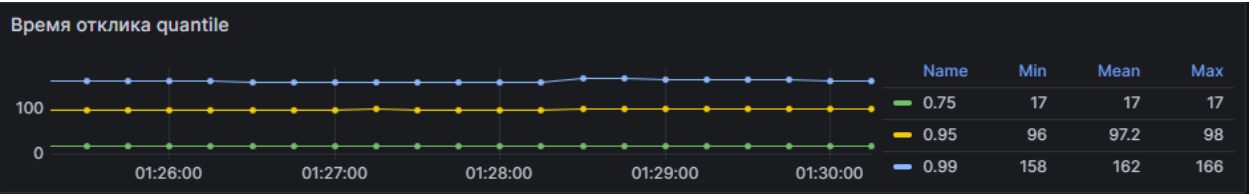
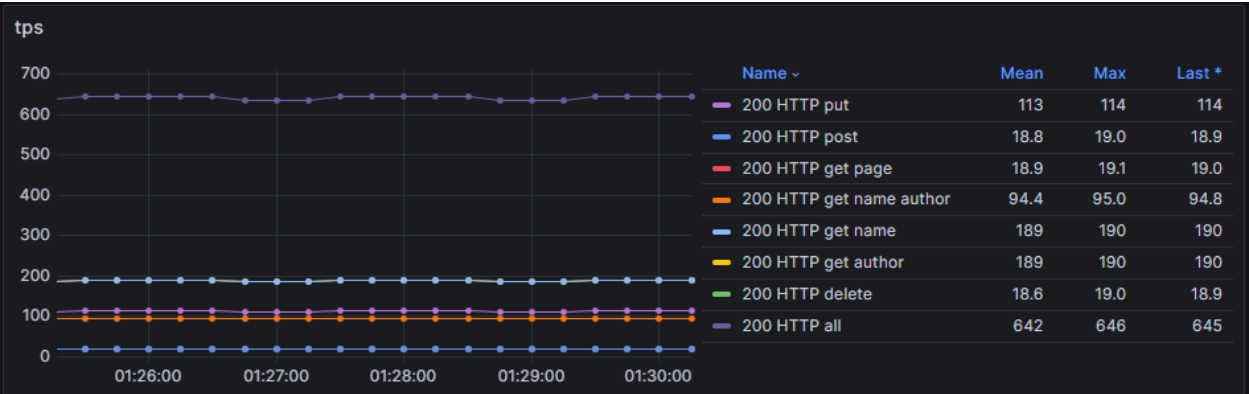
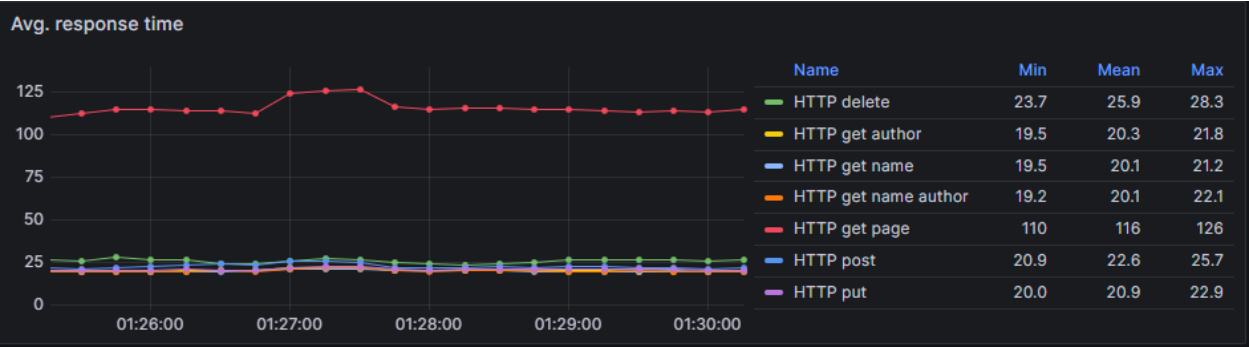




Threads-40

Maximum pool size-24

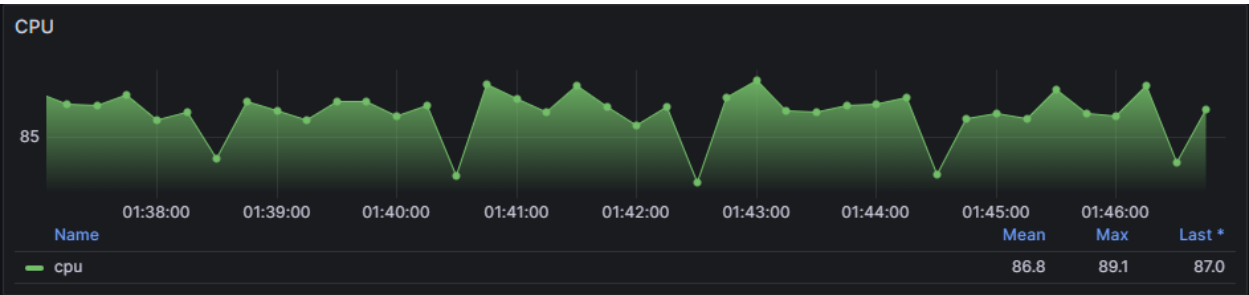
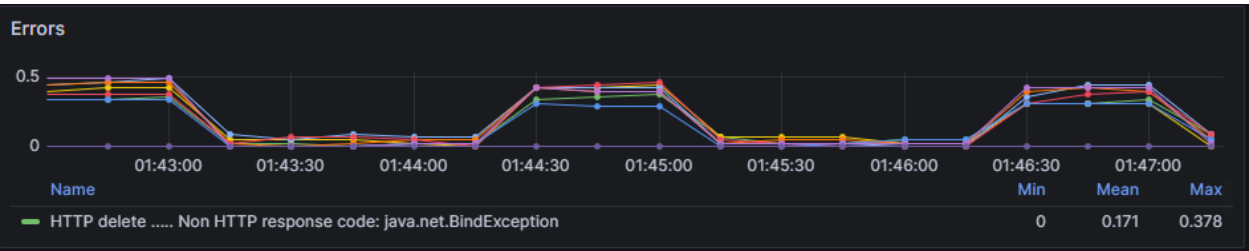
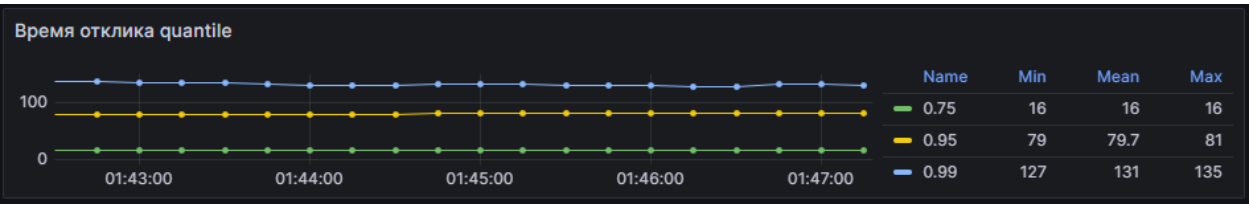
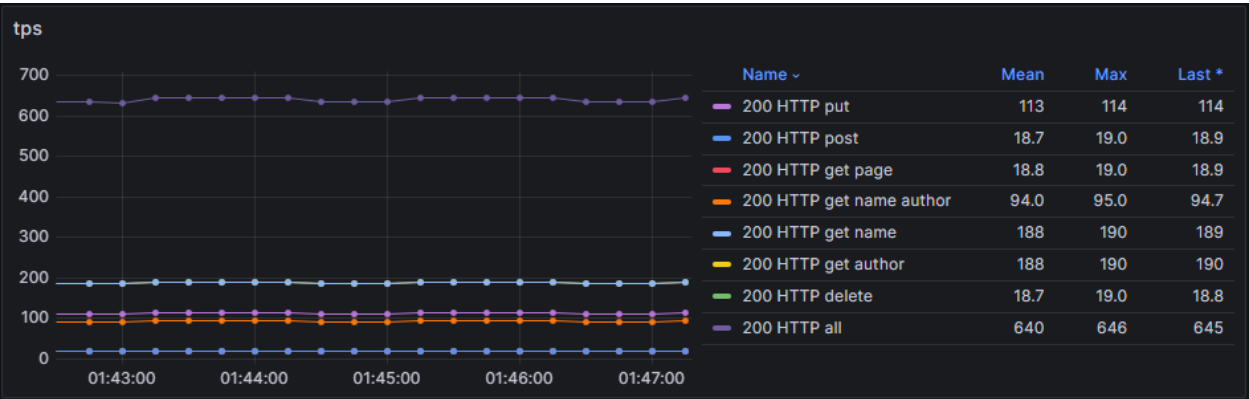
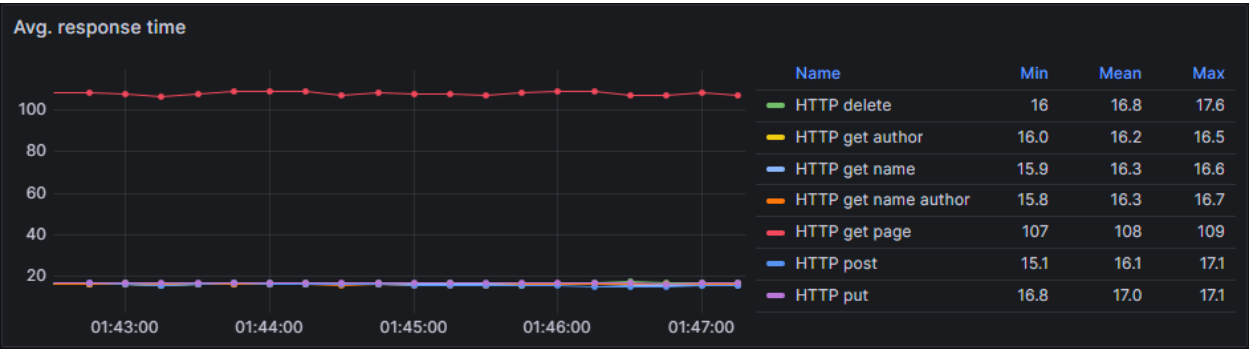
tps-N19



Threads-80

Maximum pool size-48

tps-N19

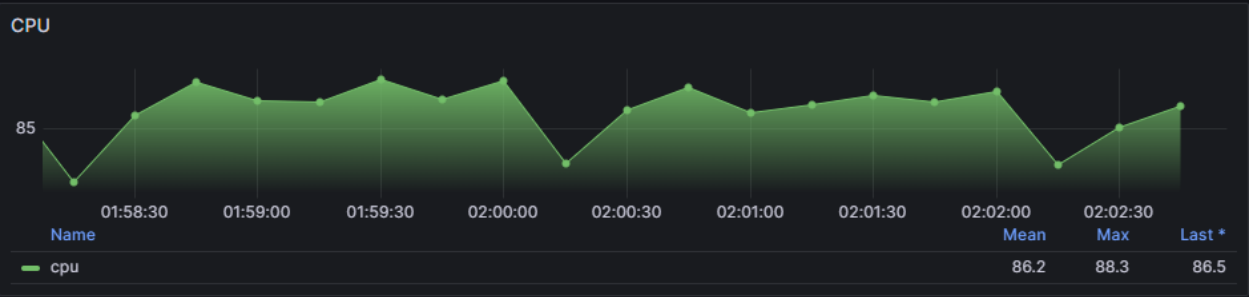
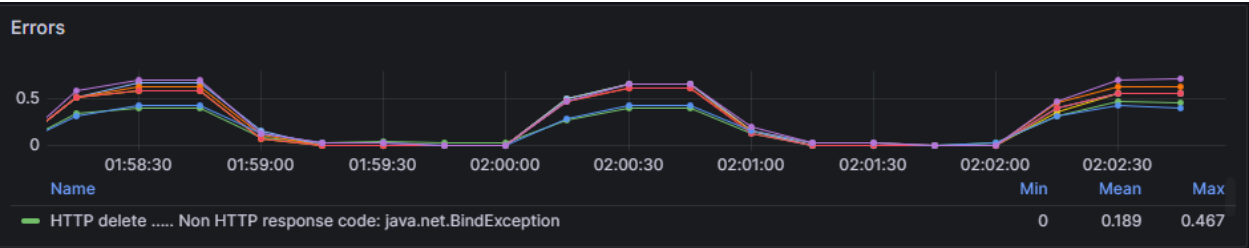
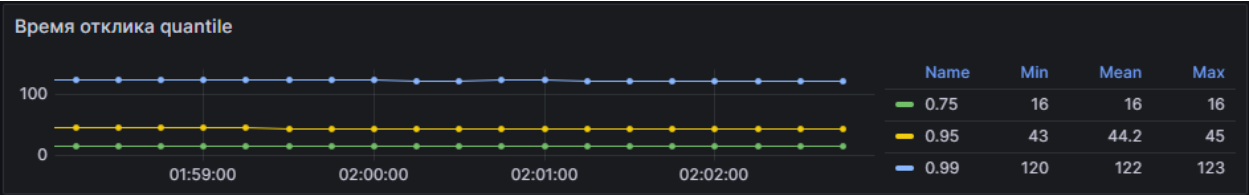
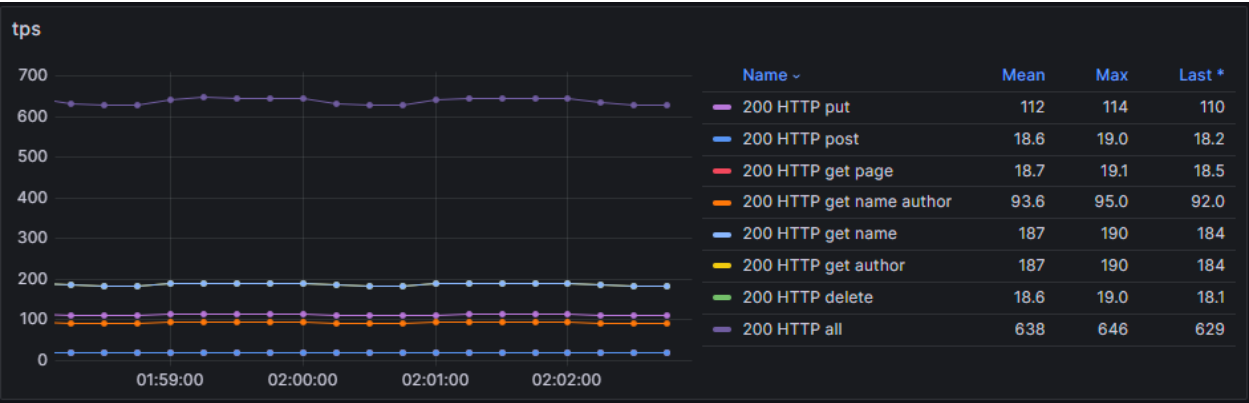
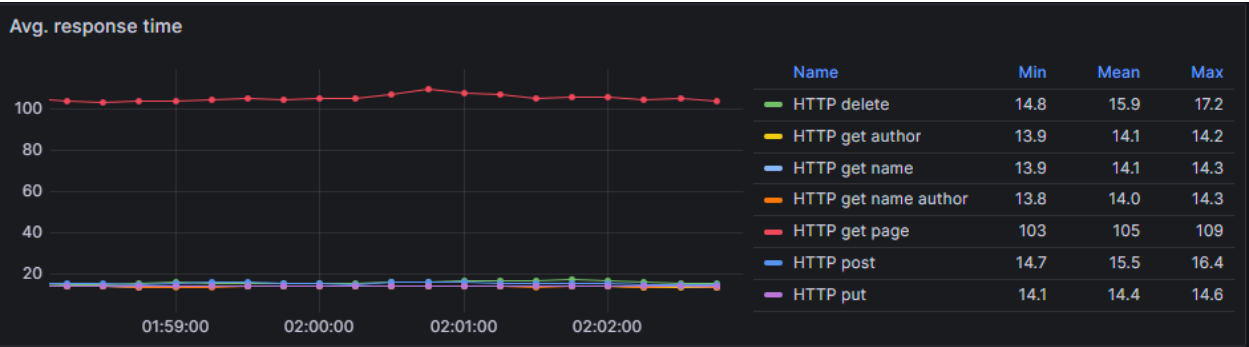


Threads-160

Maximum pool size-96

tps-N19

Рост незначителен.

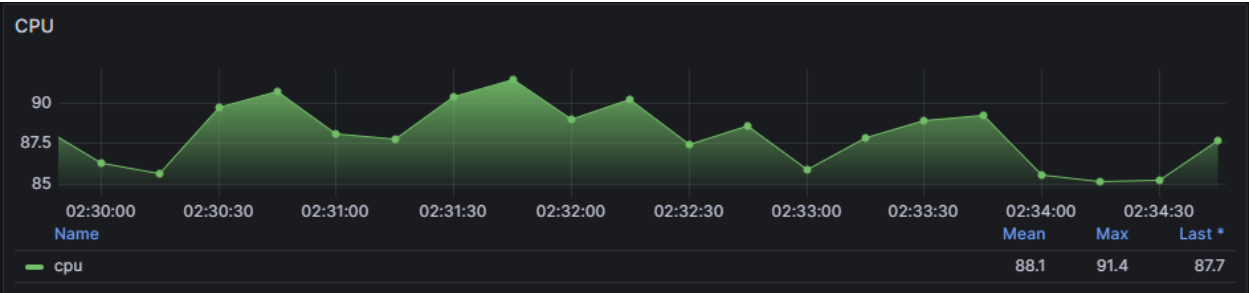
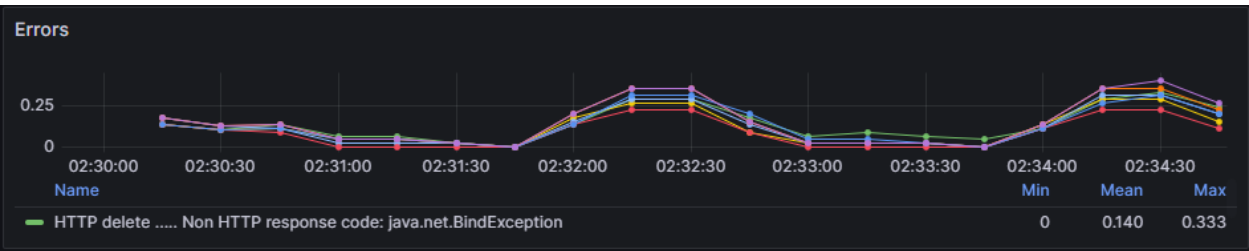
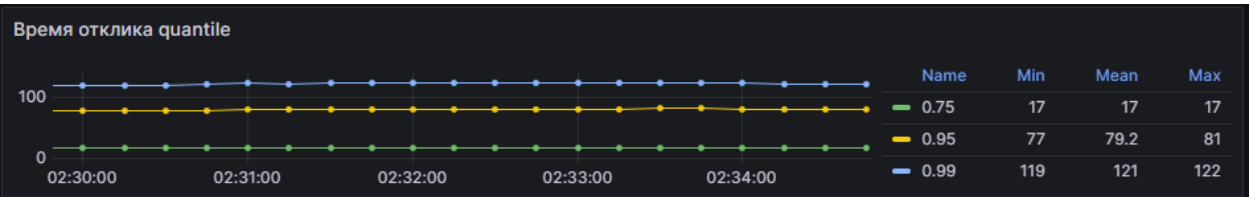
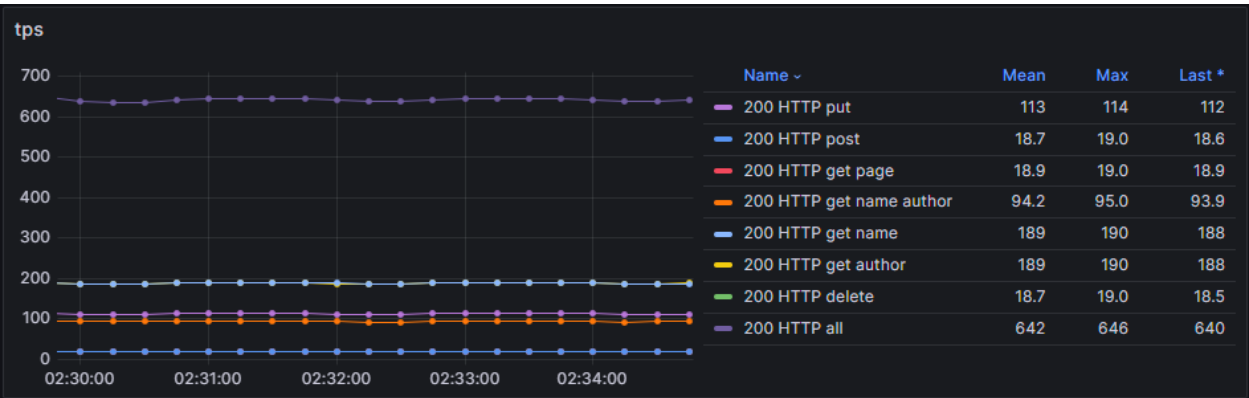
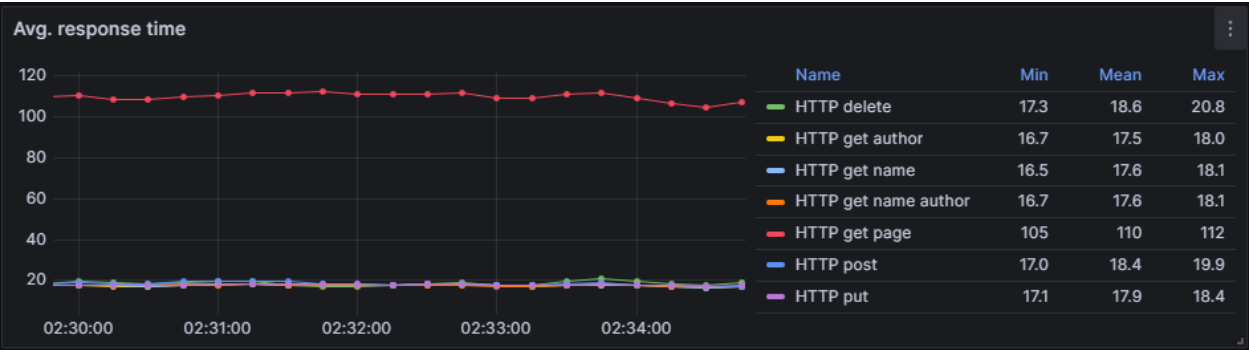


Threads-80

Maximum pool size-3

tps-N19

Примерное оптимальное количество потоков и коннектов.



Threads-5

Maximum pool size-48

tps-N19

Времена ответов заметно больше.

