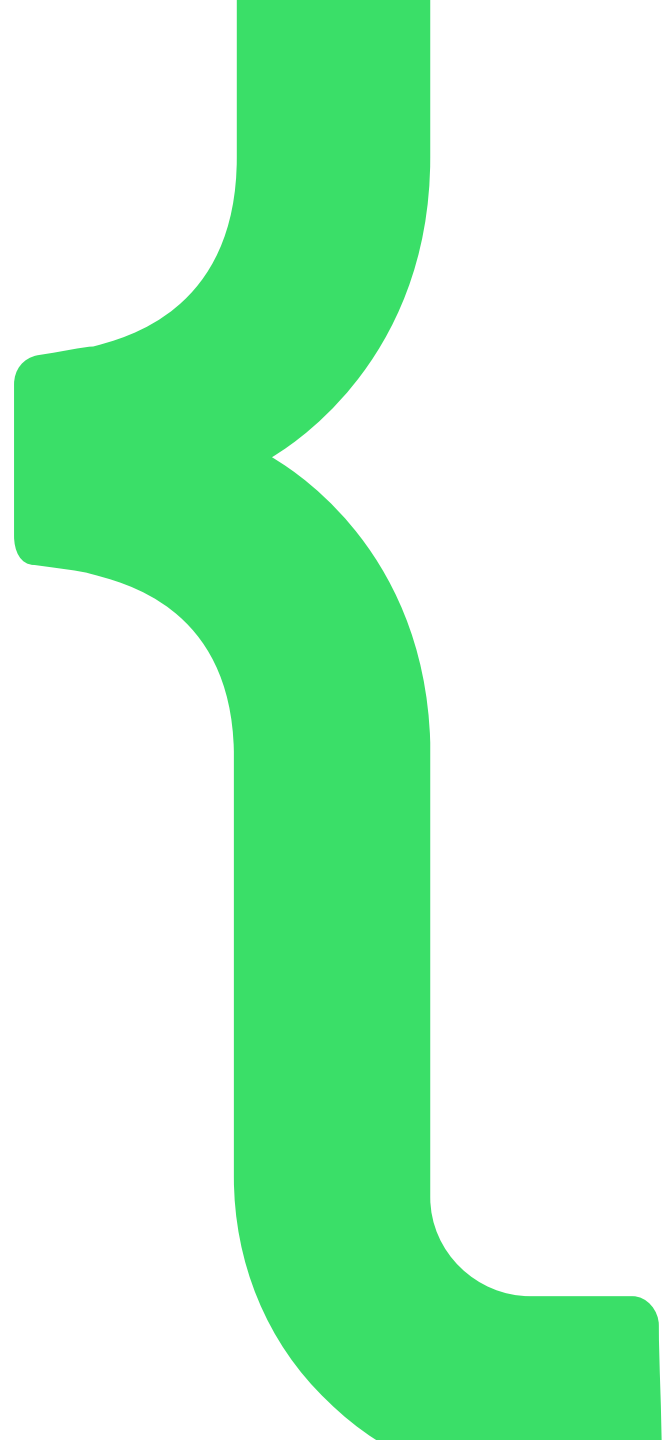




# Метастабильные состояния отказа



**Андрей Синицын**  
Руководитель отдела MessageBus



# Андрей Синицын



✉ [andrey@sinits.in](mailto:andrey@sinits.in)

- ➔ Больше **20 лет** в индустрии
- ➔ Был разработчиком и девопсом, сейчас **менеджер**
- ➔ Работал в очень больших компаниях
- ➔ Сотни пройденных инцидентов, больших и маленьких



# AGENDA

Введение



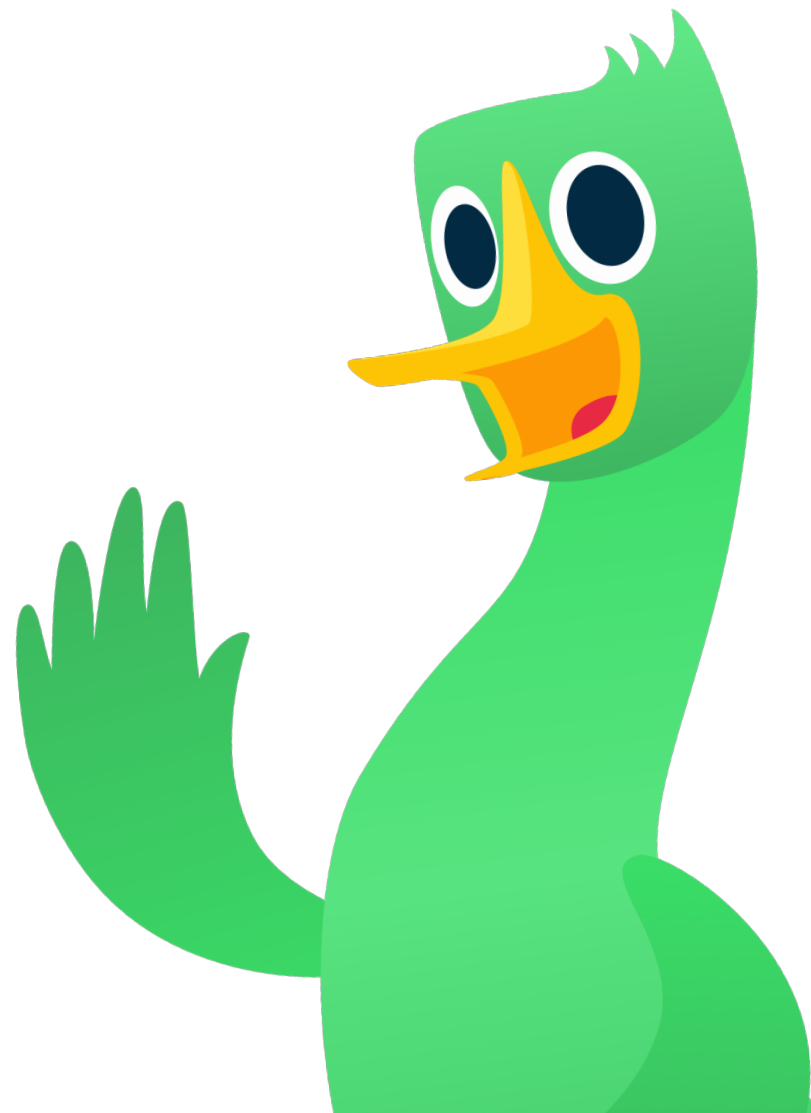
Инструменты мониторинга

Кейс: Kubernetes и микросервисы

Кейс: Kafka

Методологии предотвращения

Выводы



**Профессионалы в любой профессии:**  
**Мы знаем, что делаем.**  
**Программисты:**



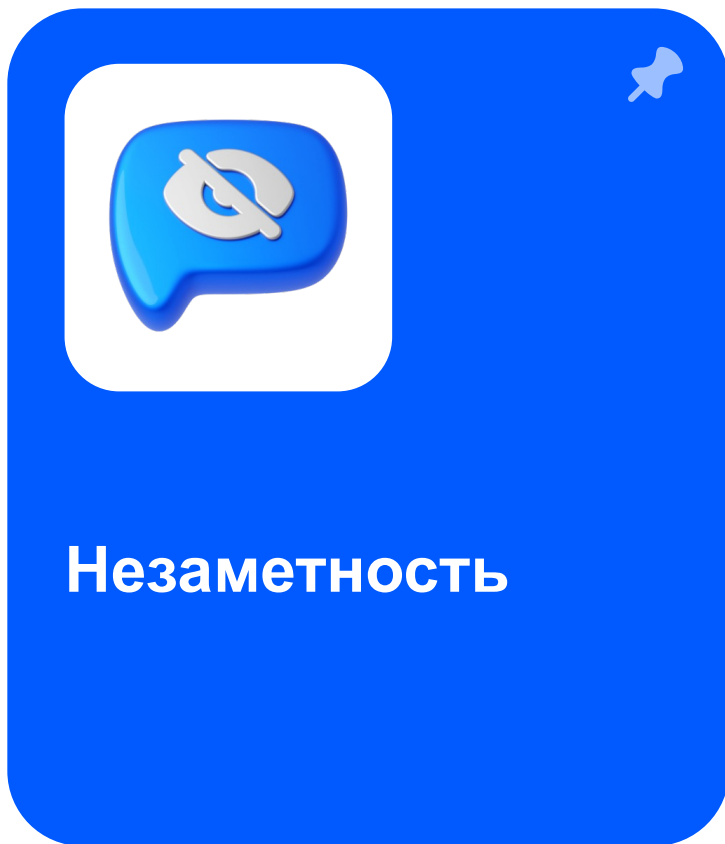




## Метастабильное состояние отказа

длительное состояние системы с  
существенно сниженной  
производительностью при формально  
работающих компонентах

# Почему это опасно?



# Почему это опасно?



Незаметность



Сложность  
диагностики

# Почему это опасно?



Незаметность



Сложность  
диагностики



**Накопительный  
эффект**



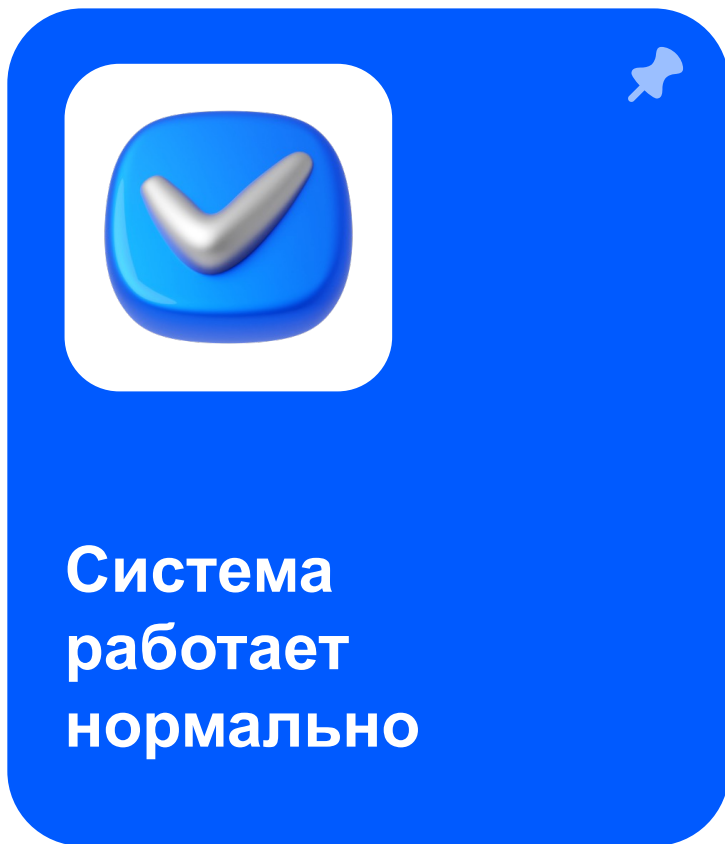


1 час даунтайма  
=  
1 час даунтайма



- 50% производительности  
=  
**84 часа в неделю**

# Состояния системы



# Состояния системы



Система  
работает  
нормально



Система не  
работает



# Состояния системы



Система  
работает  
нормально



Система не  
работает



**Метастабильное  
состояние**

# Каскадные эффекты

-100 ms



# Каскадные эффекты

-100 ms



-100 ms



# Каскадные эффекты

-100 ms



-100 ms



Замедляется





# Каскадные эффекты

-100 ms



-100 ms



Замедляется

Страдает UX





**Модели  
обнаружения**



**Пороговый мониторинг**



## Модели обнаружения



Пороговый мониторинг



Мониторинг аномалий



## Модели обнаружения

⚡ Пороговый мониторинг

⚡ Мониторинг аномалий

⚡ **Мониторинг трендов**



# AGENDA

Введение

**Инструменты мониторинга**

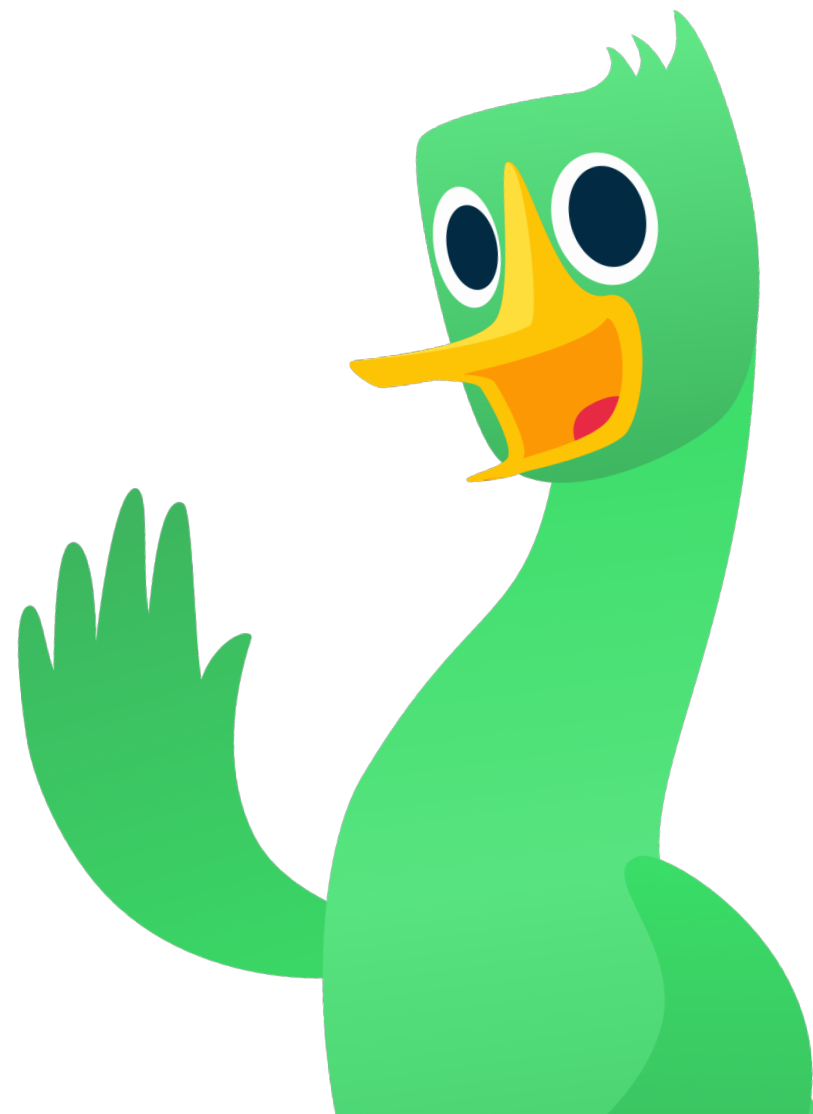


Кейс: Kubernetes и микросервисы

Кейс: Kafka

Методологии предотвращения

Выводы



# Prometheus Grafana



# Prometheus + Grafana



**Алерты на изменение тренда**

# Prometheus + Grafana



Алерты на изменение тренда



**Алерты на разброс метрик**

# Prometheus + Grafana

 Алерты на изменение тренда

 Алерты на разброс метрик

 **Алерты на аномалии в распределении**

# Prometheus + Grafana



```
rate(http_request_duration_seconds_sum[5m]) /  
rate(http_request_duration_seconds_count[5m]) > 1.2 *  
avg_over_time(rate(http_request_duration_seconds_sum[5m])  
/ rate(http_request_duration_seconds_count[5m]) [1d:5m])
```

# ELK Stack



# ELK Stack



**Структурированное логирование**



# ELK Stack



Структурированное логирование



**Агрегации для выявления аномалий**

# ELK Stack



Структурированное логирование



Агрегации для выявления аномалий



**Визуализации трендов**

# OpenTelemetry Jaeger



# OpenTelemetry + Jaeger



**Трассировка запросов**

# OpenTelemetry + Jaeger



Трассировка запросов



**Отслеживание времени каждого этапа обработки**

# OpenTelemetry + Jaeger

 Трассировка запросов

 Отслеживание времени каждого этапа обработки

 **Алерты на растущие «хвосты» распределения**

Chaos  
Engineering



Litmus

# Chaos Engineering



**Целенаправленное внесение задержек**



# Chaos Engineering



Целенаправленное внесение задержек



**Искусственное ограничение ресурсов**

# Chaos Engineering

 Целенаправленное внесение задержек

 **Искусственное ограничение ресурсов**

 **Симуляция частичной недоступности**

# AGENDA

Введение

Инструменты мониторинга

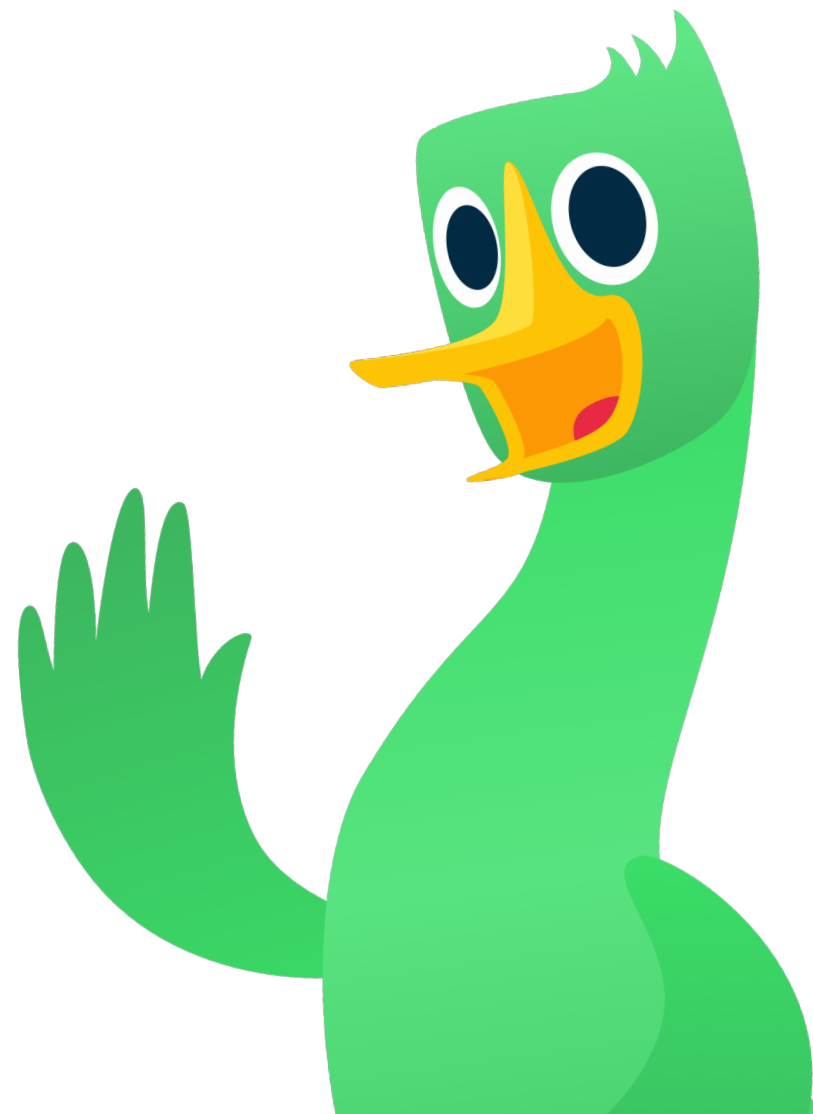
**Кейс: Kubernetes и микросервисы**



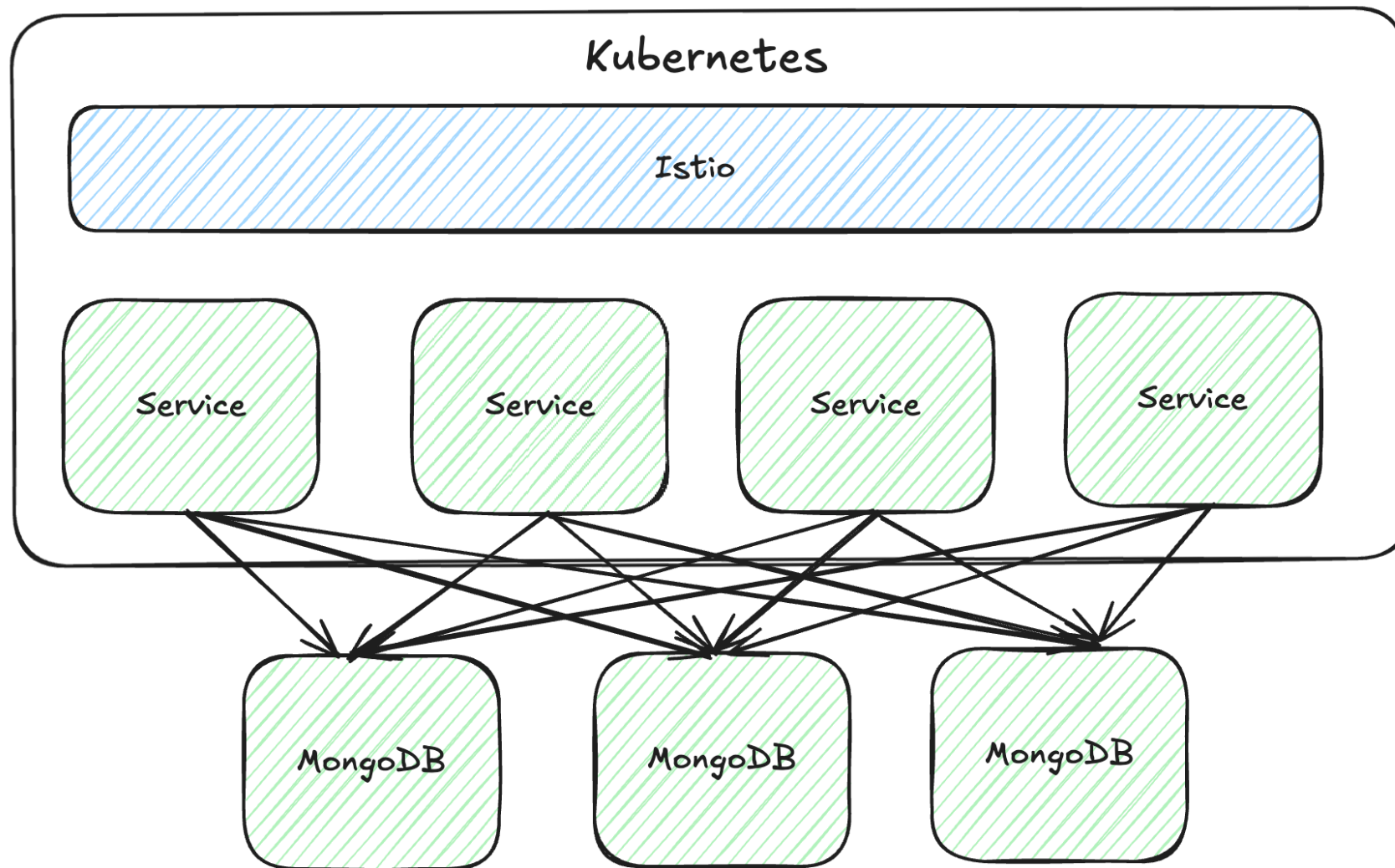
Кейс: Redis

Методологии предотвращения

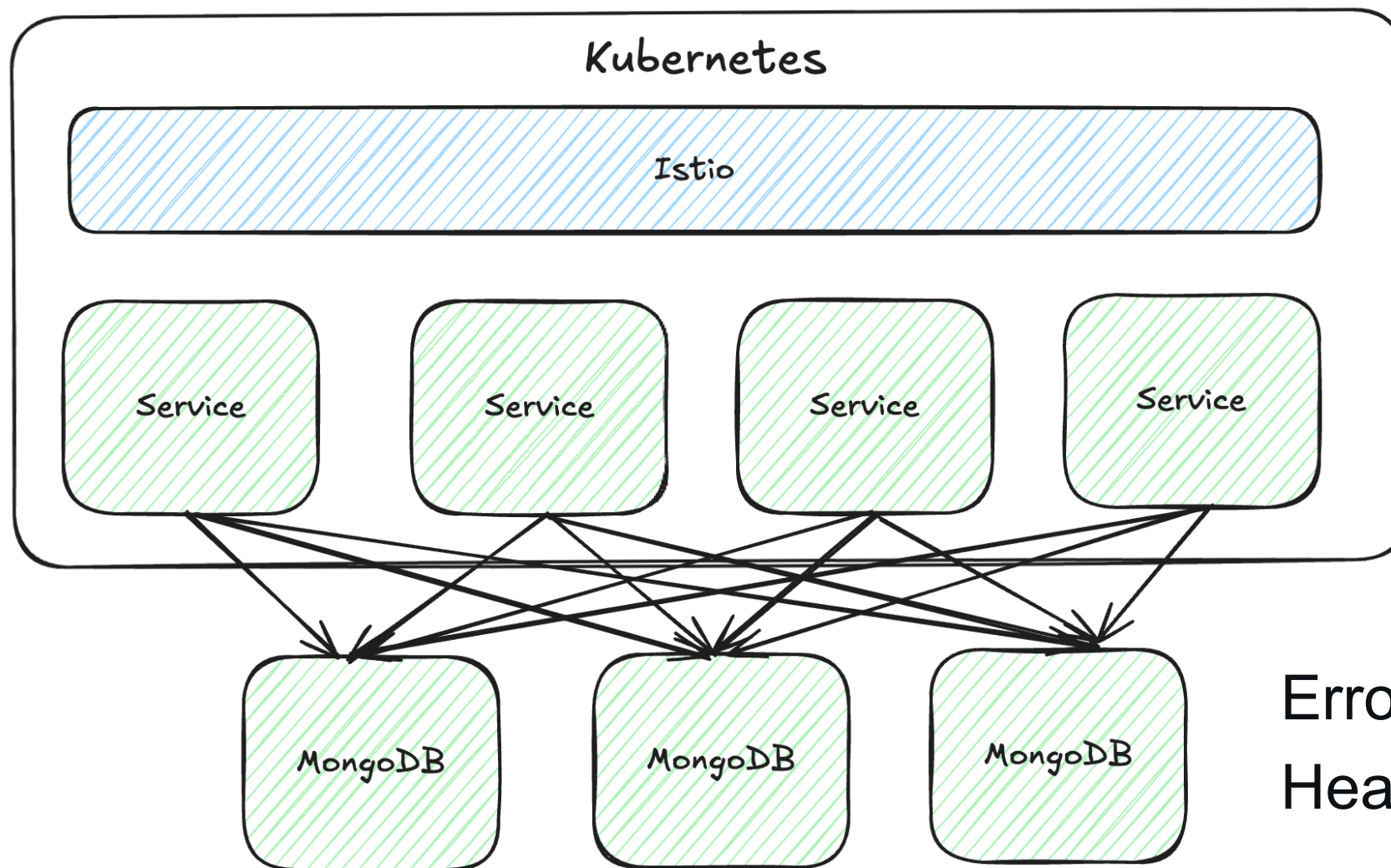
Выводы



# Кейс: кubernetes и микросервисы

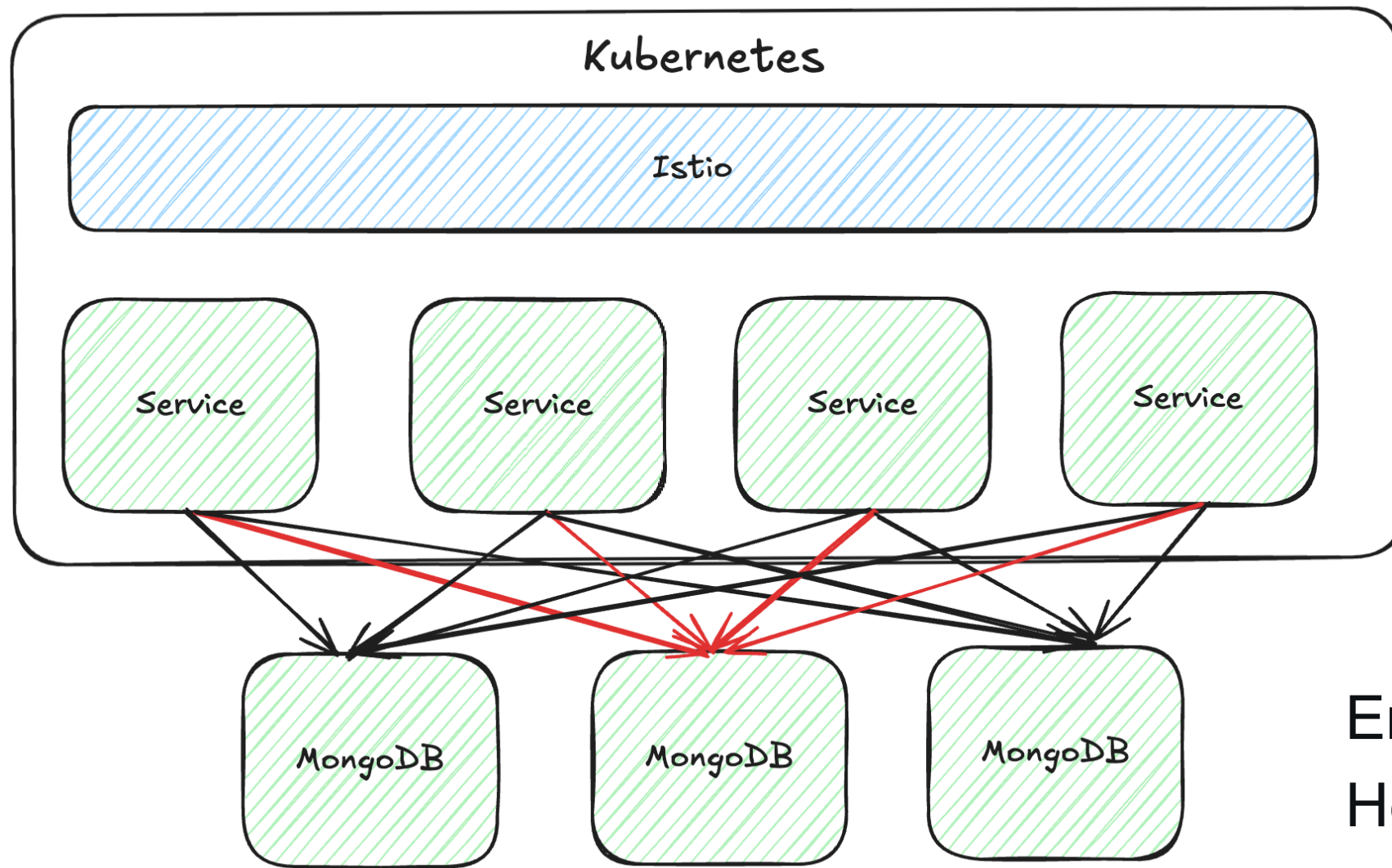


# Кейс: кubernetes и микросервисы



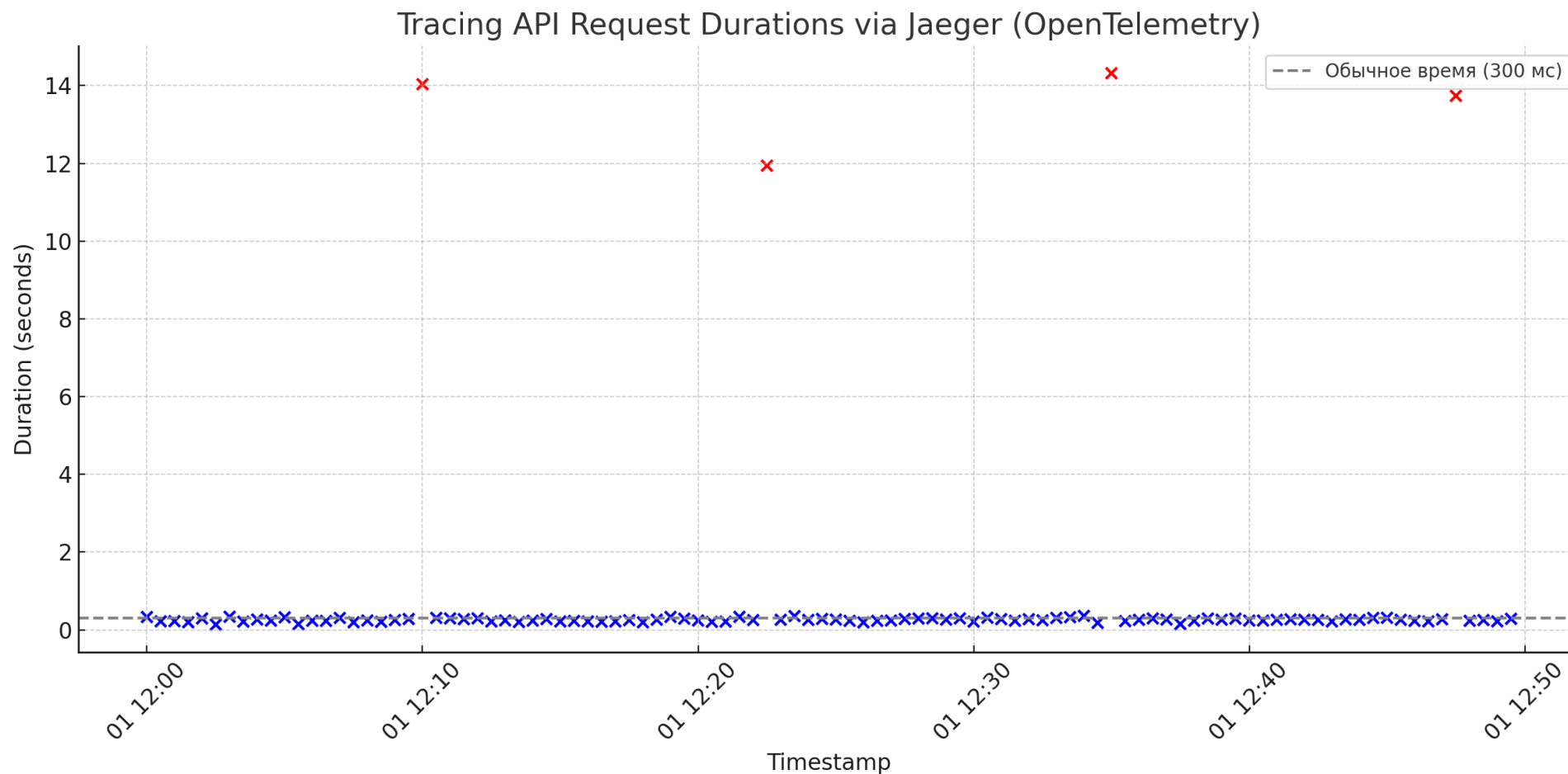
ErrorRate: **OK**  
HealthChecks: **OK**

# Кейс: кubernetes и микросервисы

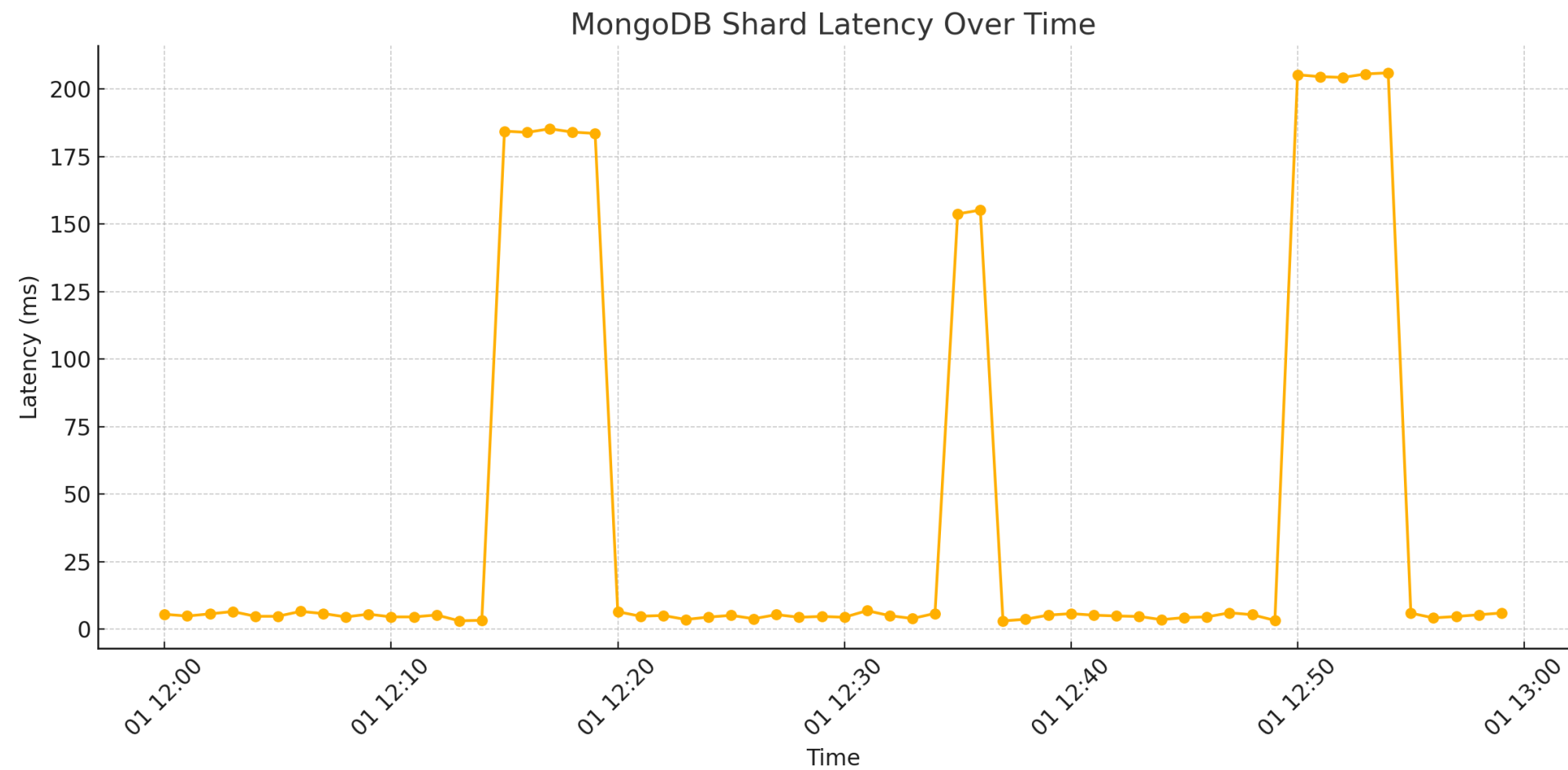


ErrorRate: **OK**  
HealthChecks: **OK**

# Кейс: кubernetes и микросервисы

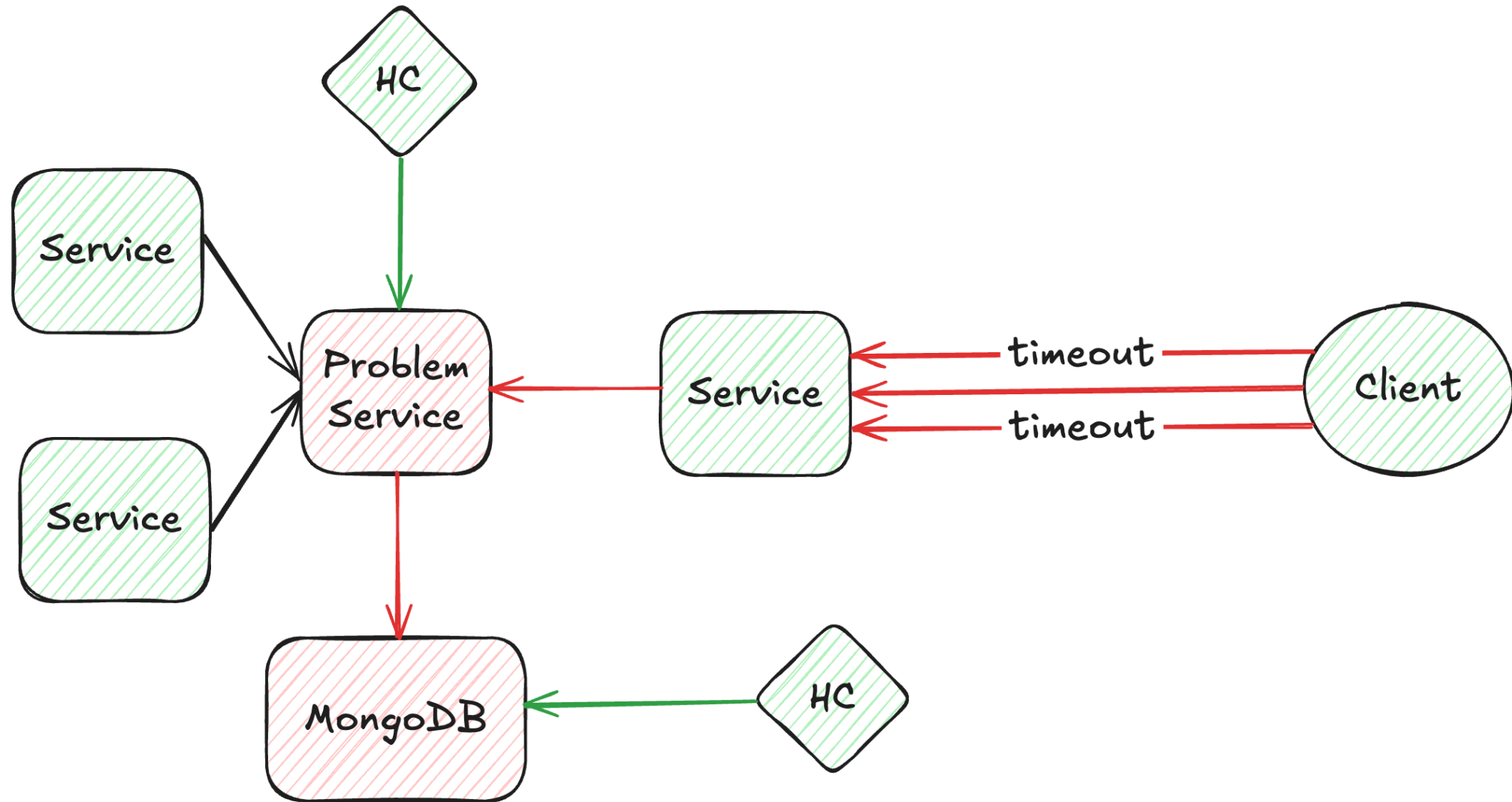


# Кейс: кубернетес и микросервисы

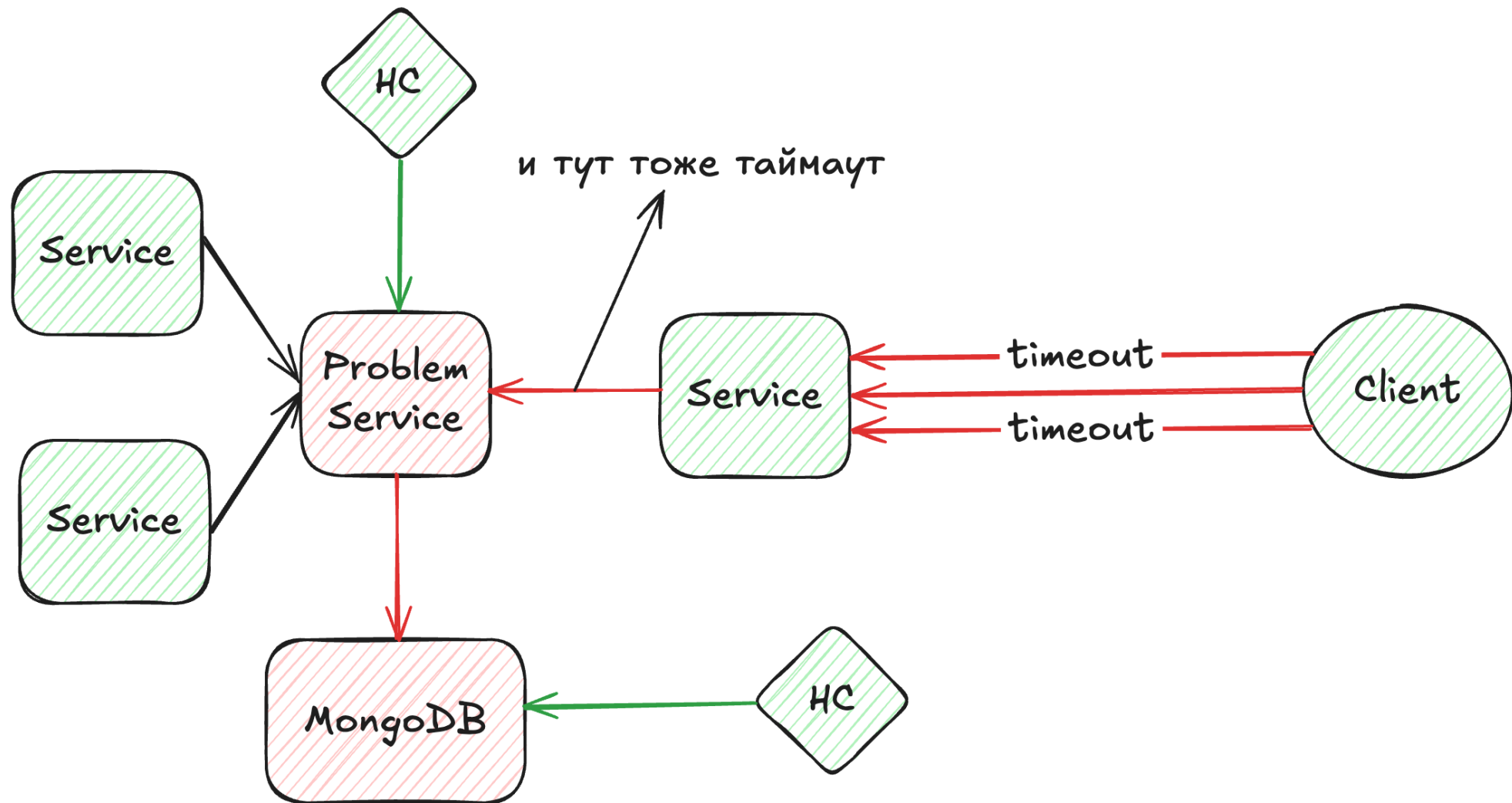




# Кейс: кubernetes и микросервисы



# Кейс: кубернетес и микросервисы



# Решение кейса



## Istio и Circuit Breaker

# Решение кейса



Istio и Circuit Breaker



Улучшенные health-чеки

# Решение кейса



Istio и Circuit Breaker



Улучшенные health-чеки



**Правильная настройка таймаутов**

# Решение кейса



Istio и Circuit Breaker



Улучшенные health-чеки



Правильная настройка таймаутов



**Мониторинг времени ответа с перцентилями**

# Решение кейса

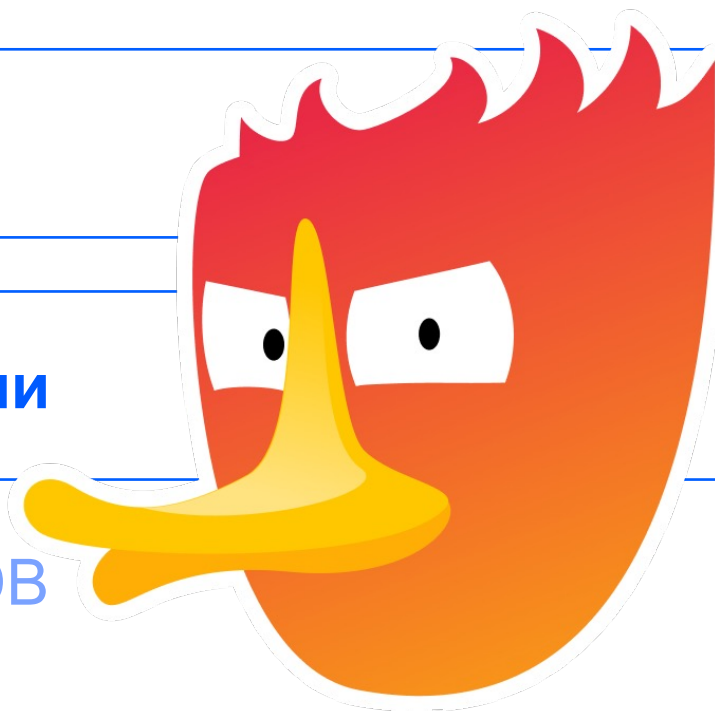
⚙ Istio и Circuit Breaker

⚙ Улучшенные health-чеки

⚙ Правильная настройка таймаутов

⚙ **Мониторинг времени ответа с перцентилями**

И выкинули монгу починили индексы в MongoDB



# AGENDA

Введение

Инструменты мониторинга

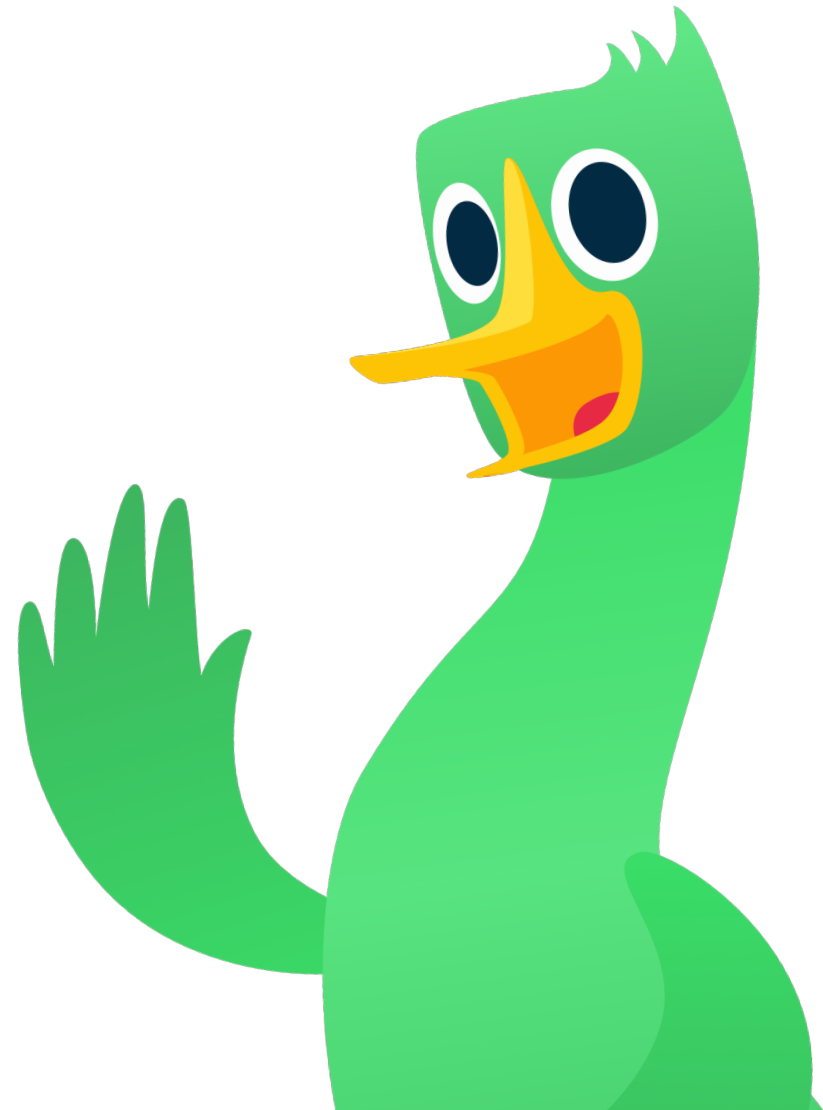
Кейс: Kubernetes и микросервисы

**Кейс: Kafka**



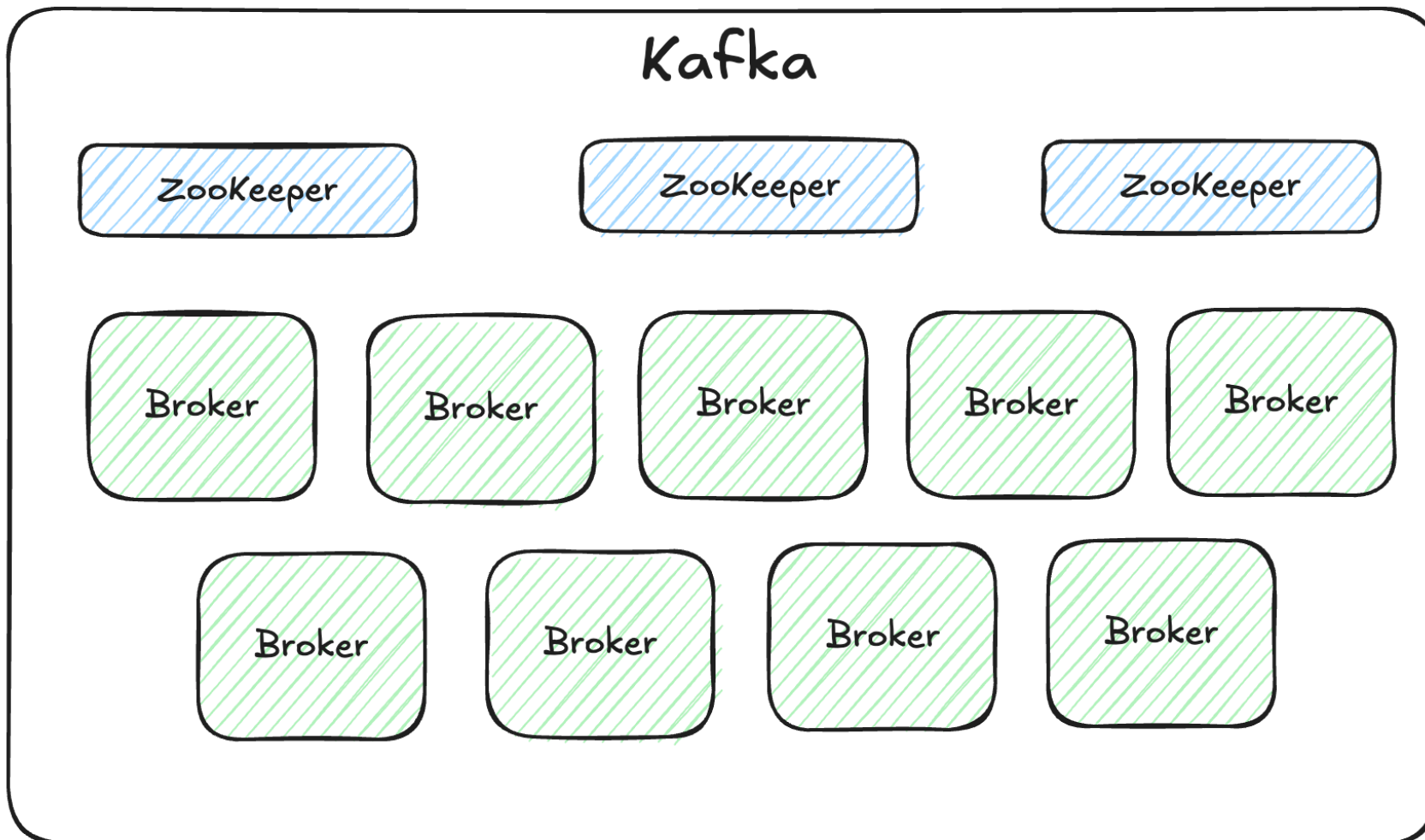
Методологии предотвращения

Выводы





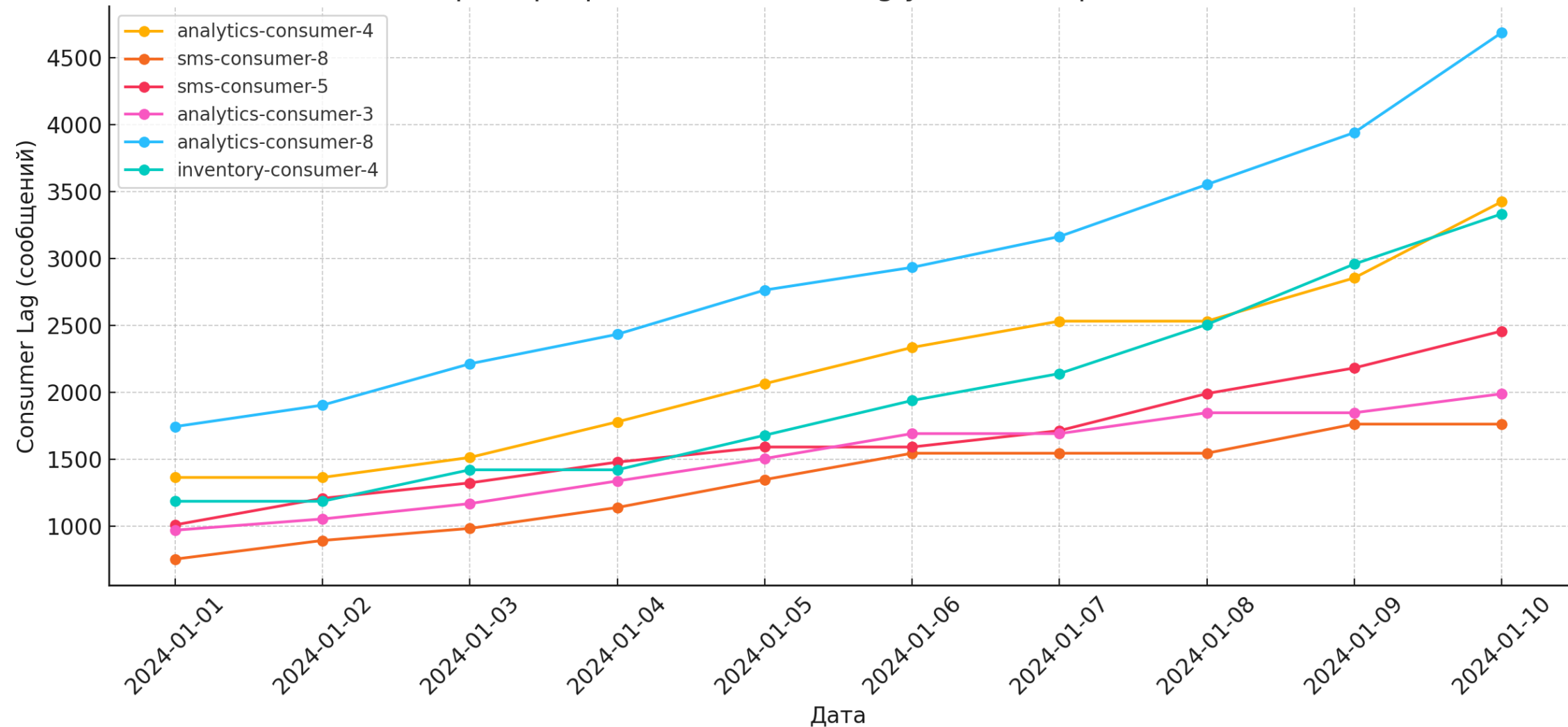
# Кейс: кафка



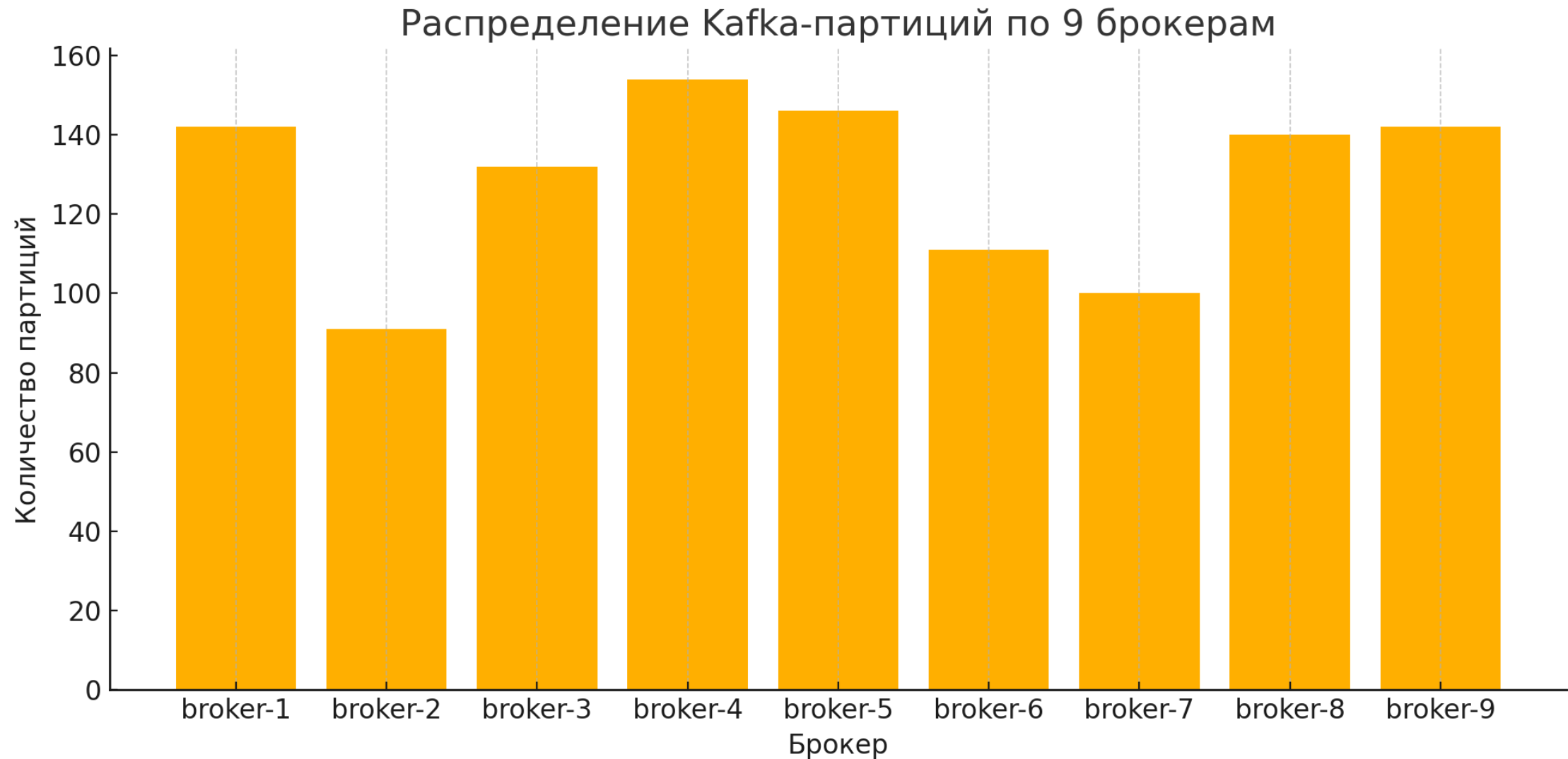
Несколько десятков топиков, некоторые с 50+ партициями  
Около 30 различных consumer-групп

# Кейс: кафка

Примеры роста Consumer Lag у Kafka-потребителей



# Кейс: кафка



# Кейс: кафка



# Решение кейса



**Неоптимальная конфигурация retention**

# Решение кейса



Неоптимальная конфигурация retention



**Проблемы с дисковым I/O**

# Решение кейса

 Неоптимальная конфигурация retention

 Проблемы с дисковым I/O

 **Неравномерная нагрузка на партиции**

# Решение кейса

 Неоптимальная конфигурация retention

 Проблемы с дисковым I/O

 Неравномерная нагрузка на партиции

 **Настройки consumer groups**



# Решение кейса



# Решение кейса: zookeeper



**Чрезмерное количество ephemeral nodes**

# Решение кейса: zookeeper



Чрезмерное количество ephemeral nodes



**Неоптимальные настройки JVM**

# Решение кейса: zookeeper



Чрезмерное количество ephemeral nodes

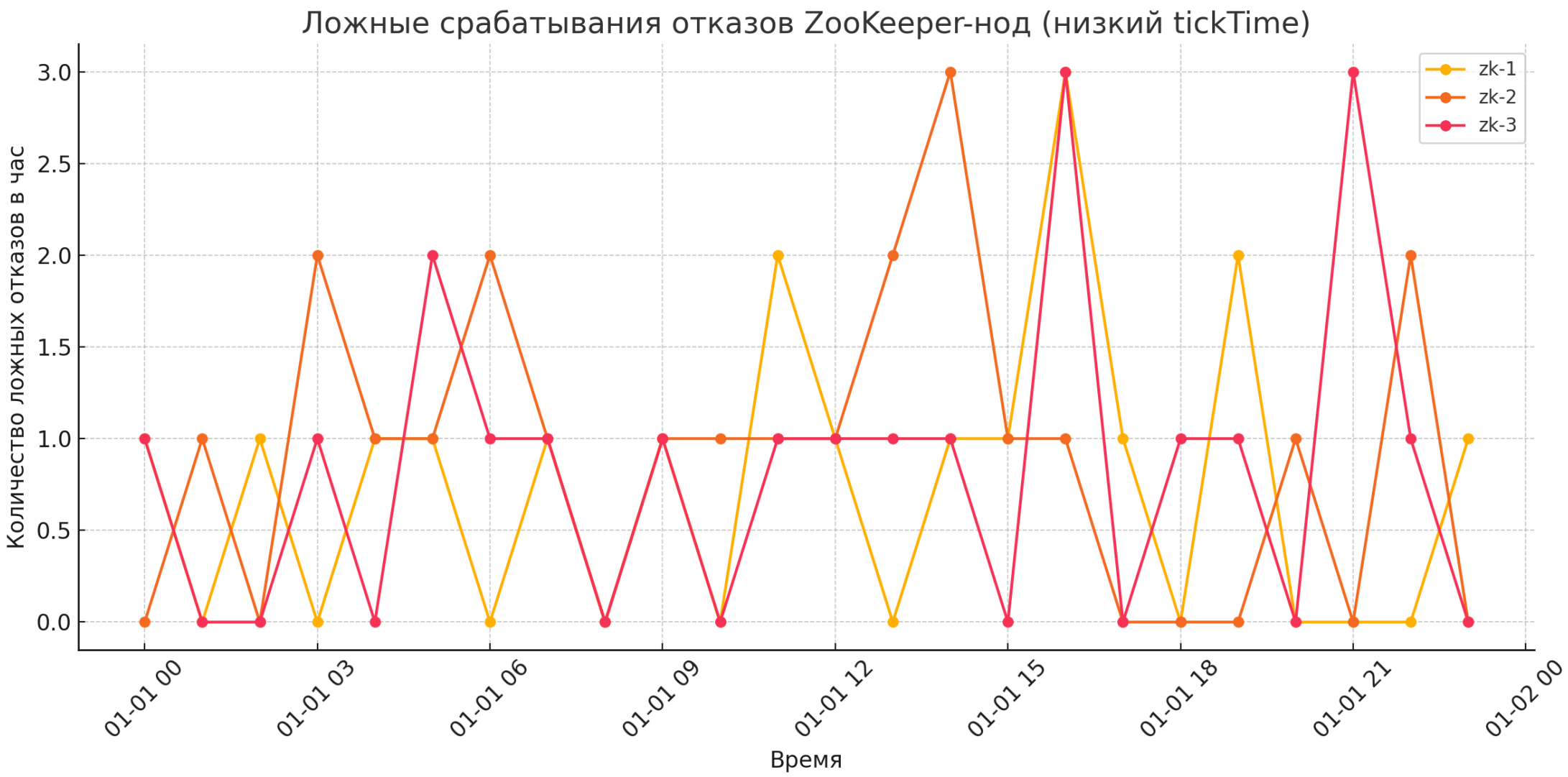


Неоптимальные настройки JVM



**Неверное значение `ticktime`**

# Решение кейса: zookeeper



# Решение кейса



## Реконфигурация брокеров

# Решение кейса



Реконфигурация брокеров



**Оптимизация retention**

# Решение кейса



Реконфигурация брокеров



Оптимизация retention



**Оптимизация партиционирования**



# Решение кейса

**Было:**

`user_id`

**Стало:**

`tenant_id + hash(event_type + timestamp)`

# Решение кейса



Реконфигурация брокеров



Оптимизация retention



Оптимизация партиционирования



**Настройки zookeeper**

# AGENDA

Введение

Инструменты мониторинга

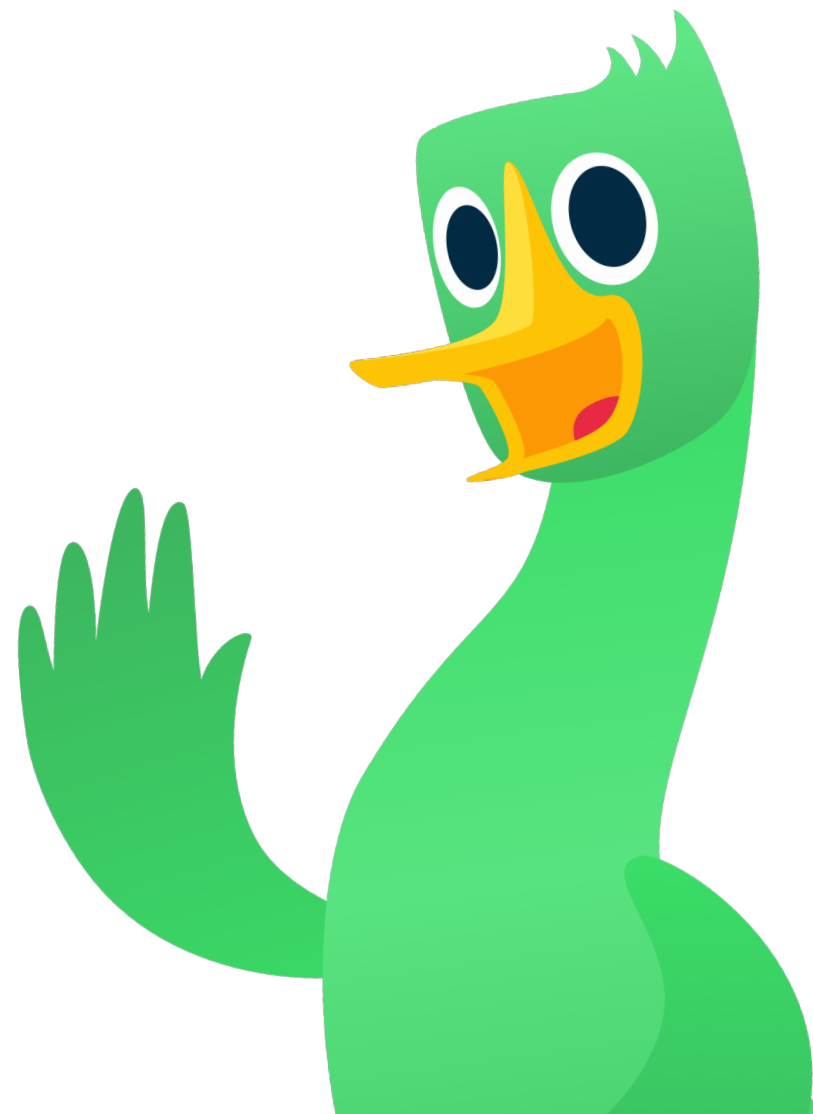
Кейс: Kubernetes и микросервисы

Кейс: Kafka

**Методологии предотвращения**



Выводы





Проактивный  
мониторинг



Перцентили времени ответа



Проактивный  
мониторинг

⚡ Перцентили времени ответа

⚡ Тренды и аномалии



Проактивный  
мониторинг

⚡ Перцентили времени ответа

⚡ Тренды и аномалии

⚡ **Ресурсные метрики**



## Проактивный мониторинг

⚡ Перцентили времени ответа

⚡ Тренды и аномалии

⚡ Ресурсные метрики

⚡ **Бизнес-метрики**



**Graceful  
Degradation**



**Приоритизация функциональности**





**Graceful  
Degradation**



Приоритизация функциональности



**Circuit Breaker**



## Graceful Degradation

⚡ Приоритизация функциональности

⚡ Circuit Breaker

⚡ Стратегии Timeout и Retry



## Graceful Degradation

⚡ Приоритизация функциональности

⚡ Circuit Breaker

⚡ Стратегии Timeout и Retry

⚡ **Ограничение ресурсов**



Принципы  
проектирования



Изоляция отказов



## Принципы проектирования

⚡ Изоляция отказов

⚡ Асинхронность и буферизация



## Принципы проектирования

⚡ Изоляция отказов

⚡ Асинхронность и буферизация

⚡ **Предсказуемость под нагрузкой**



## Принципы проектирования

⚡ Изоляция отказов

⚡ Асинхронность и буферизация

⚡ Предсказуемость под нагрузкой

⚡ Учитывайте задержки



**Best  
Practices**



**Мониторинг в непрерывном режиме**





## Best Practices



Мониторинг в непрерывном режиме



**Тестирование отказоустойчивости**



## Best Practices



Мониторинг в непрерывном режиме



Тестирование отказоустойчивости



**Постепенное внедрение изменений**



## Best Practices

⚡ Мониторинг в непрерывном режиме

⚡ Тестирование отказоустойчивости

⚡ Постепенное внедрение изменений

⚡ **Документирование и обучение**

# AGENDA

Введение

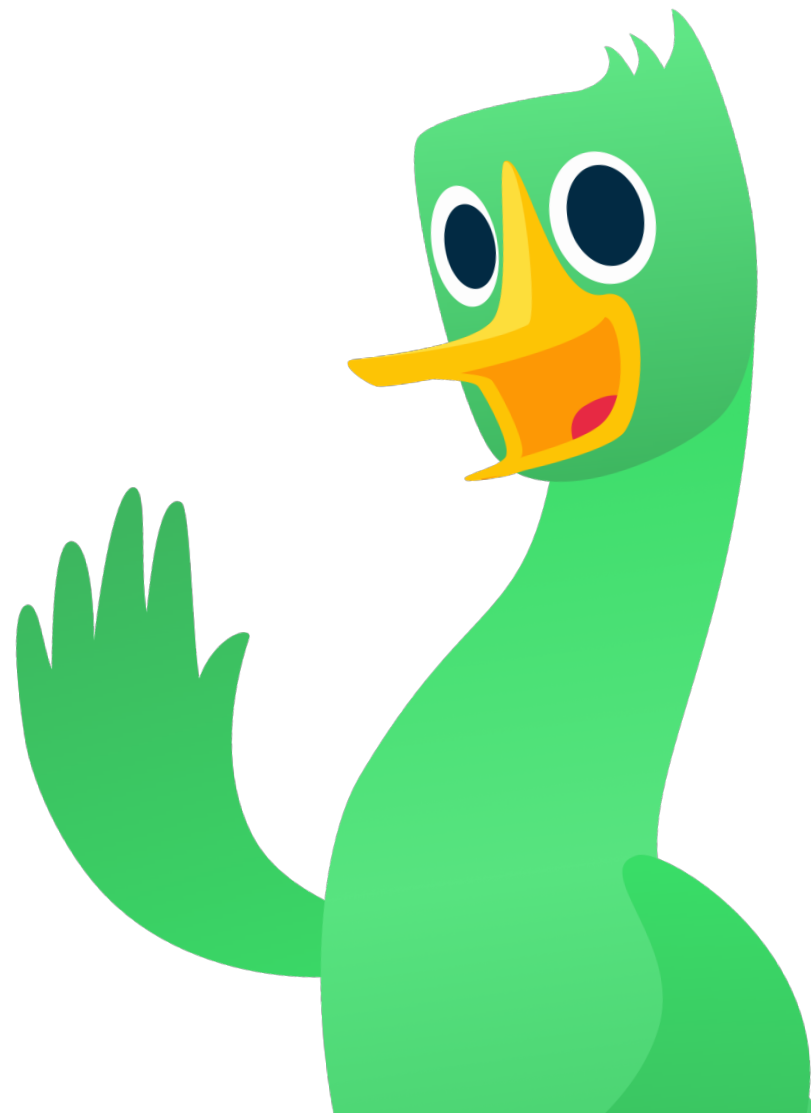
Инструменты мониторинга

Кейс: Kubernetes и микросервисы

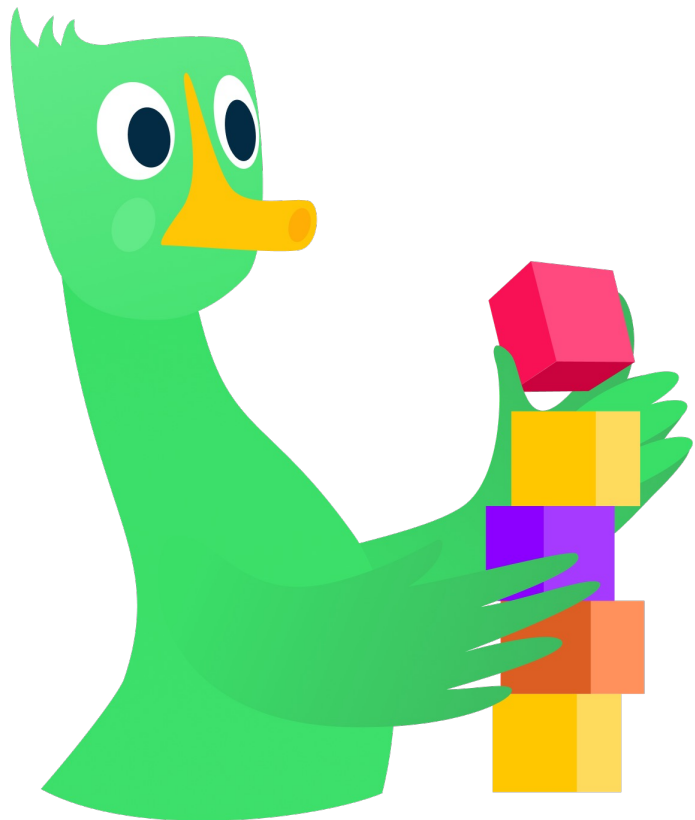
Кейс: Kafka

Методологии предотвращения

**Выводы**



# Выводы



⚡ **Метастабильные отказы опаснее полных**

# Выводы



⚡ **Метастабильные отказы опаснее полных**

⚡ **Мониторинг критически важен**

# Выводы



⚡ **Метастабильные отказы опаснее полных**

⚡ **Мониторинг критически важен**

⚡ **Каскадные эффекты взаимозависимостей**

# Выводы



⚡ **Метастабильные отказы опаснее полных**

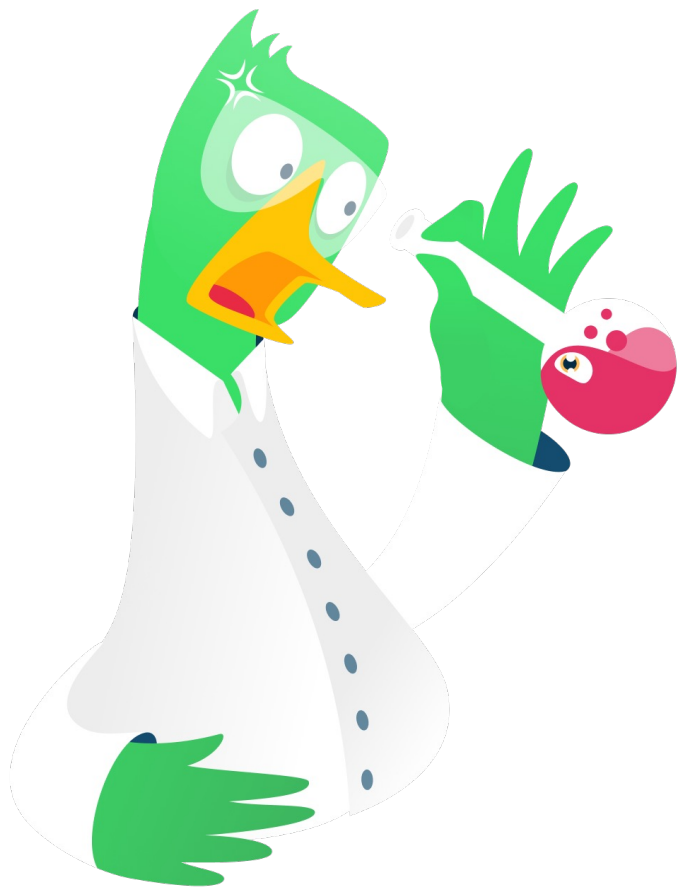
⚡ **Мониторинг критически важен**

⚡ **Каскадные эффекты взаимозависимостей**

⚡ **Учитывать деградацию**



# Выводы



⚡ **Метастабильные отказы опаснее полных**

⚡ **Мониторинг критически важен**

⚡ **Каскадные эффекты взаимозависимостей**

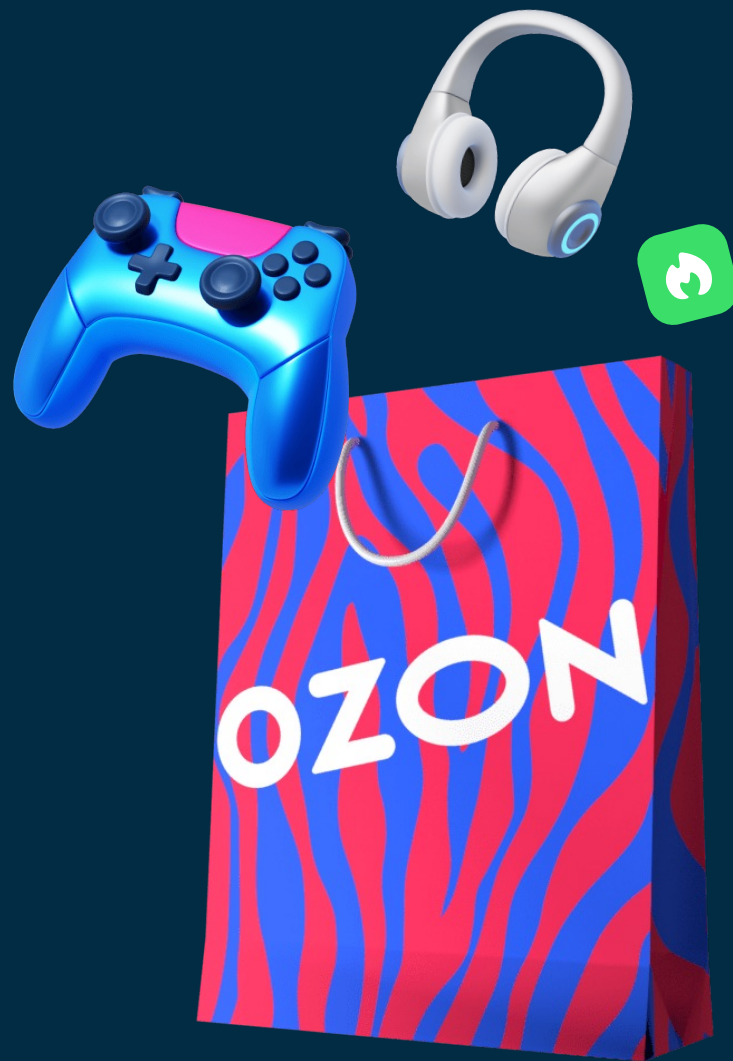
⚡ **Учитывать деградацию**

⚡ **Проактивное тестирование**



# Приходите к нам работать!

Мы ищем крутых разработчиков, девопсов и SRE  
в команду Платформы OZON





Спасибо  
за внимание!  
Вопросы?



**Андрей Синицын**  
Руководитель отдела MessageBus



 <https://andrey.sinits.in>

 [andrey@sinits.in](mailto:andrey@sinits.in)