

Получили результаты, что дальше?

Липин Григорий,
инженер по тестированию

1. Анализируем результаты

2. Планируем мощности

3. Реальные нагрузки

4. Как легче переживать
пиковые нагрузки

5. Live Demo

6. Запускаем тест

Анализируем результаты

- Смотрим графики
- Табличные данные
- Консольный вывод

Percentiles (all/last 1m/last), ms:

100.0% < 203.0 203.0 124.7

99.5% < 171.0 171.0 124.7

99.0% < 129.0 129.0 113.9

95.0% < 114.0 114.0 111.0

90.0% < 112.0 112.0 110.7

85.0% < 111.0 111.0 110.6

80.0% < 111.0 111.0 110.4

75.0% < 111.0 111.0 110.3

70.0% < 111.0 111.0 110.3

60.0% < 110.0 110.0 110.0

50.0% < 72.0 72.0 91.6

40.0% < 62.0 62.0 61.0

30.0% < 61.0 61.0 61.0

20.0% < 61.0 61.0 60.0

10.0% < 60.0 60.0 60.0

HTTP codes:

474 +100 100.00% : 200 OK

Net codes:

474 +100 100.00% : 0 Success

Average Sizes (all/last), bytes:

Request: 0.00 / 0.00

Response: 0.00 / 0.00

Average Times (all/last), ms:

Overall: 86.49 / 85.79

Connect: 0.00 / 0.00

Send: 0.00 / 0.00

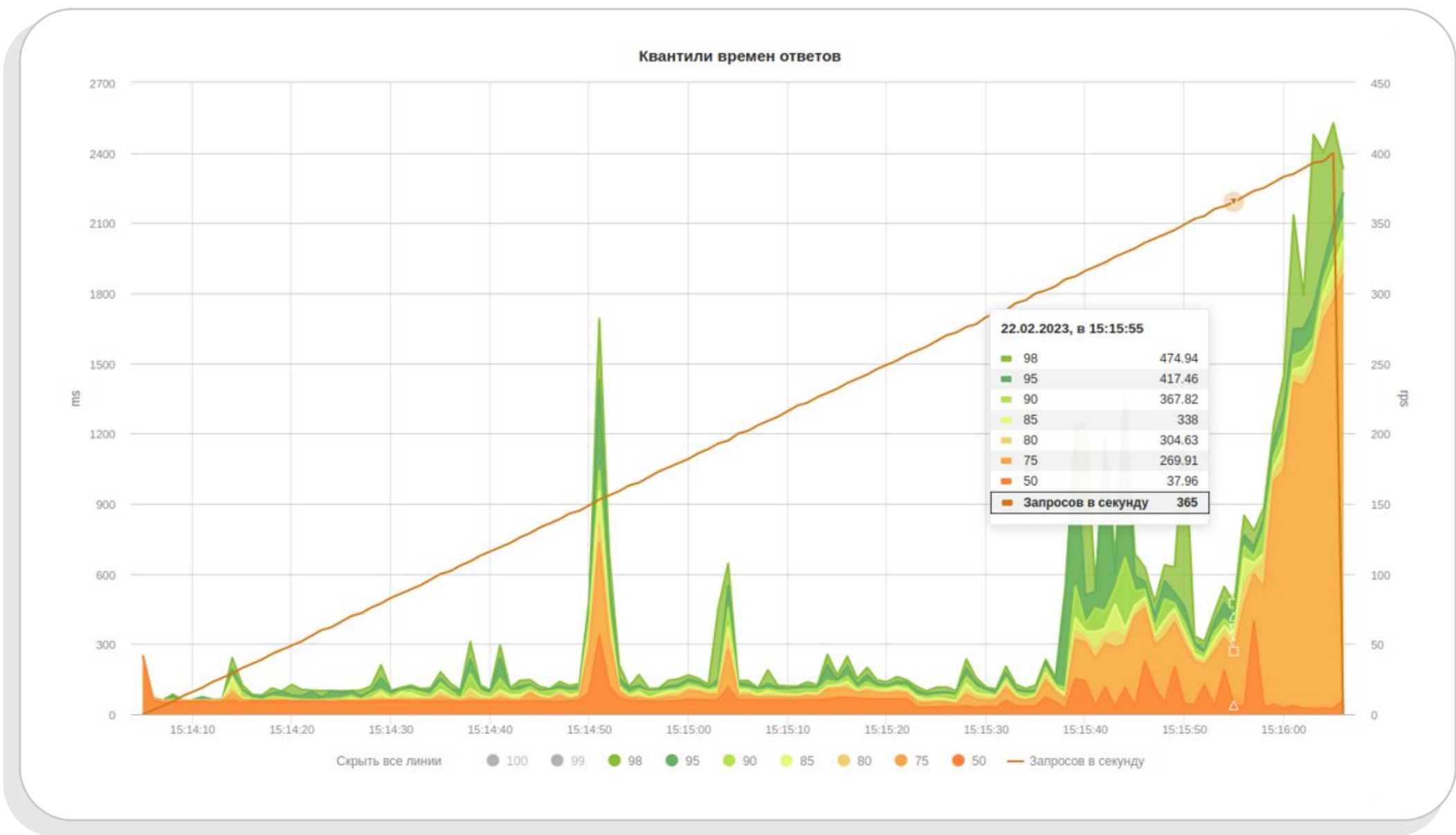
Latency: 0.00 / 0.00

Receive: 0.00 / 0.00

Распределение HTTP кодов		
	Кол-во	Доля
0 - ???	2427	2.200
200 - OK	107901	97.800

Распределение сетевых кодов		
	Кол-во	Доля
0 - Success	107901	97.800
104 - Connection reset by peer	275	0.249
110 - Connection timed out	2152	1.951

Распределение времен ответа			
	Количество	Доля	Квантиль
11050	989	0.896	0.896
11000	3	0.003	0.899
10900	4	0.004	0.903
10800	3	0.003	0.905
10700	3	0.003	0.908
10600	2	0.002	0.910
10500	2	0.002	0.912
10400	2	0.002	0.914



Анализируем результаты

На что смотрим?

- Времена ответа
- Ошибки
- Скорость
- Стабильность

Анализируем результаты

На что смотрим?

- Времена ответа
- Ошибки
- Скорость
- Стабильность

С чем сравниваем?

- Времена ответа
- Ошибки
- Скорость
- Стабильность

Анализируем результаты

На что смотрим?

- Времена ответа
- Ошибки
- Скорость
- Стабильность

С чем сравниваем?

- Времена ответа
- Ошибки
- Скорость
- Стабильность

Требования

- Времена ответа
- Ошибки
- Скорость
- Стабильность

1. Анализируем результаты

2. Планируем мощности

3. Реальные нагрузки

4. Как легче переживать
пиковые нагрузки

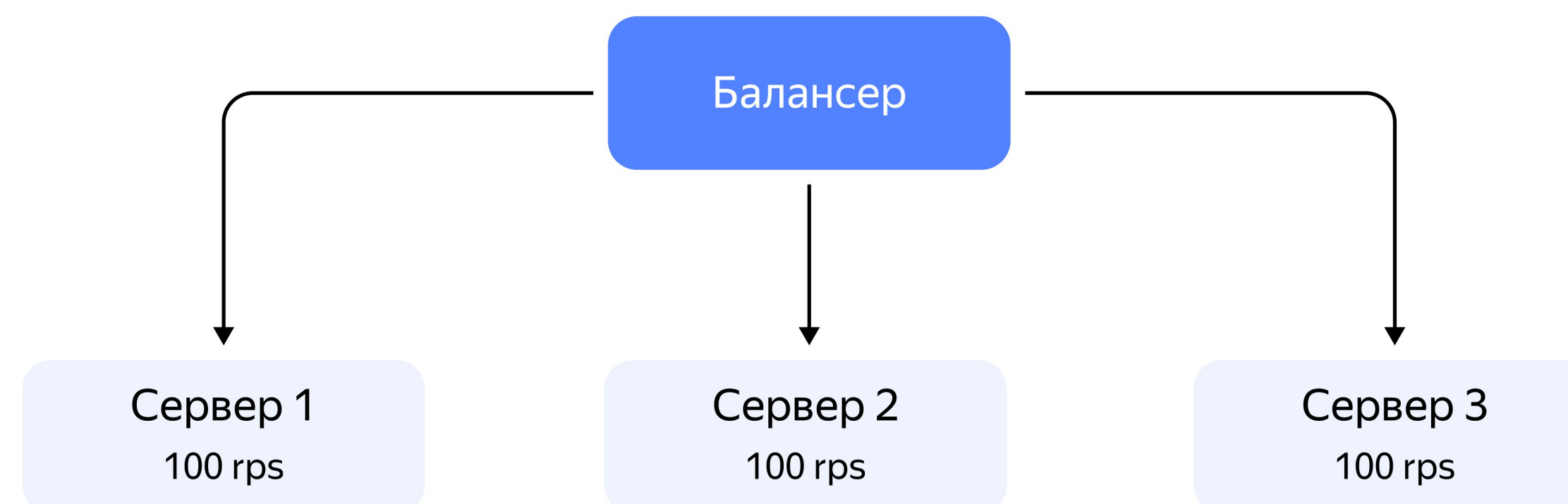
5. Live Demo

6. Запускаем тест

Планируем мощности

Тестовый сервис

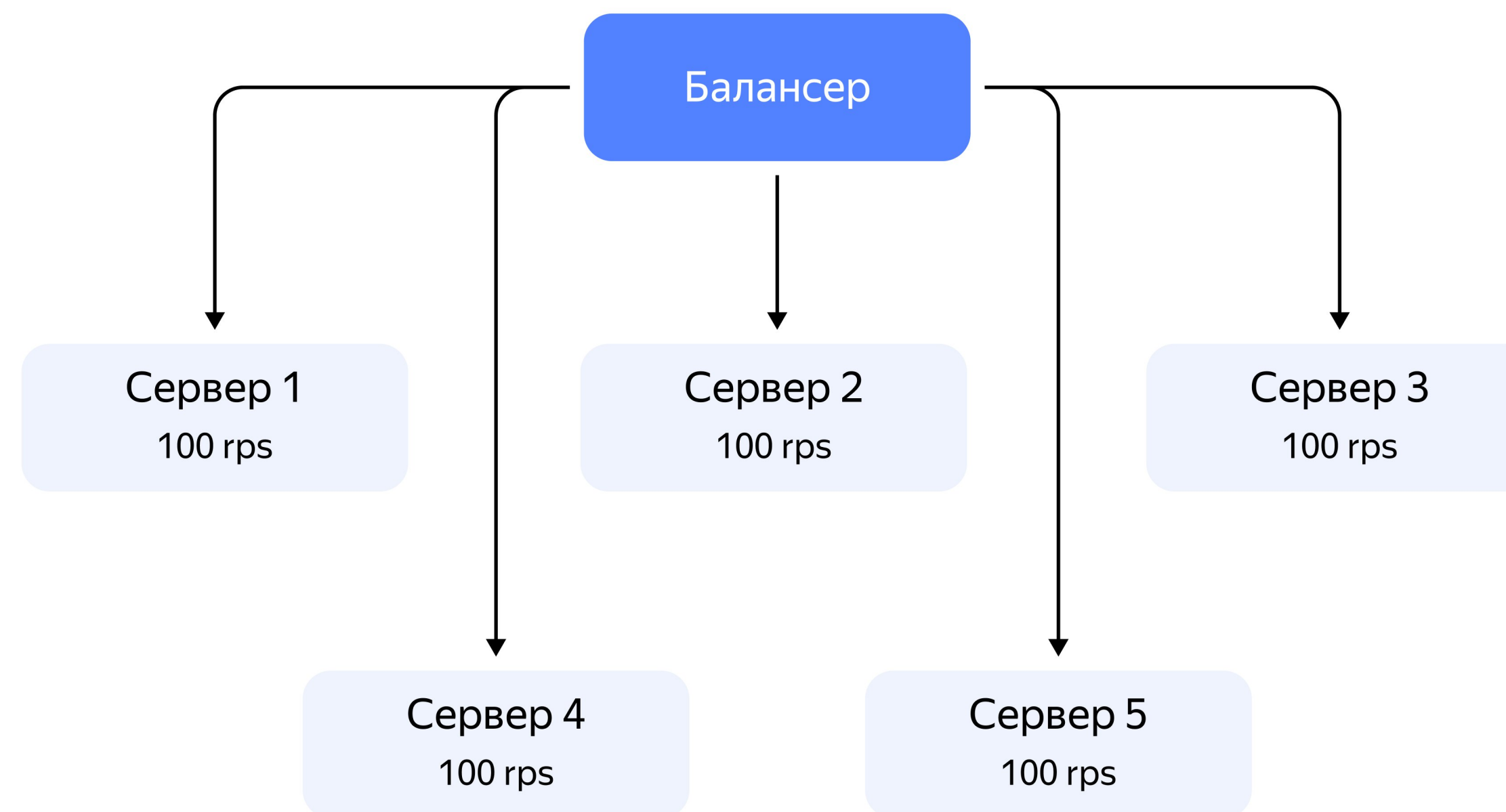
- 100 rps
- Ожидаем 300 rps



Планируем мощности

Тестовый сервис

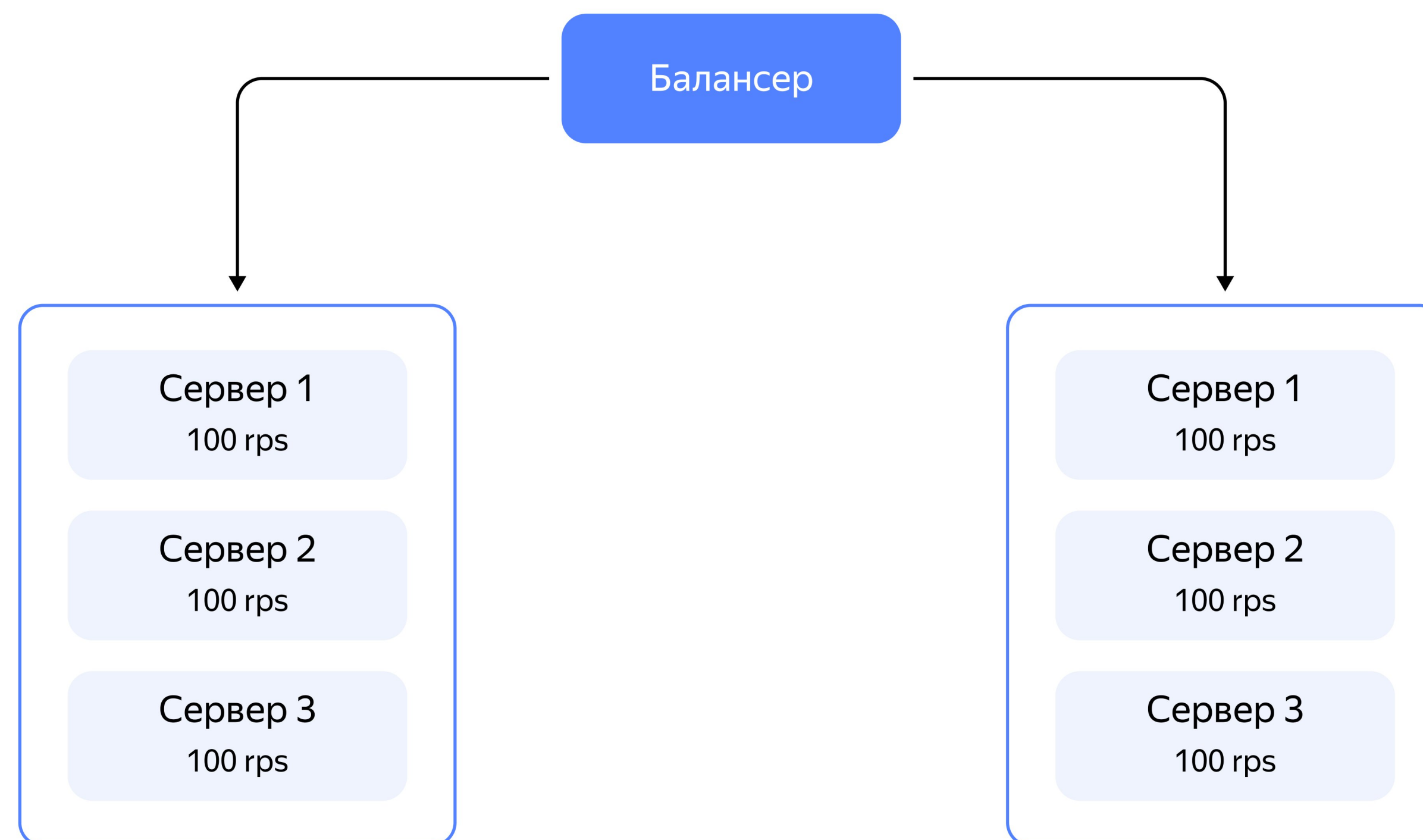
- 100 rps
- Ожидаем 300 rps
- Кластер — 500 rps
- Загружен на 60%
- Запас 40%



Планируем мощности

В условиях 2 ДЦ

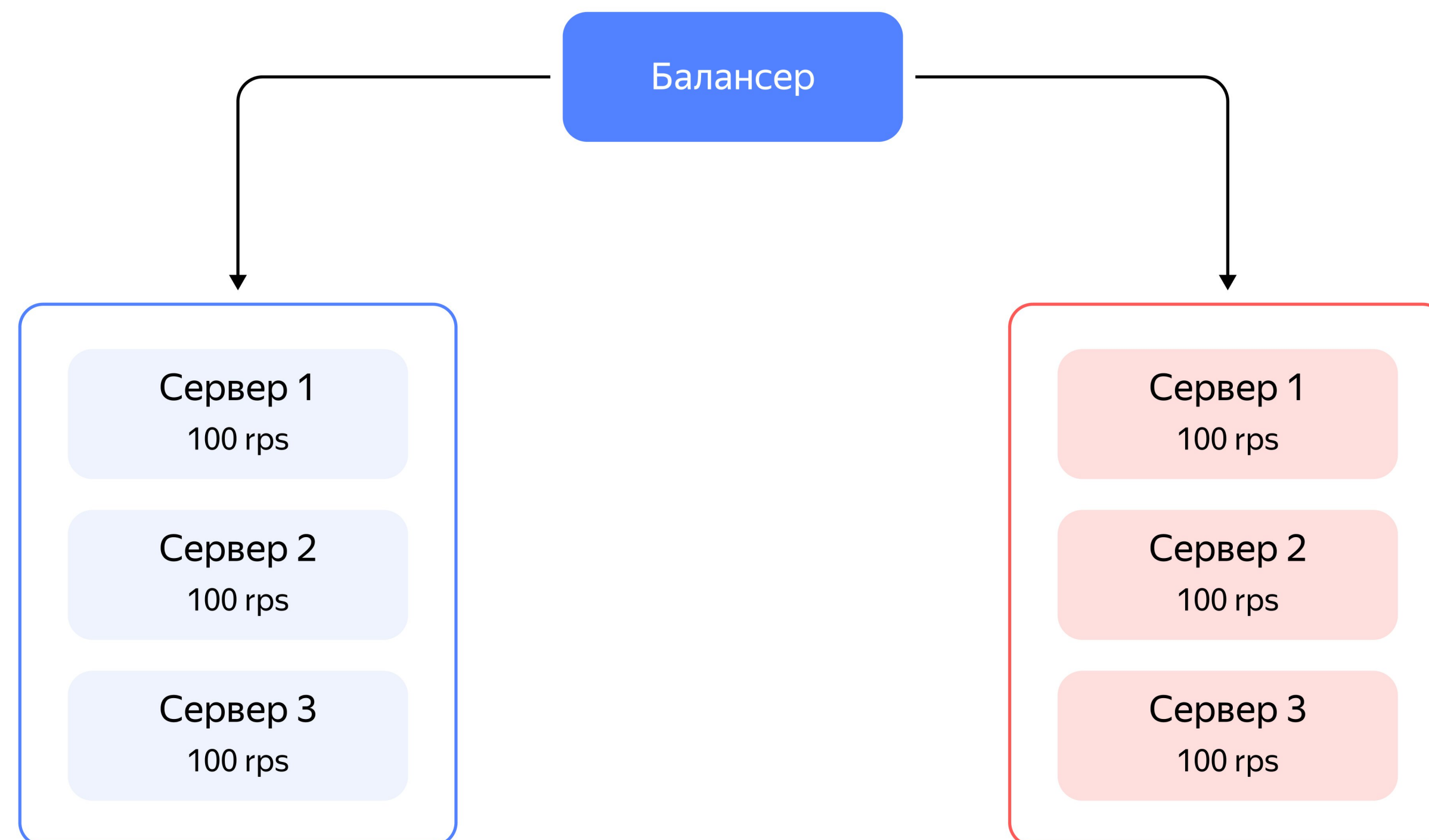
- 100 rps
- Ожидаем 300 rps
- Кластер — 600 rps
- Загружен на 50%
- Запас 50%



Планируем мощности

Отказ одного из ДЦ

- 100 rps
- Ожидаем 300 rps
- Кластер — 300 rps
- Загружен на 100%
- Запас 0%



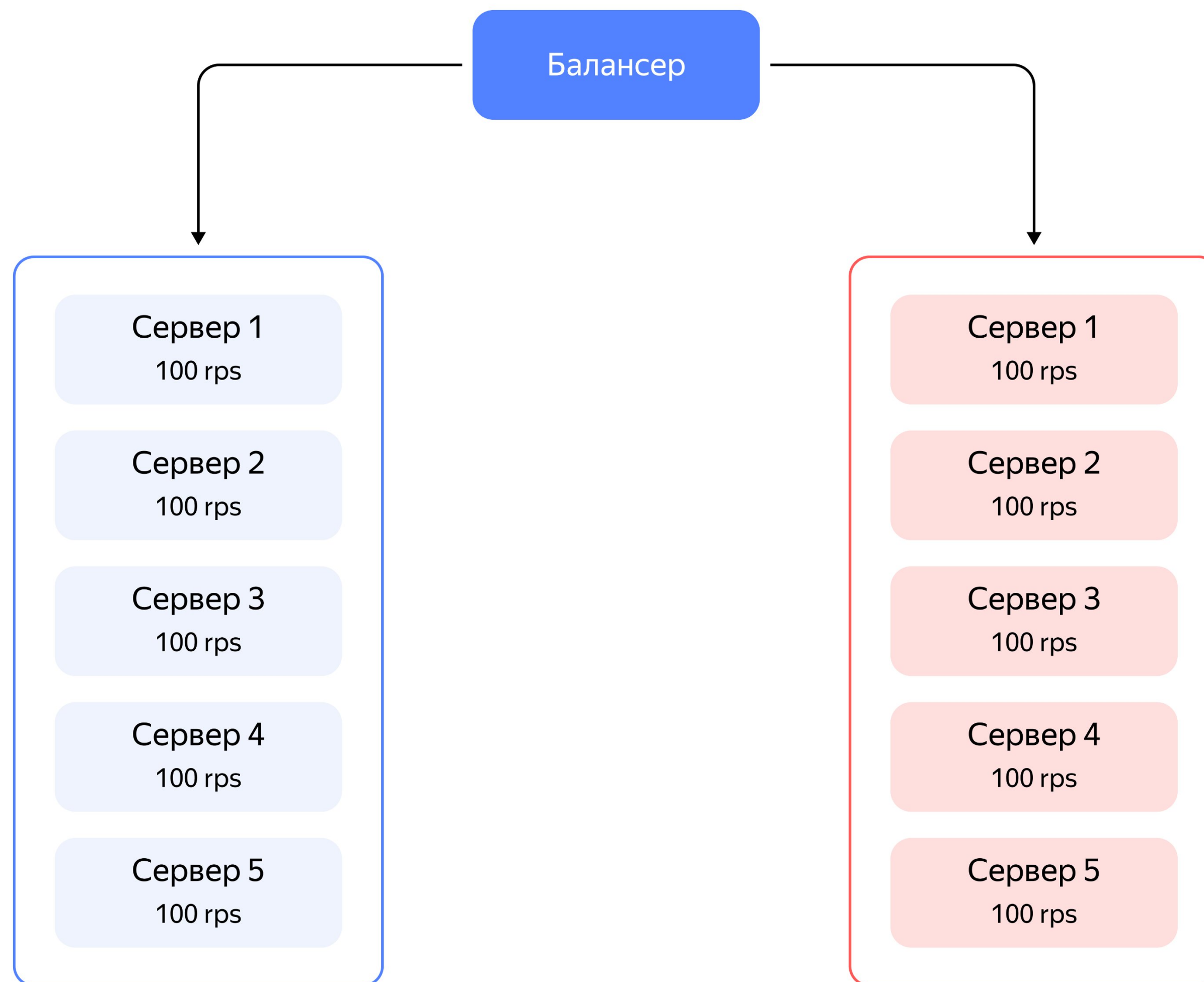
Планируем мощности

Отказ одного из ДЦ

- Один сервер — 100 rps
- Ожидаем 300 rps
- Кластер — 1000 rps
- Запас > x3

Отказ одного из ДЦ

- -1 ДЦ — 500 rps
- Загружен на 60%
- Запас 40%



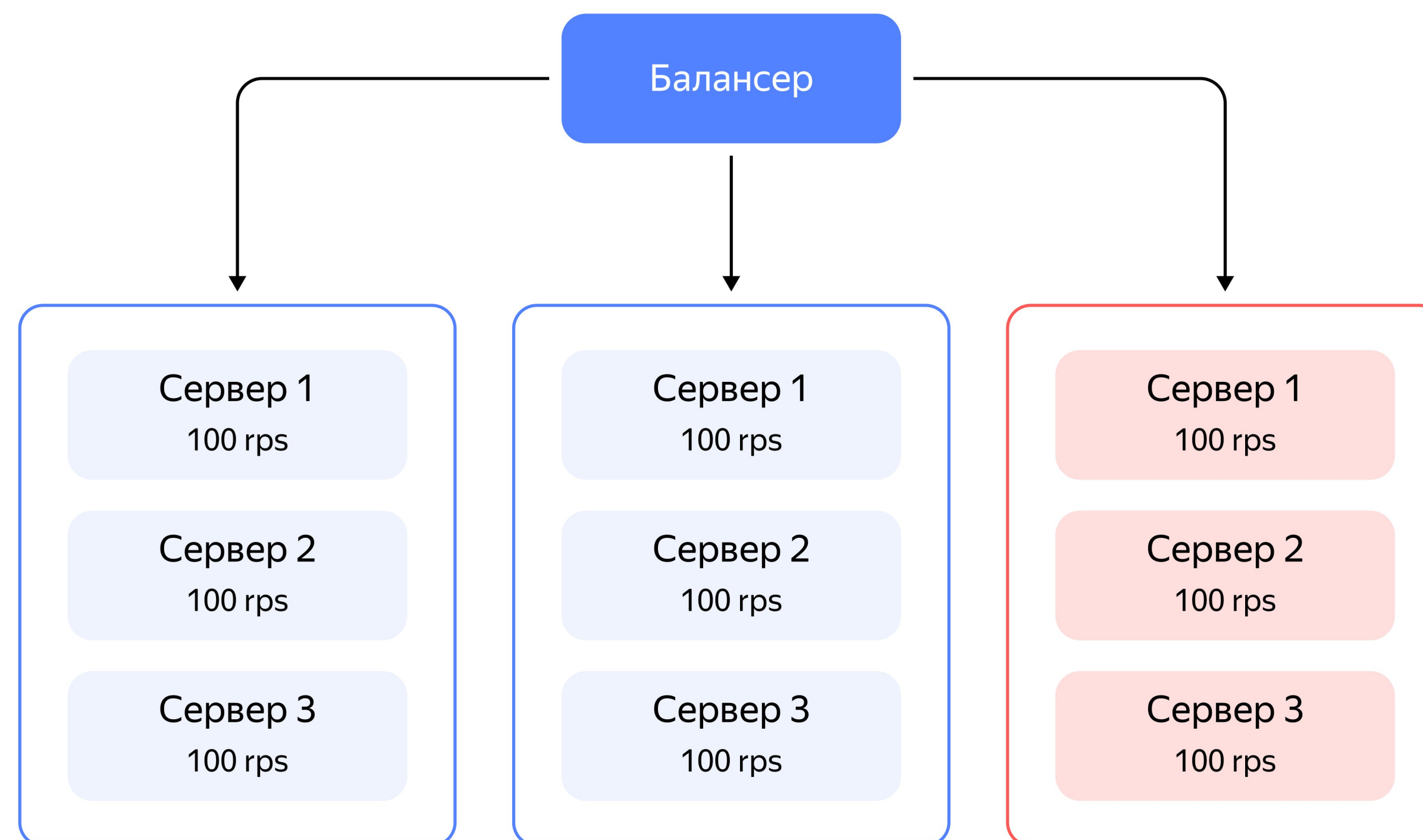
Планируем мощности

В условиях 3-х ДЦ

- Один сервер — 100 rps
- Ожидаем 300 rps
- Кластер — 900 rps
- Загружен на 33%
- Запас = x3

Отказ одного из ДЦ

- -1 ДЦ — 500 rps
- Загружен на 60%
- Запас 40%



1. Анализируем результаты

2. Планируем мощности

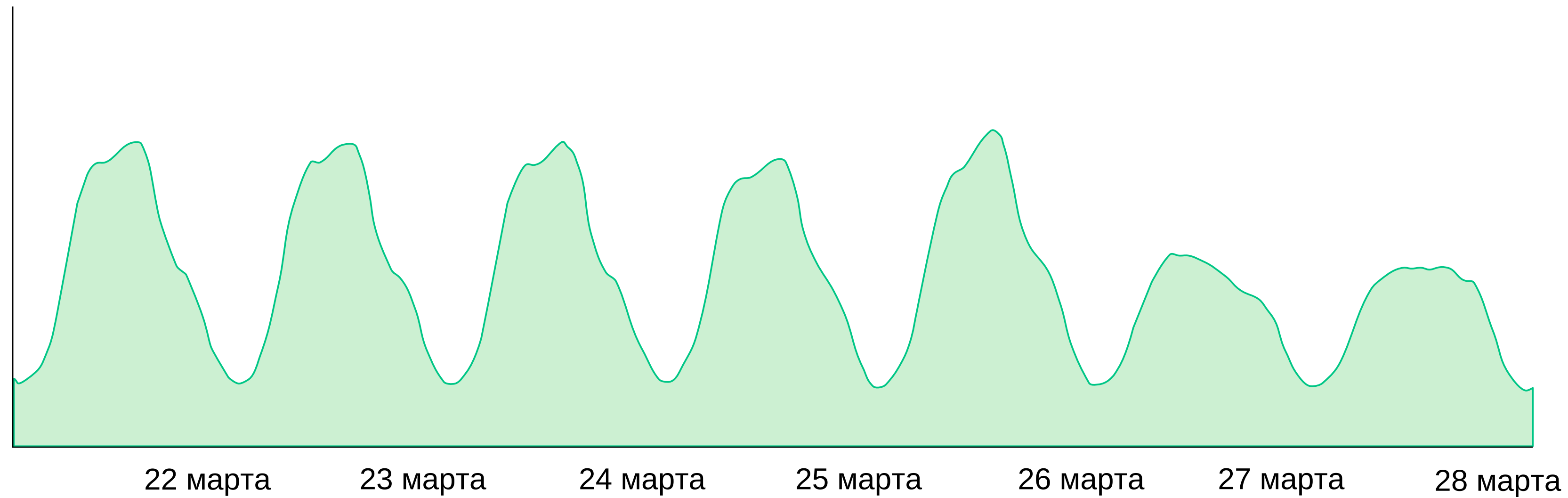
3. Реальные нагрузки

4. Как легче переживать
пиковые нагрузки

5. Live Demo

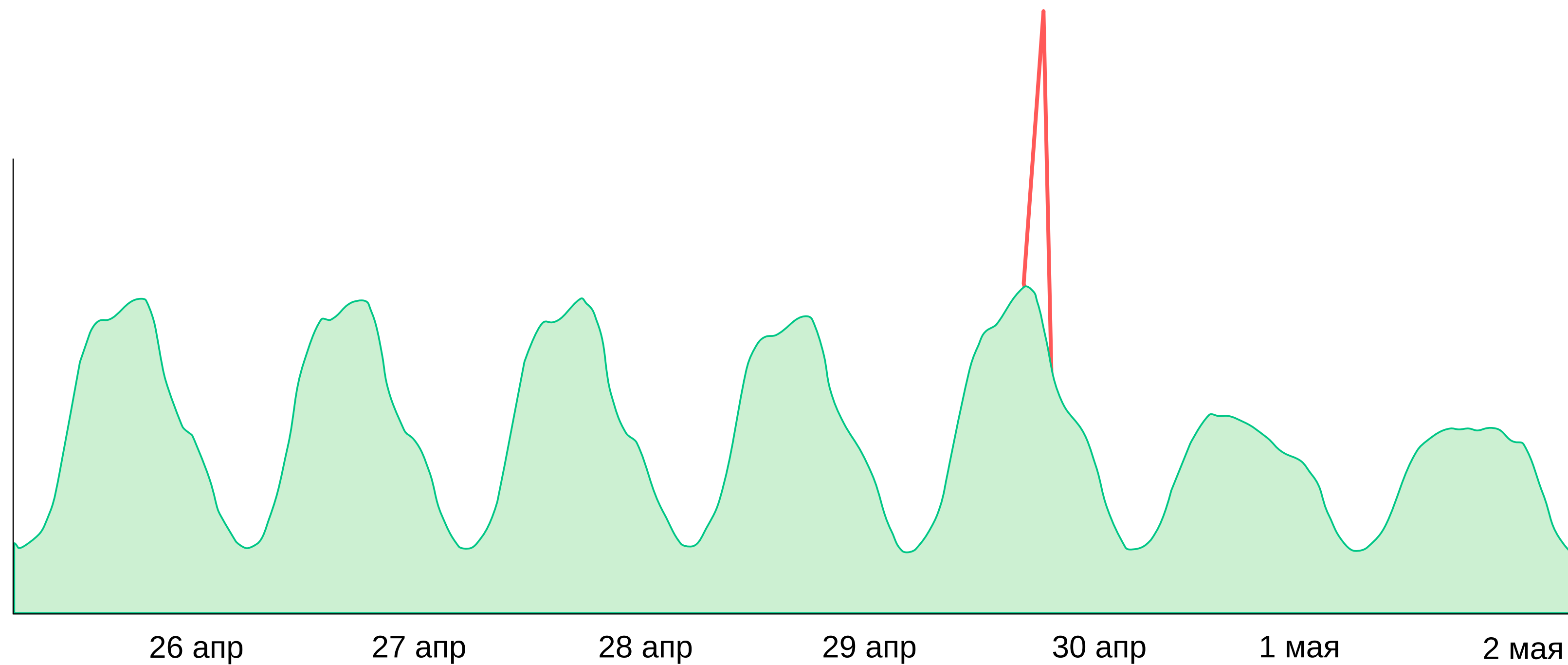
6. Запускаем тест

Реальные нагрузки



Сервис
Яндекс Карты

Реальные нагрузки



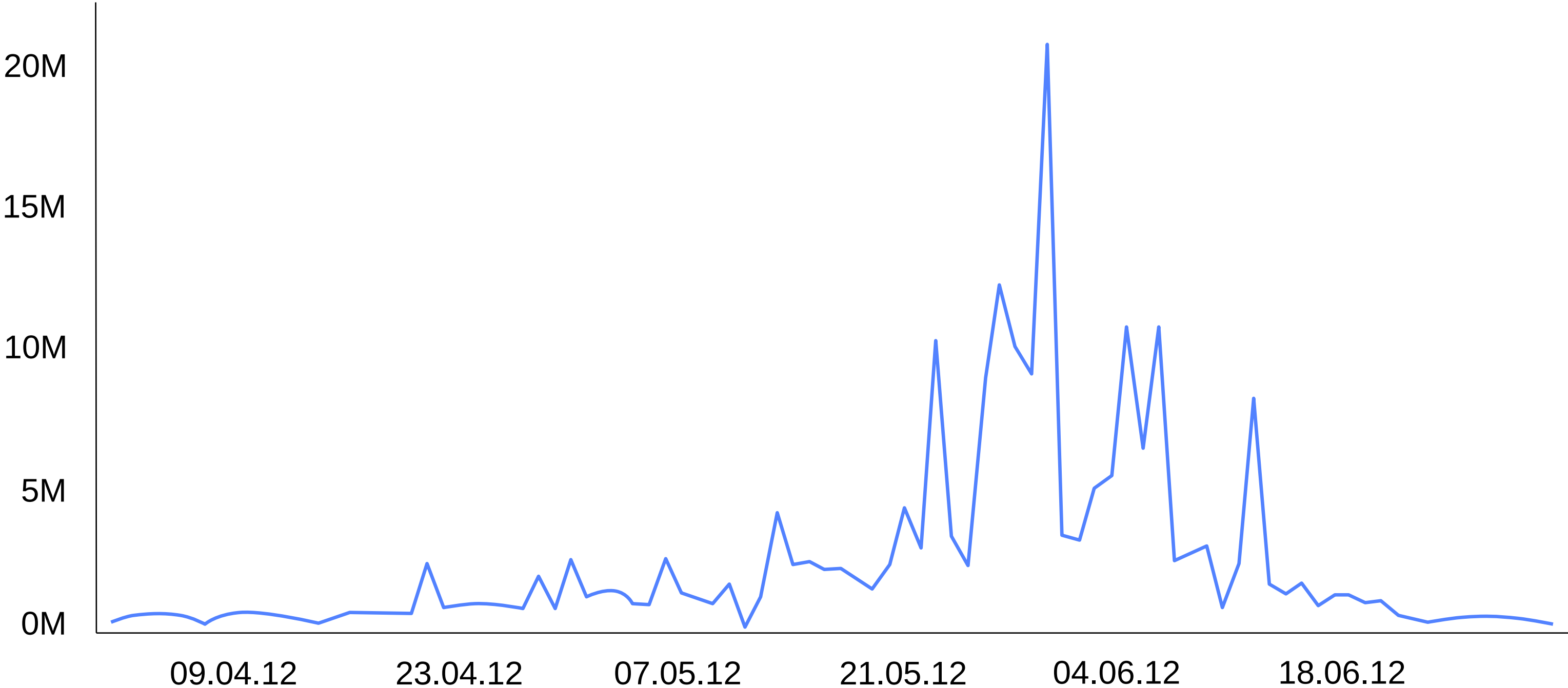
Сервис
Яндекс Карты

×5

От обычной

Реальные нагрузки

Уточненные хиты



Сервис ЕГЭ

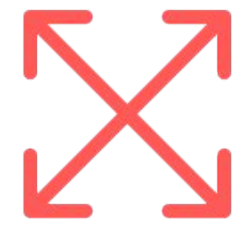
×20

от обычной

1. Анализируем результаты
2. Планируем мощности
3. Реальные нагрузки
4. Как легче переживать
пиковые нагрузки
5. Live Demo
6. Запускаем тест

Жизнь до облаков

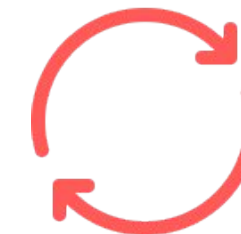
Какие есть минусы



Ручное
масштабирование



Долгое расширение
кластера



Низкая
утилизация



Дорого

Жизнь в облаках

Какие есть минусы



Возможность
автомасштабирования

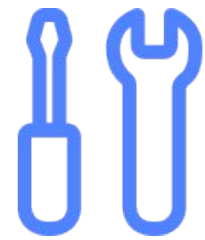


Быстрее расширить
кластер



Более эффективная
утилизация ресурсов

Жизнь в облаках



Удобные инструменты
для работы с
инфраструктурой

Web UI

CLI

API



Сервис нагрузочного тестирования
Yandex Load Testing

Генератор нагрузки — Агенты

Отчеты с результатами теста

Хранит конфигурацию ваших тестов

Хранит тестовые данные в Object Storage(S3)

Мониторинг ресурсов

1. Анализируем результаты

2. Планируем мощности

3. Реальные нагрузки

4. Как легче переживать
пиковые нагрузки

5. Live Demo

6. Запускаем тест

Тестовое приложение

Сервис сортировки чисел

- Пузырьковая сортировка
- Post запрос с телом: {"numbers": [28917,6030,4760,24502,13025,28312]}
- Docker, Gunicorn + Flask приложение

1. Анализируем результаты
2. Планируем мощности
3. Реальные нагрузки
4. Как легче переживать пиковые нагрузки
5. Live Demo
6. Запускаем тест

Автомасштабируемая группа VM

Следит за целевой метрикой

- Average CPU
- Custom метрика

Тип масштабирования

- Зональный
- Региональный

Общие настройки

- Период стабилизации
- Время разогрева
- Промежуток измерения нагрузки
- Максимальный и минимальный размер группы

Выводы

1

Готовьтесь к пиковым
нагрузкам заранее

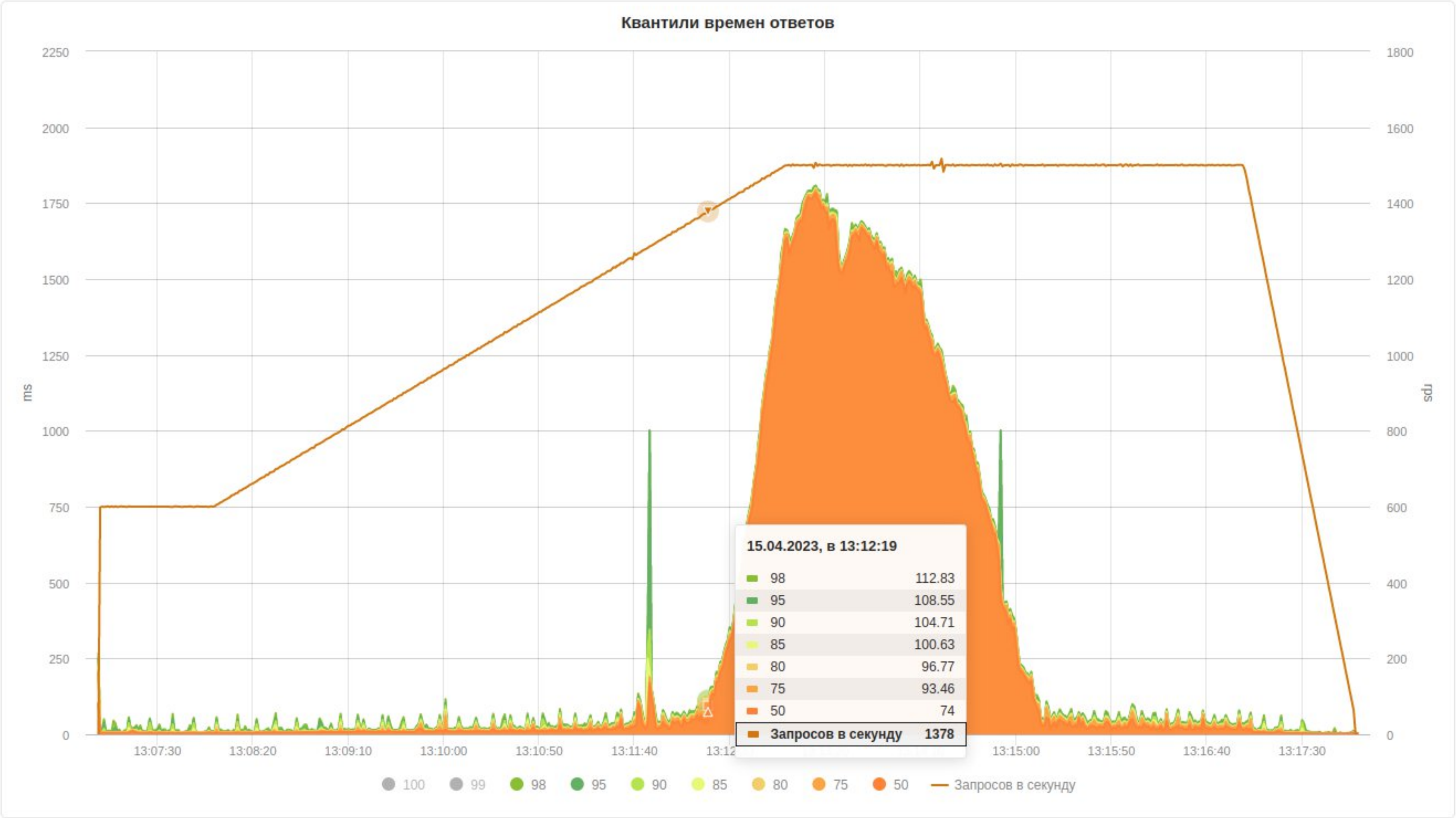
2

Продумайте ваш сценарий
действий

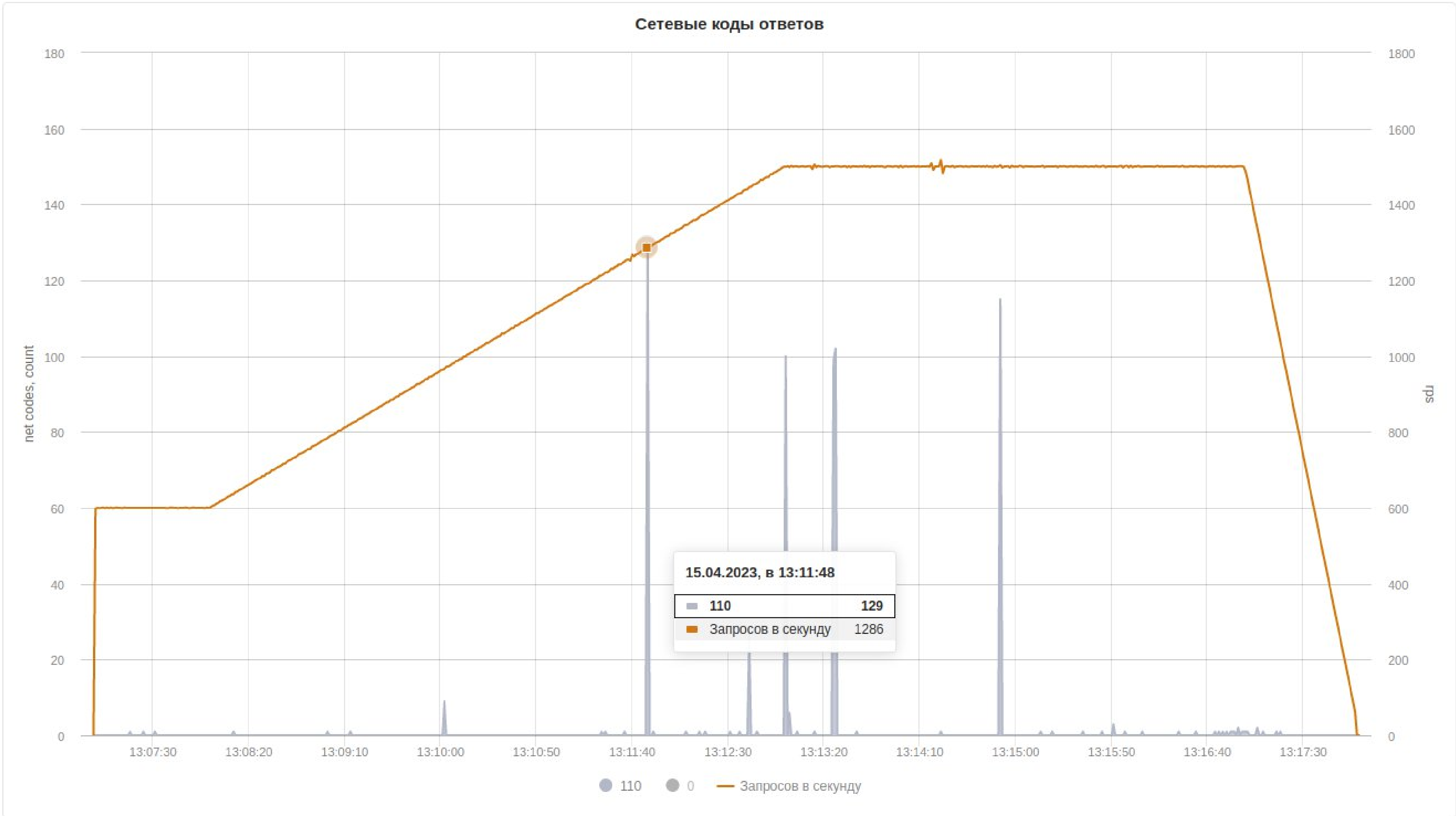
3

Выберете подходящее
для вас решение

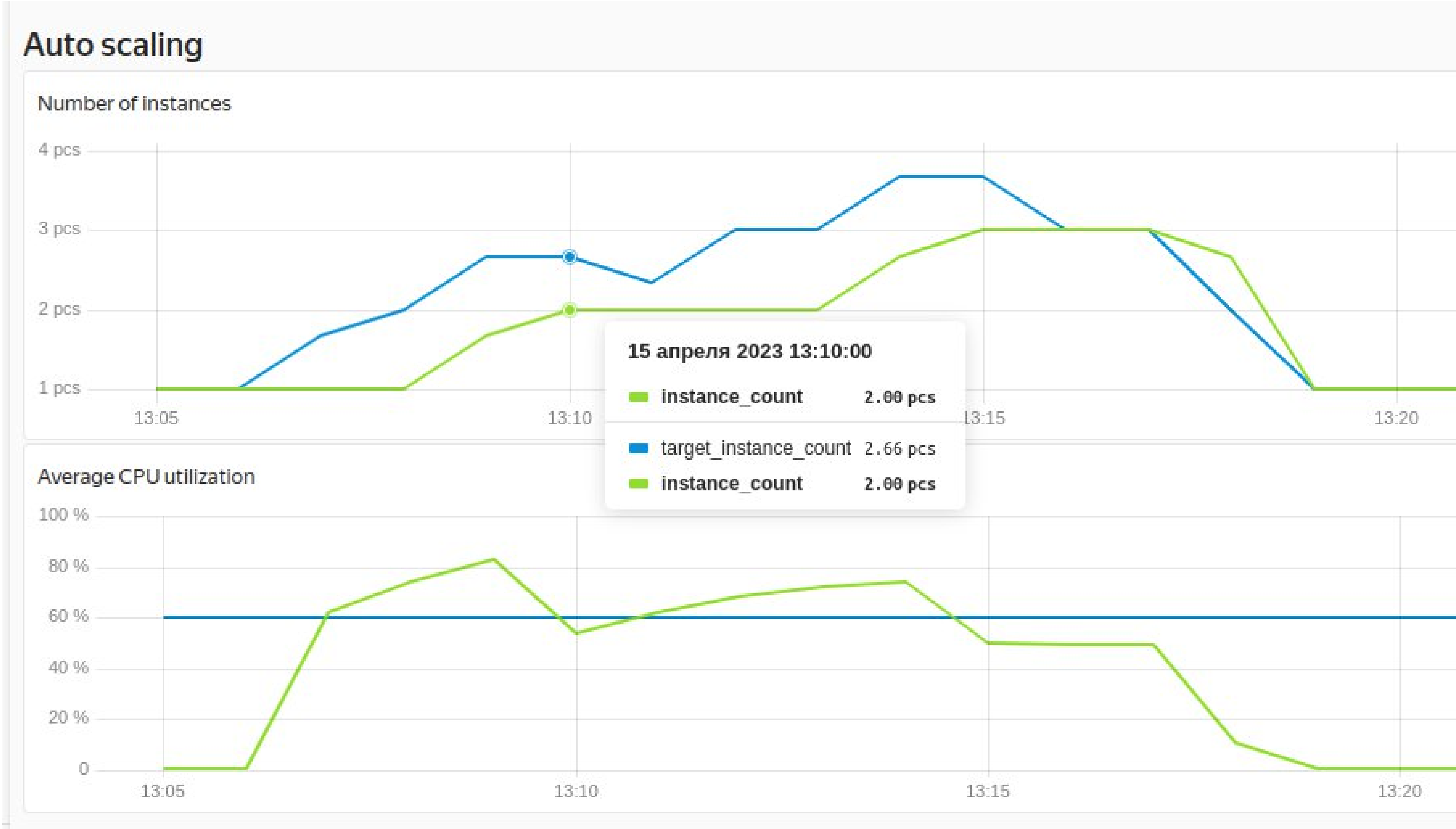
Демо - времена ответов



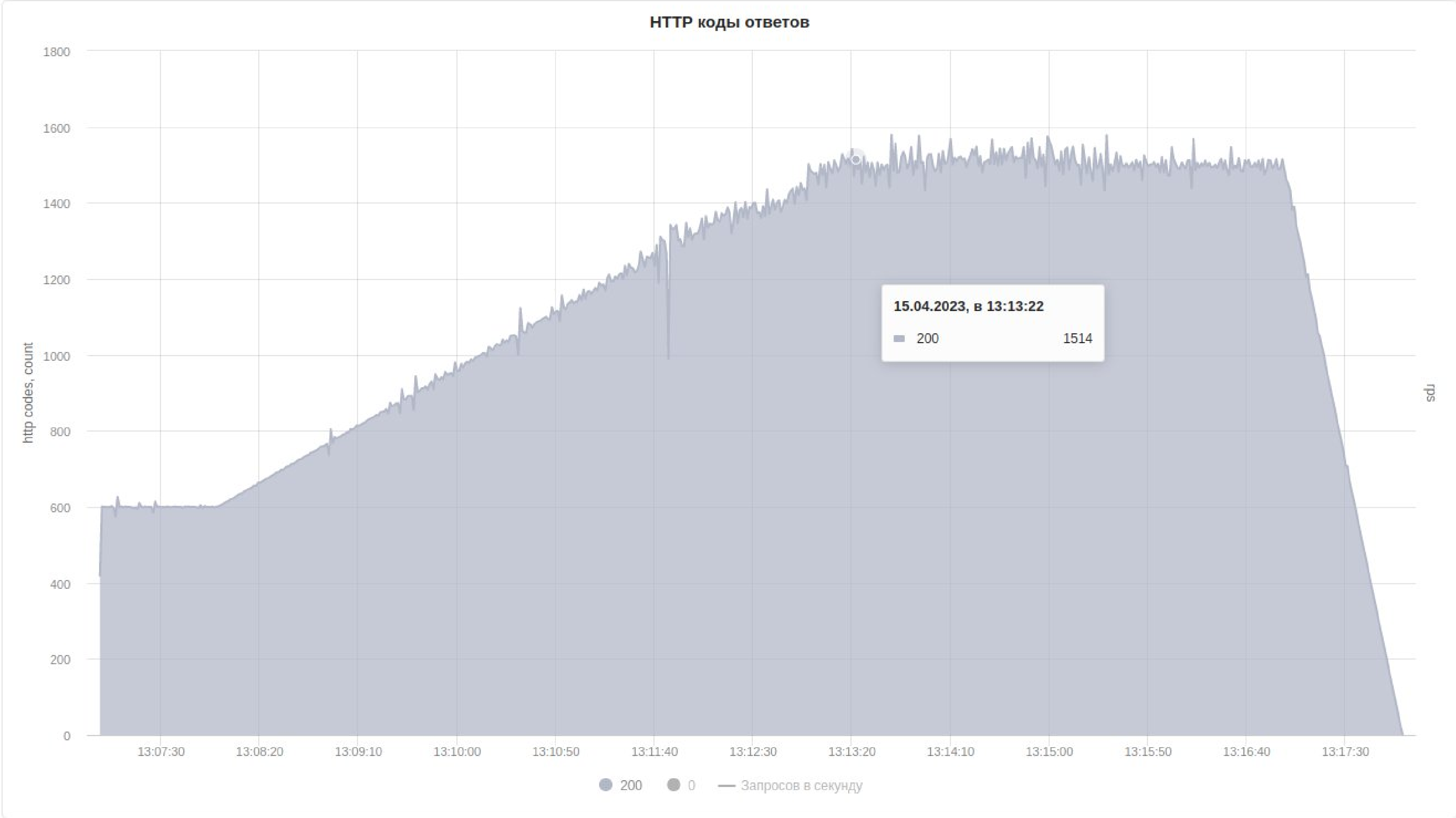
Демо - сетевые коды



Демо - метрики автомасштабирования



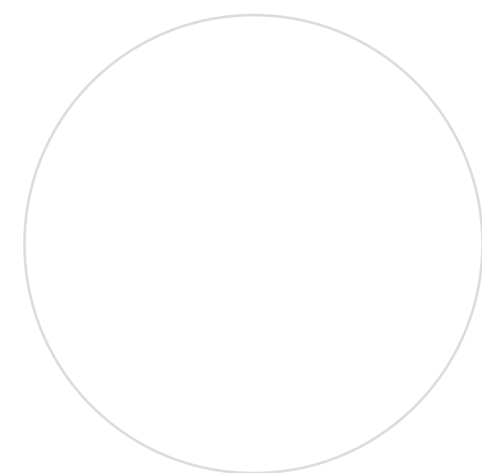
Демо - HTTP коды



Спасибо!



t.me/YandexLoadTesting



Липин Григорий
Инженер по тестированию
ligreen@yandex-team.ru