НТБот

Нагрузочное тестирование на автопилоте ИИ

Алексей Дементьев · ООО «Рондем» · SQA Days 39



39 sqa days

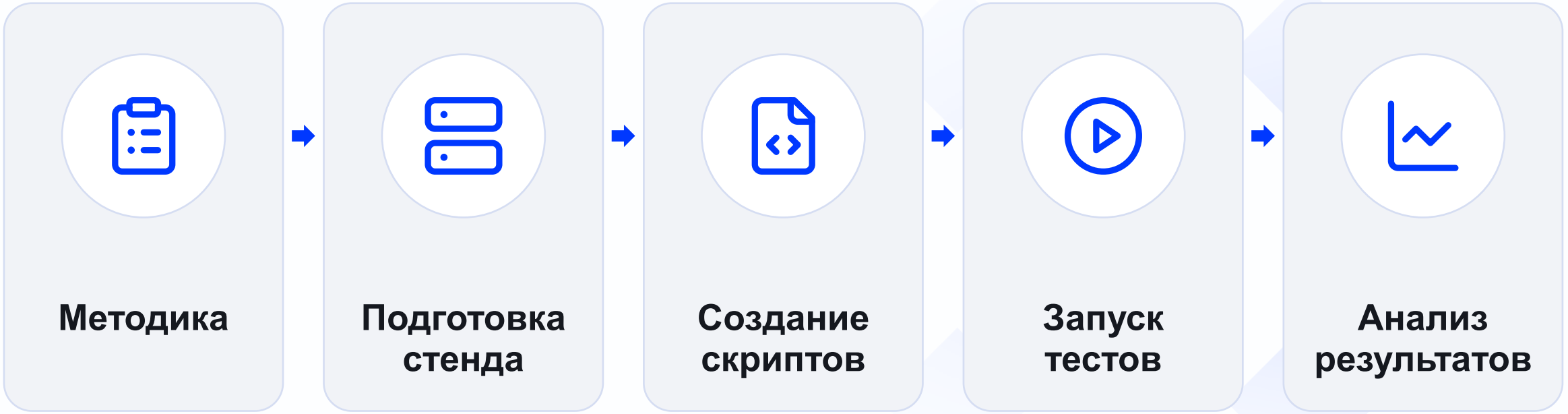
План доклада

НТБот

1. **Процесс НТ**
   - Общее описание
   - Обозначаем проблемы
2. **Что дает ИИ**
   - Рутина и варианты автоматизации с помощью LLM
3. **Возможности НТБот**
   - Создание скриптов
     - UI
     - API
   - Анализ результатов тестов
4. **Выводы**

Классический процесс НТ

НТБот



Методика и профиль

НТБот

Рутина

- Разбор логов и исторических данных
- Подсчет интенсивности по часам
- Обновление профиля нагрузки по свежим данным

Что дает ИИ-агент

- Сбор данных из множества источников
- Построение любых срезов

Профит

- Инженеру остается аналитика, агенту – механика
- Каждая цифра профиля прослеживается до источника

Подготовка стенда

НТБот

Рутина

- Уже во многом решается специализированными средствами

Что дает ИИ-агент

- Промпт: «разверни стенд с версией X»
- Дополняет Terraform, Ansible, Helm
- Анализирует ошибки раскатки

Профит

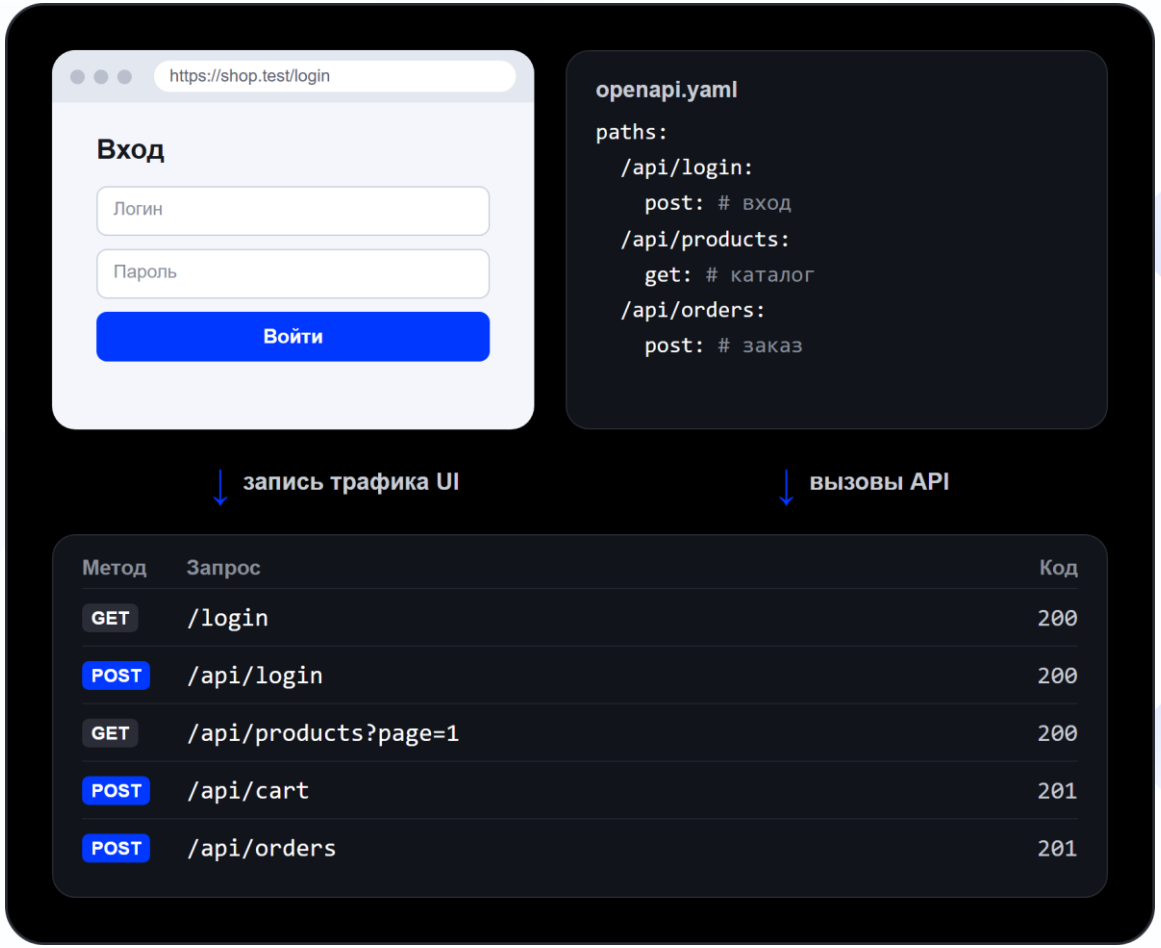
- 100% автоматизация
- История и снимки настроек стенда

Создание скриптов UI / API

НТБот

1 Запросы | 2 Формат | 3 Отладка

Для подачи нагрузки нужны скрипты, а для скриптов нужны запросы



Создание скриптов UI / API

НТБот

1 Запросы | **2 Формат** | 3 Отладка

APACHE JMeter™ .jmx

```
<HTTPSamplerProxy
    testname="POST /api/orders">
  <stringProp name="HTTPSampler.path">
    /api/orders</stringProp>
</HTTPSamplerProxy>
```

k6 .js

```
import http from 'k6/http';
export default function () {
  http.post(`${BASE}/api/orders`, body);
}
```

Gatling .scala

```
exec(
  http("POST /api/orders")
    .post("/api/orders")
    .body(StringBody(body))
)
```

LOCUST .py

```
class Shopper(HttpUser):
    @task
    def order(self):
        self.client.post(
            "/api/orders", json=body)
```

Создание скриптов UI / API

НТБот

1 Запросы | 2 Формат | **3 Отладка**

Самая рутинная работа в НТ требует самого опытного инженера

Параметризация

```
{ "login": "${login}",
  "password": "${password}" }
```

Корреляция

```
{ "token": "eyJhbGciOiJIUzI1Ni…" }
POST /api/orders
Authorization: Bearer ${token}
```

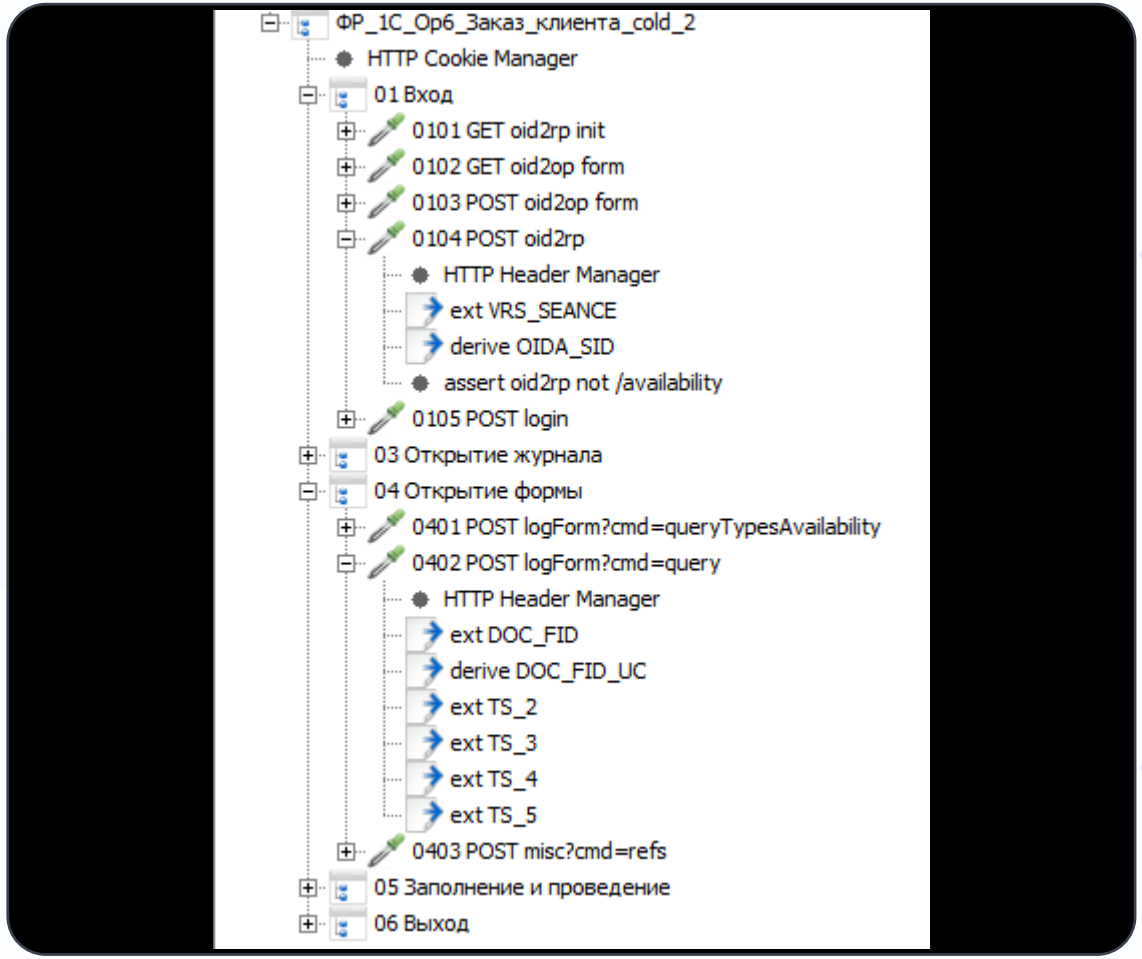
Валидация

✓ status == 201
✓ $.orderId существует

Промпт – Скрипт

НТБот

```
1. Открой https://***/3859415
2. Введи *** в поле логина
3. Введи *** в поле пароля
4. Нажми кнопку Войти
5. Нажми в левом меню на раздел Продажи
6. Нажми Документы продажи (все)
7. Открой любой проведённый документ
двойным кликом
8. Нажми кнопку Еще
9. Нажми Провести
10. Нажми Назад
11. Нажми на значок пользователя в
правом верхнем углу
12. Нажми Завершить работу (выйти)
```

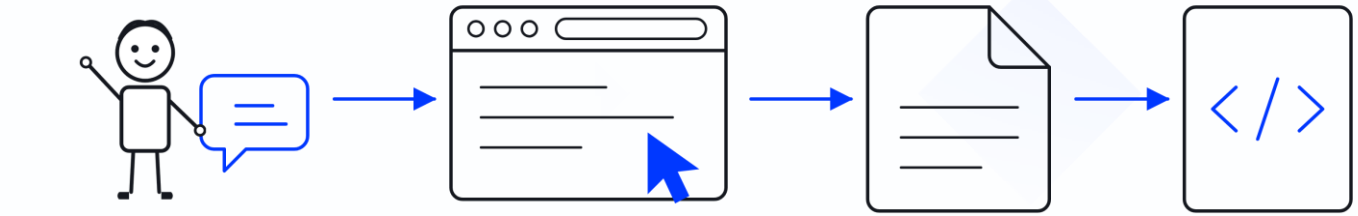


Создание скрипта

НТБот

1. Инженер: пишет промпт
2. **Агент**: ищет объект, выполняет действие, проверяет результат, пишет трафик
3. Инженер: наблюдает и подсказывает
4. **Агент**: собирает скрипт

```
// 1: title на обёртке <li>
<li title="Назад">
  <button></button></li>
// 2: Ant — имя внутри <span>
<button>
  <span role="img"
    aria-label="left">
  </span></button>
// 3: 1C — атрибут прямо на контроле
<a class="webBtn"
   title="Назад"></a>
```



Отладка скрипта

НТБот

Агент:

- расставляет транзакции
- выполняет корреляцию и параметризацию
- пишет экстракторы и регулярки
- добавляет проверки на смысловые части ответов

Самая «увлекательная» часть достается агенту

Инженер
отлаживает на локальной машине

Агент
отлаживает на генераторе, с которого пойдет нагрузка в тесте

Бесконечность – не предел!

НТБот

Агент не пытается «понять» новую систему заново и не полагается только на модель.

Он накапливает факты от пользователя, и использует их как готовое знание.

```
"commit_input_key": "Tab",
"input_via_clipboard": true,
"input_rejected_markers": ["нет в списке"],
"styled_checkbox_selector": "div.checkbox",
"cases": [{ "idx": 6, "target_text": "Создать",
      "result_marker": "Раскрылось меню выбора" }]
```



Скрипты для API

НТБот

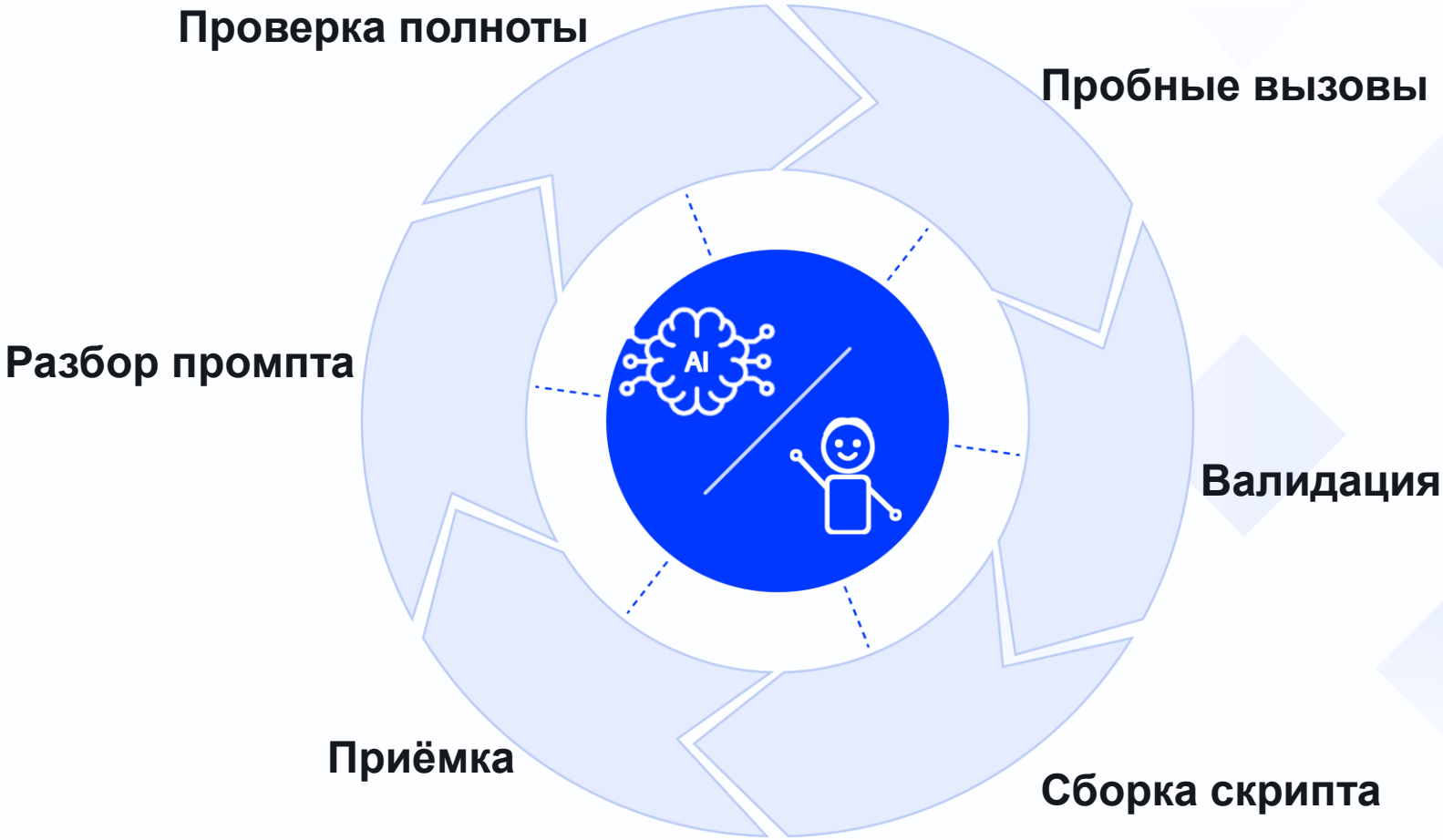
Отличие от UI – в интерфейсе

Скрипт API требует больше технической информации на входе

REST	OpenAPI · коллекция Postman · запись HAR
Очередь	адрес брокера · топик · пример сообщения
База данных	запрос или процедура · адрес · данные
gRPC	файл .proto · адрес службы

Сборка скрипта

НТБот



Что дает ИИ-агент

НТБот

- Поиск связи между вызовами по живым ответам
- Переиспользование реквизитов для подключения и других данных как «знание» о системе
- Проверка изменений в новых версиях OpenAPI
- Пересборка скрипта под разные инструменты
- Валидация ответов по смыслу операции, а не только код 200
- Корреляция и параметризация

Запуск тестов

НТБот

Рутина

- Подбор ступеней в поиске максимальной производительности
- Наблюдение за ходом теста
- Ночные тесты, за которыми никто не следит

Что дает ИИ-агент

- Проводит серии тестов по цели: меняет уровни нагрузки и анализирует результат
- Мониторит тест

Профит

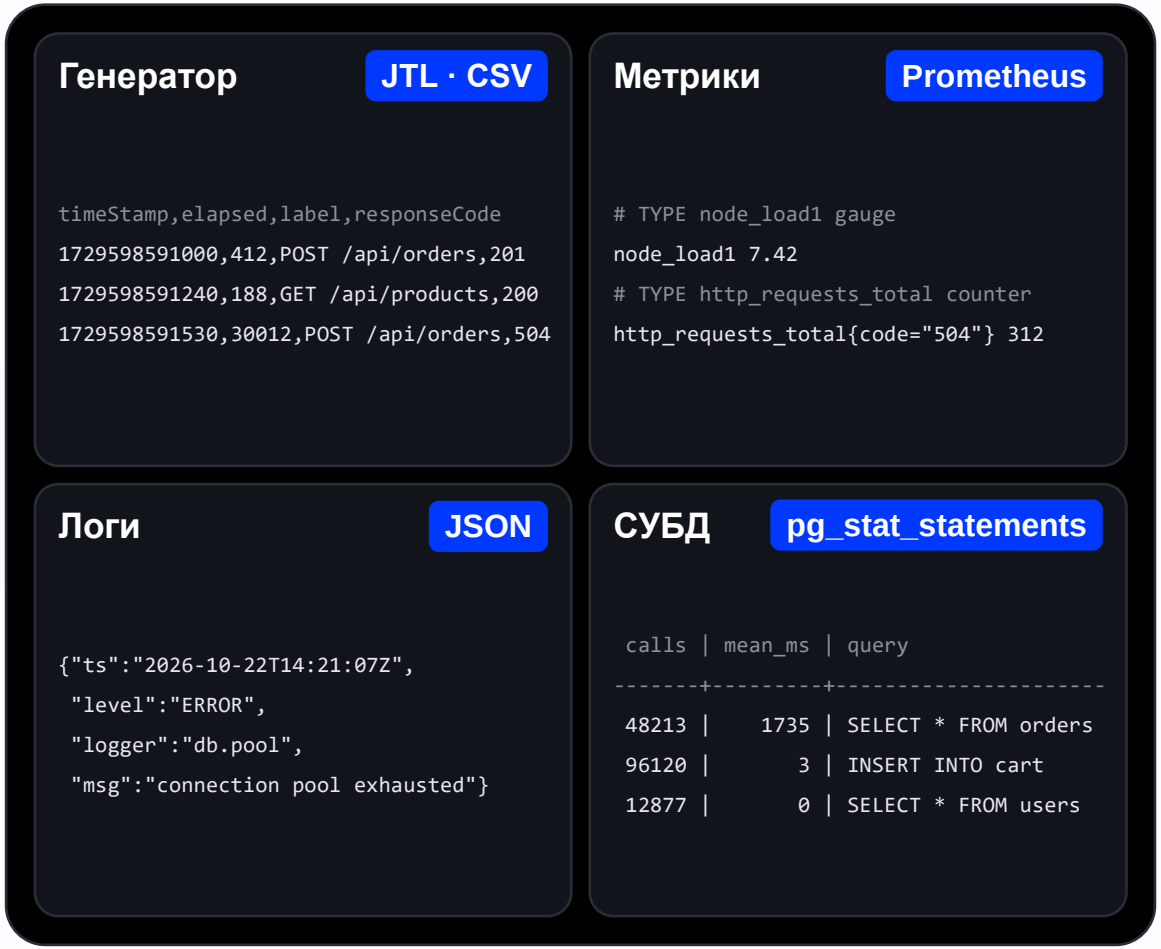
- Больше тестов за то же время
- Инженер не дежурит у монитора
- Полный журнал работ: уровни нагрузки, критерии и причины остановки

Анализ результатов тестов

НТБот

1 Сбор | 2 Привязка | 3 Фильтрация

Для вердикта PASS / FAIL нужно свести много метрик прежде, чем делать выводы

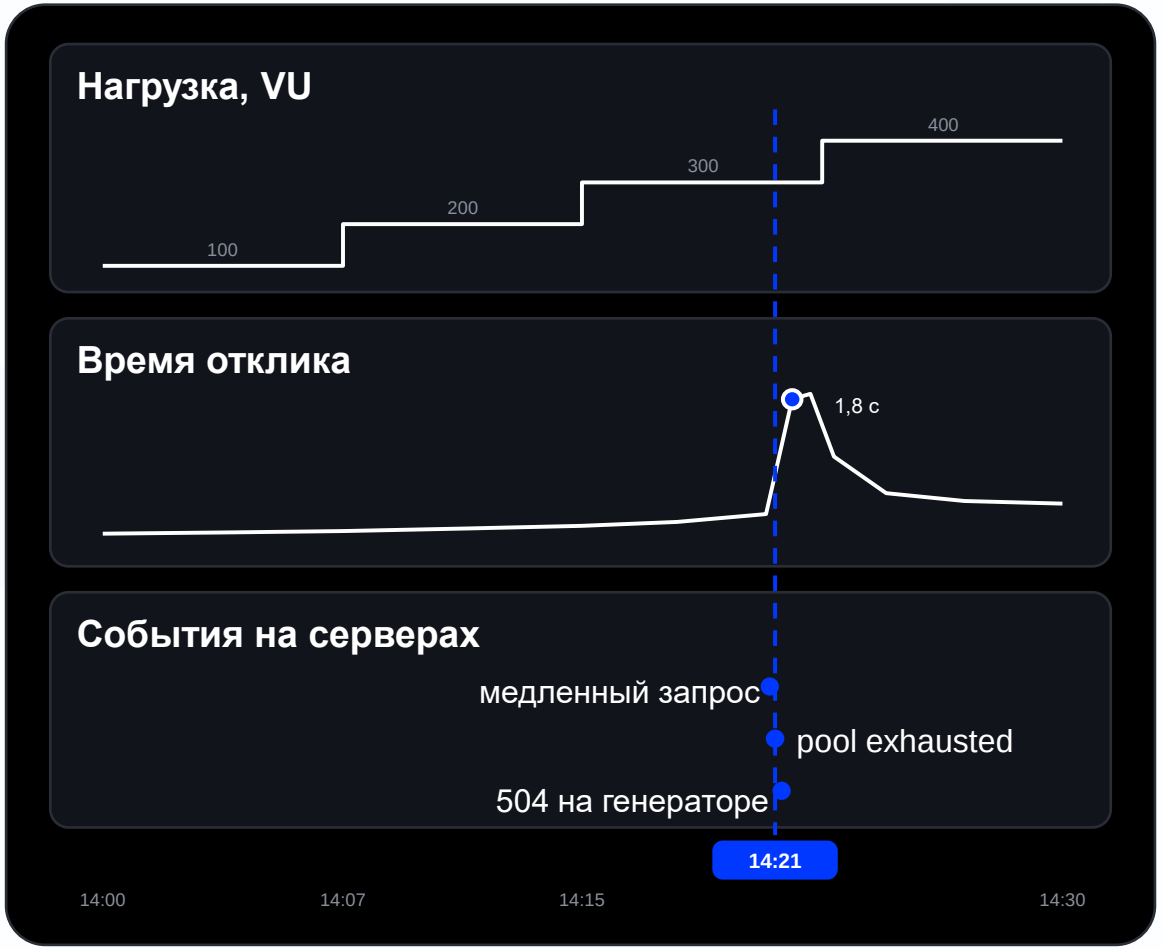


Анализ результатов тестов

НТБот

1 Сбор | **2 Привязка** | 3 Фильтрация

Сопоставление событий с уровнями нагрузки и транзакциями теста

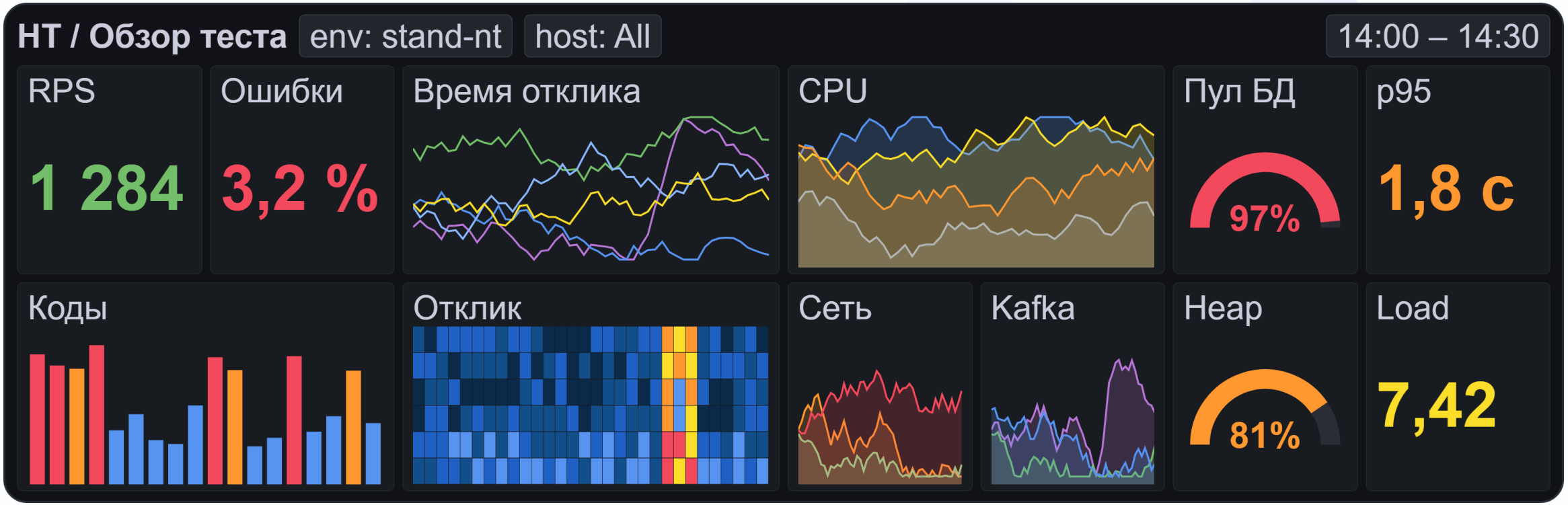


Анализ результатов тестов

НТБот

1 Сбор | 2 Привязка | **3 Фильтрация**

Время уходит на сбор данных, а не на анализ



Этапы анализа

НТБот

Шаг 1

Находим аномалии

Агент по данным генератора смотрит, где и когда:

- транзакции стали обрабатываться дольше нормы
- появились ошибки
- просела интенсивность

Шаг 2

Снимаем телеметрию

Из доступных источников по стенду собирается телеметрия для понимания, что происходило на серверах во время аномалии

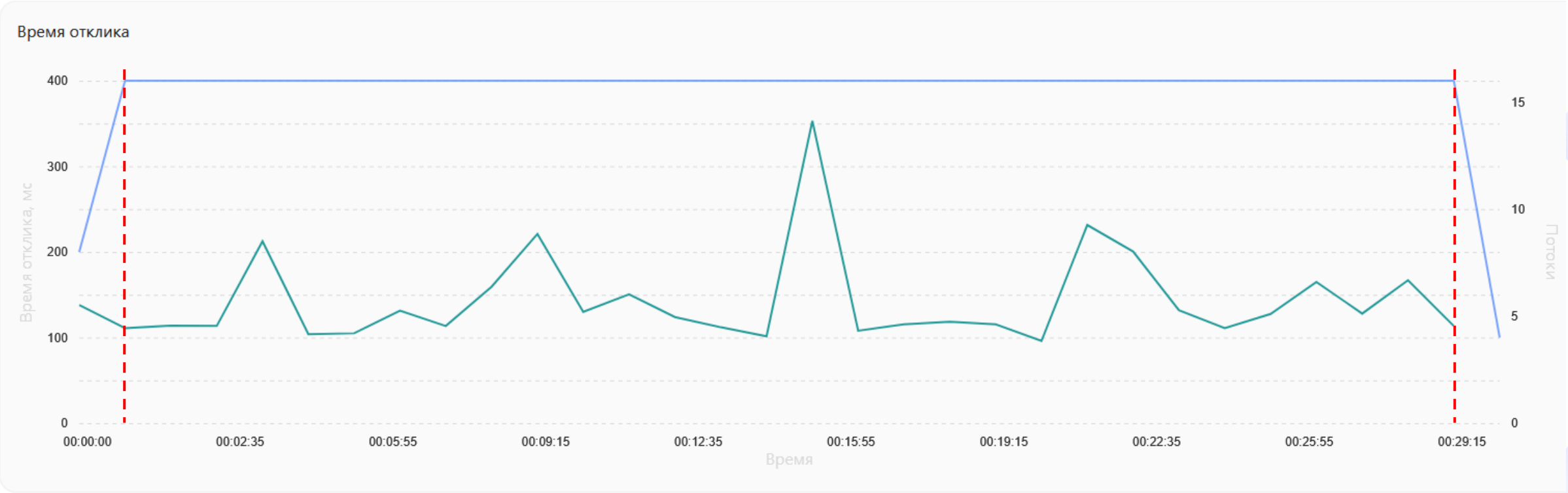
Шаг 3

Определяем причину

Факты проходят через обработку с ранжированием по уровню уверенности

Аномалии

НТБот



Транзакции с превышением времени отклика

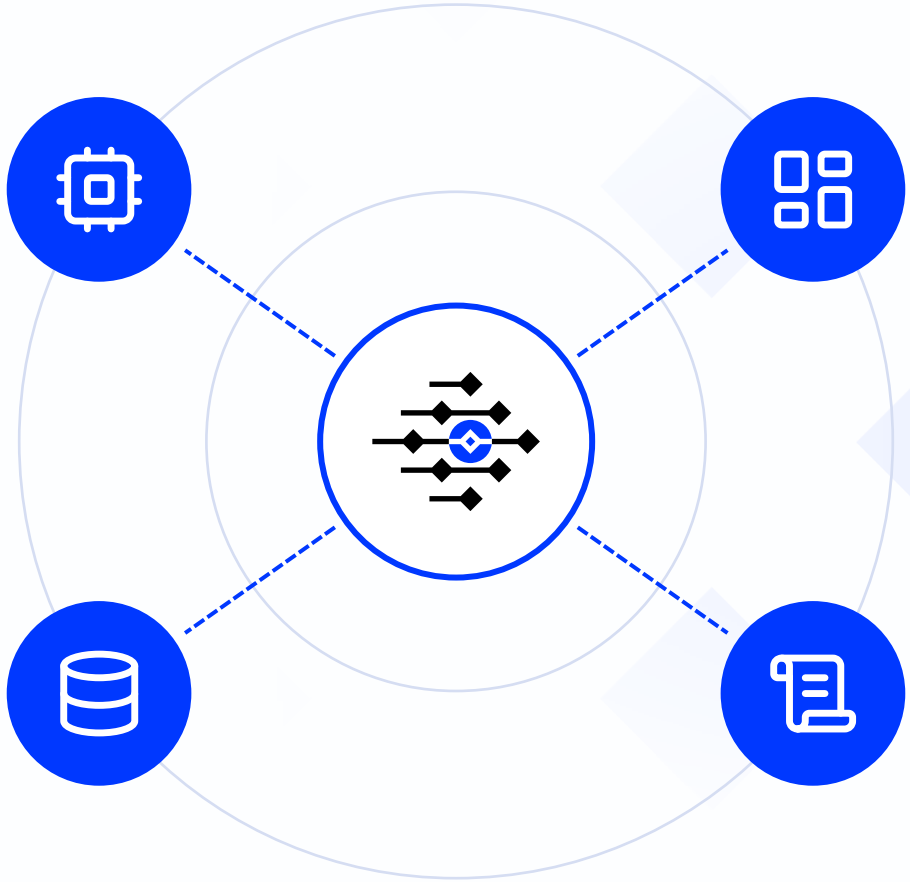
Транзакции с превышением допустимого процента ошибок

Источники

НТБот

Prometheus/Zabbix

Аппаратные метрики со всей инфраструктуры



Grafana

Кастомные запросы через datasource и запросы

СУБД

Лимиты и настройки, дедлоки, топ запросов, пул соединений, блокировки, время выполнения конкретных запросов

Логи

События с фактическими отметками даты и времени

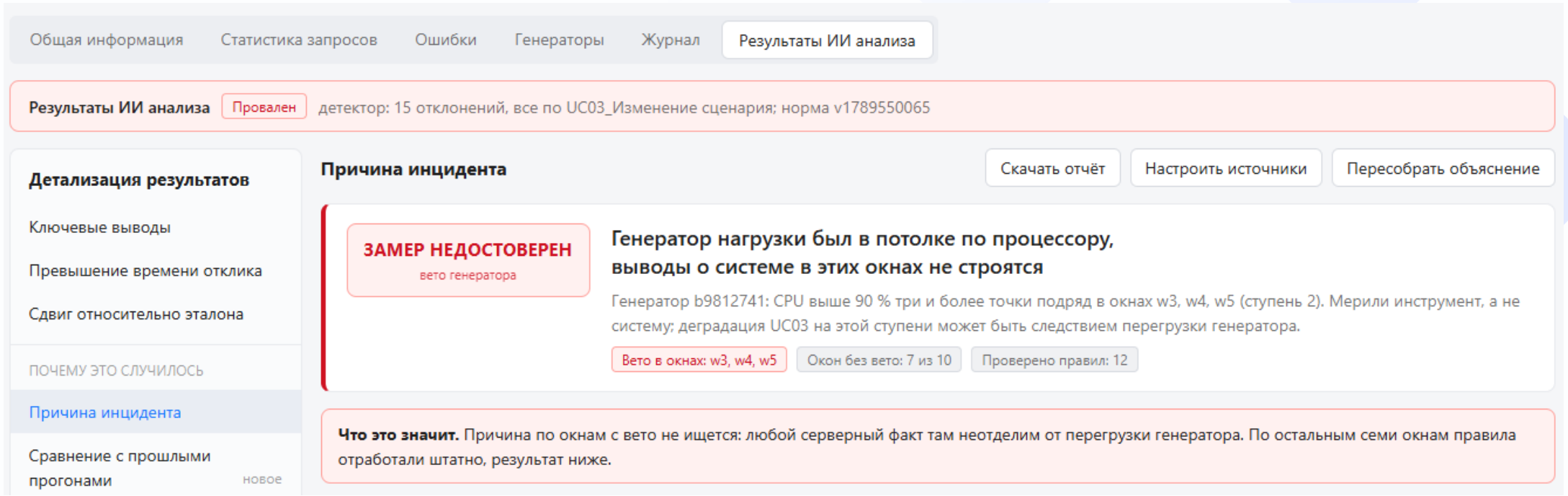
Результаты анализа

НТБот

Генераторы

Мониторинг

Логи



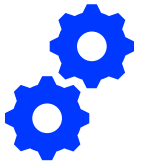
Выводы

НТБот

- Делегируйте рутину ИИ:
  - **экономия времени**
  - **снижение порога входа**
  - **переключение на более сложные задачи**

- **Модель** – необходимое условие, а не решение целиком. Свои лучшие качества она раскрывает только в правильной оправе

- И пока еще **требуется разбираться** в предметной области, чтобы эффект от инструмента был максимальным



50-80%
разработка и отладка скриптов



до 40%
подготовка и запуск тестов



>30%
анализ и сравнение результатов тестов

НТБот

НТБот

Увеличиваем производительность.
Сохраняем стабильность.

Алексей Дементьев

@Alexey_A_Dementyev

