

# StatsHouse: метрики ВКонтакте

Григорий Петросян



**HighLoad<sup>++</sup>**  
2022



# Hello, World

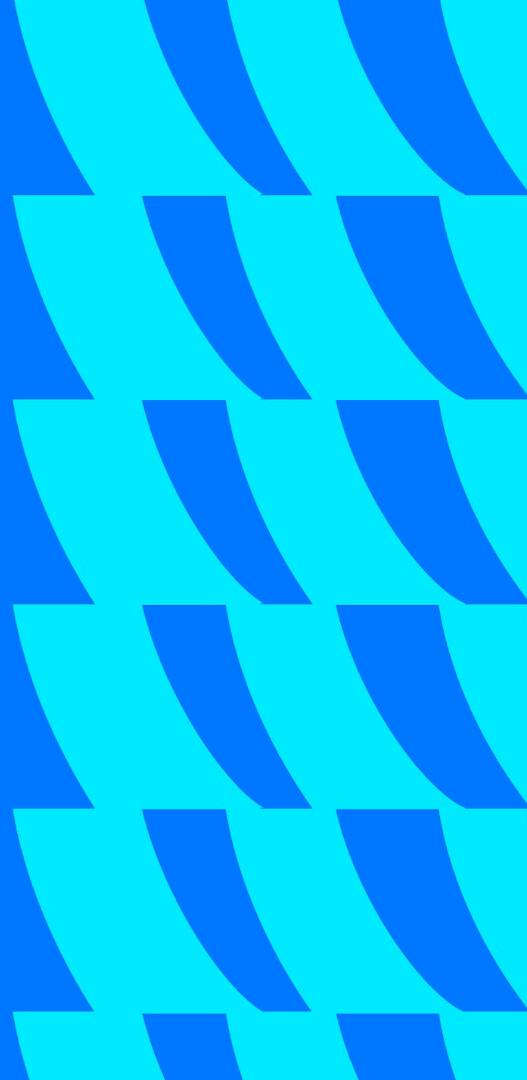
Я — Гриша, Core Infrastructure @ ВКонтакте

StatsHouse — наша система мониторинга

# План

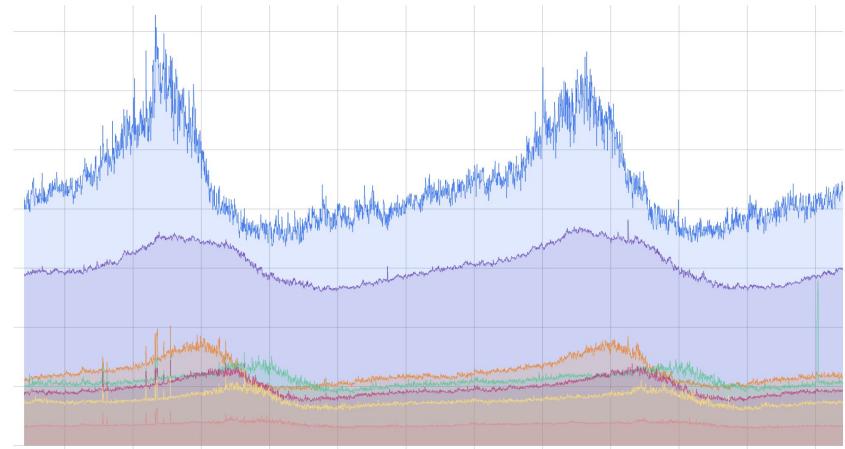
1. Про мониторинг *at scale*
2. Как эффективно собирать и хранить метрики
3. Как не дать сломать мониторинг
4. Как устроен StatsHouse
5. Анонс!

# Мониторинг *at scale*



# Мониторинг

```
cdn_ttfb {  
    country = ru  
    region = msk  
    os = android  
} = 0.239
```



# Без мониторинга



## Мониторинг

Обязан  
работать всегда

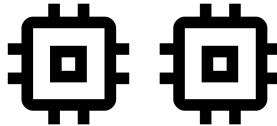
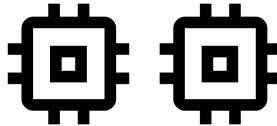
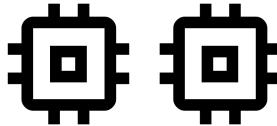
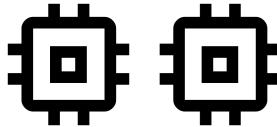
Обязан  
работать быстро

- Низкая задержка
- Высокое разрешение
- Интерактивность

# Мониторинг? Prometheus



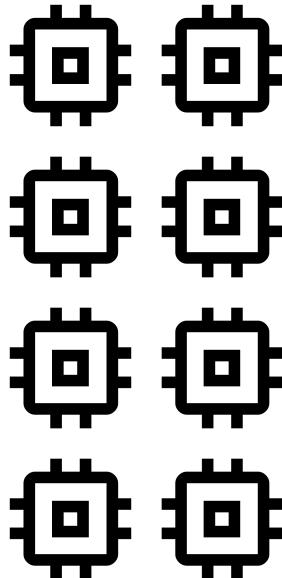
×



# Prometheus не масштабируется



× 100



**350 000 000**

метрик в секунду с 15 000 серверов собирает StatsHouse



## Мониторинг *at scale*



Унести все, за 1 секунду —  
каждую секунду

Быстрый путь  
в яхт-клуб домой



# Метрики прекрасно сжимаются

```
cdn_ttfb {  
    region = msk/spb/nsk  
} ← (count: 1, value: dt)
```

6M RPS

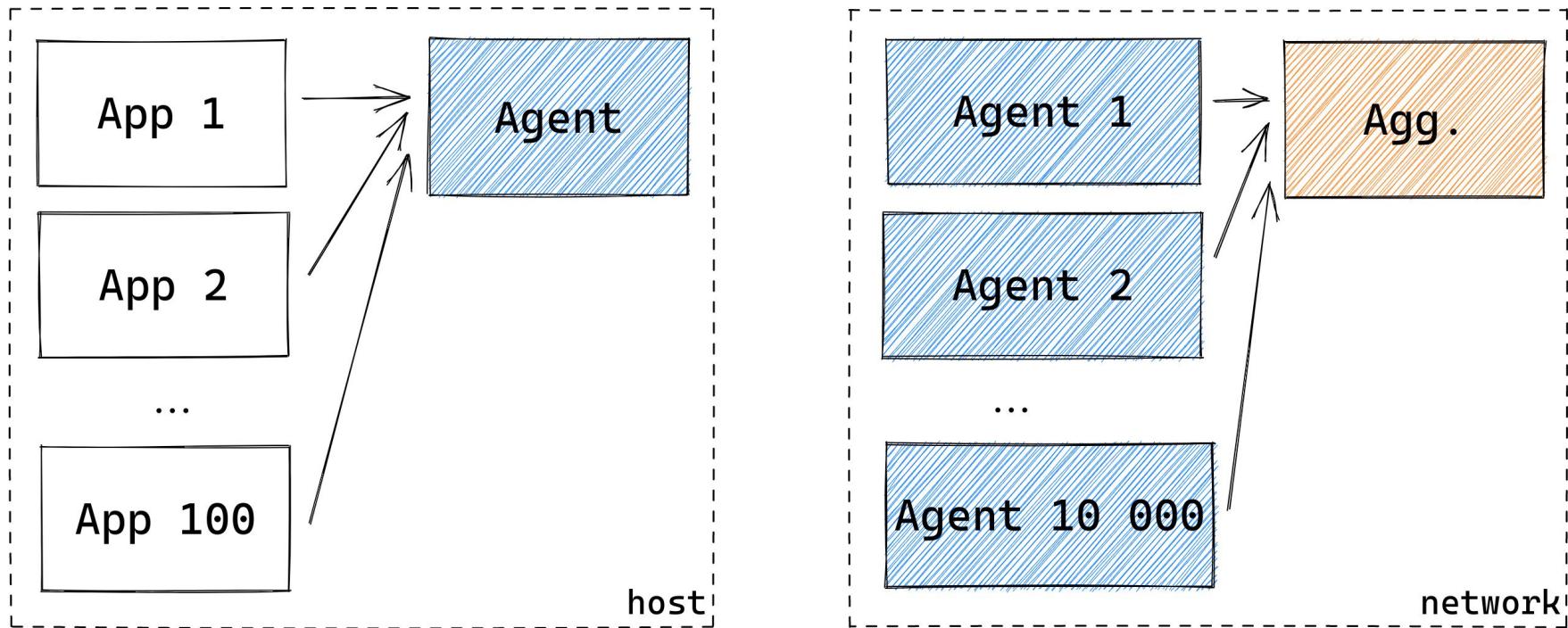
6M записей в секунду

```
cdn_ttfb{msk} ← (3M, Σdt)  
cdn_ttfb{spb} ← (2M, Σdt)  
cdn_ttfb{nsk} ← (1M, Σdt)
```

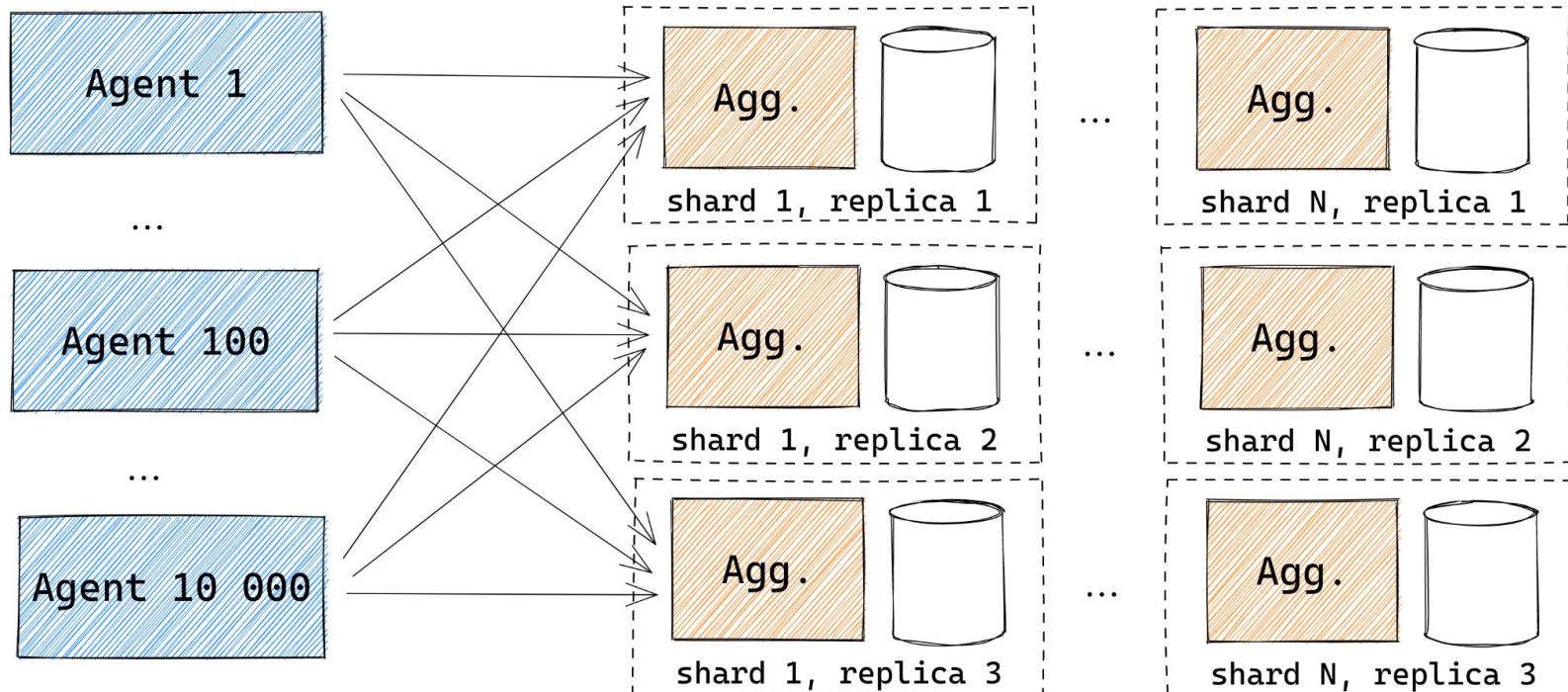
6M RPS

3 записи в секунду

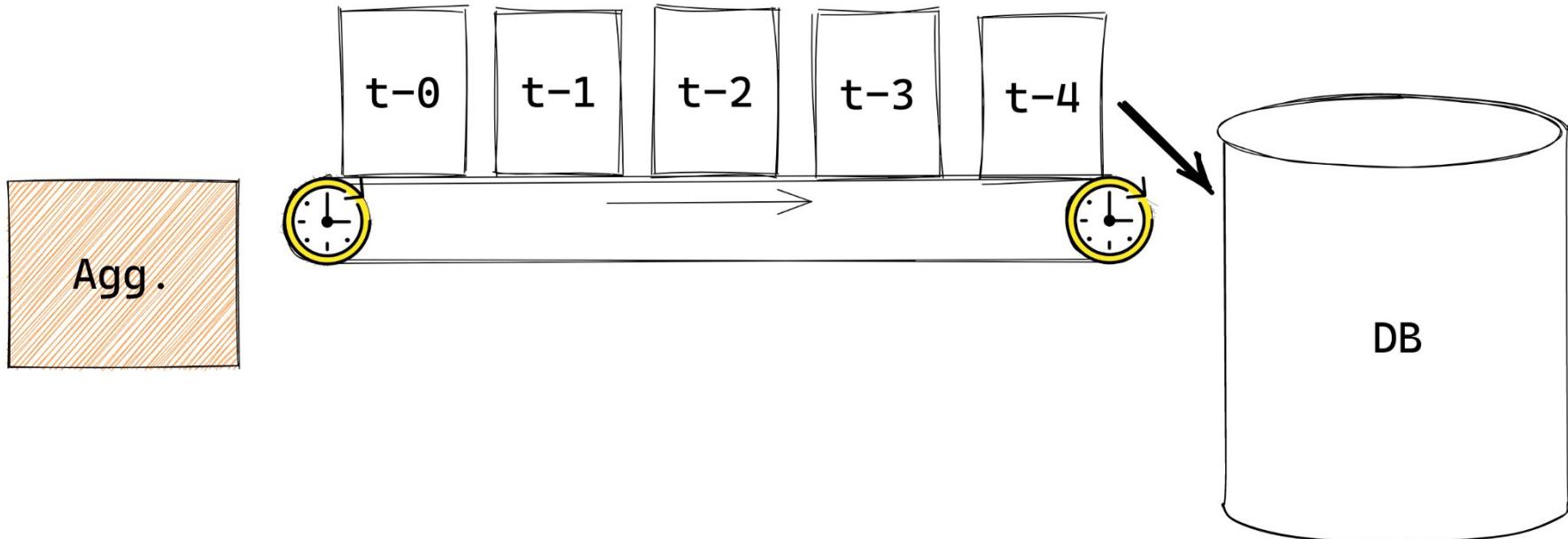
# Многоуровневая агрегация



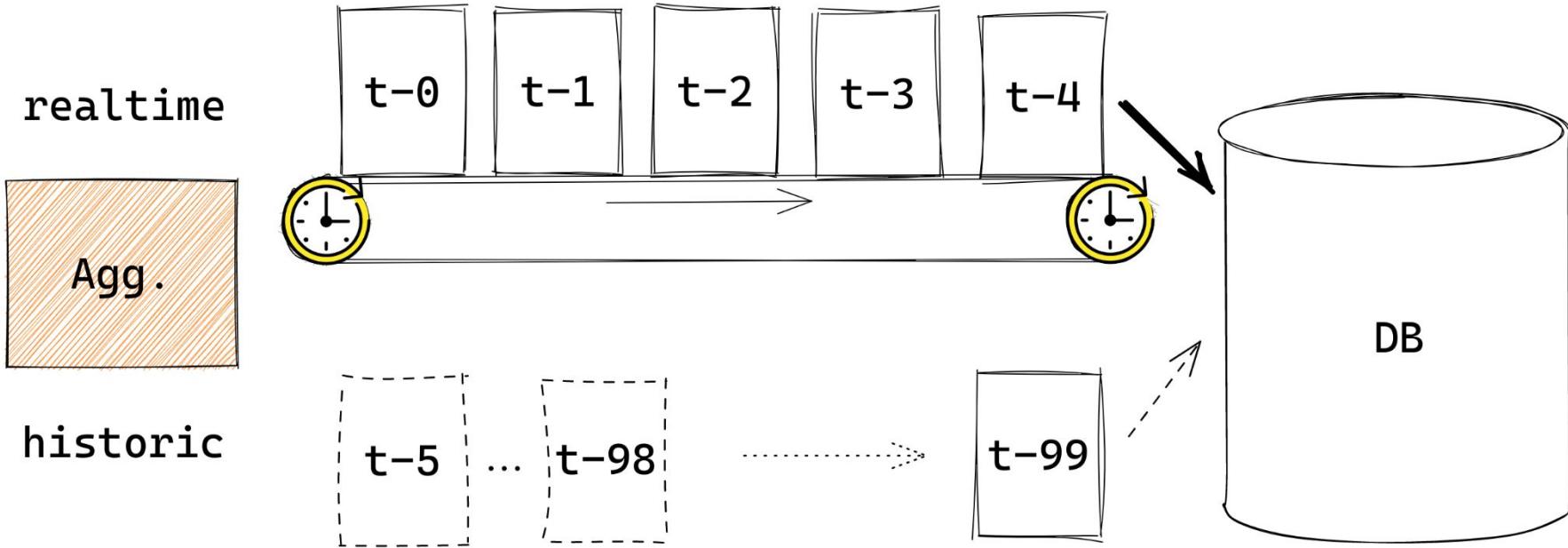
# Шардирование и репликация



# Конвейер записи



# Конвейер записи: исторические данные



**Не дать сломать  
мониторинг**



# Метрики прекрасно сжимаются

```
cdn_ttfb {  
    region = msk/spb/nsk  
} ← (count: 1, value: dt)
```

6M RPS

6M записей в секунду

```
cdn_ttfb{msk} ← (3M, Σdt)  
cdn_ttfb{spb} ← (2M, Σdt)  
cdn_ttfb{nsk} ← (1M, Σdt)
```

6M RPS

3 записи в секунду

# Метрики прекрасно разжимаются

foo{...}

100 Мбайт/с



foo{..., \$server}

100 Гбайт/с



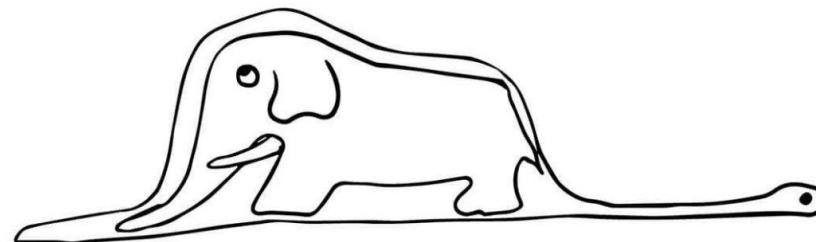
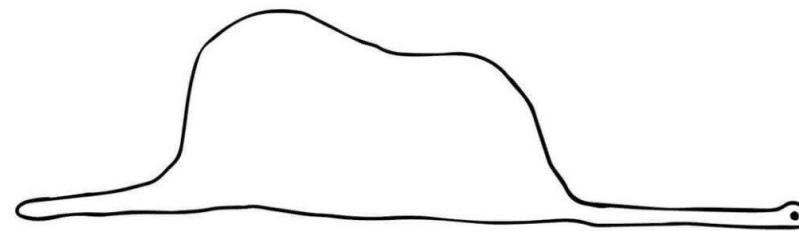
# Метрики прекрасно бесконечно разжимаются

foo{..., \$user\_id, \$url}

100 Тбайт/с



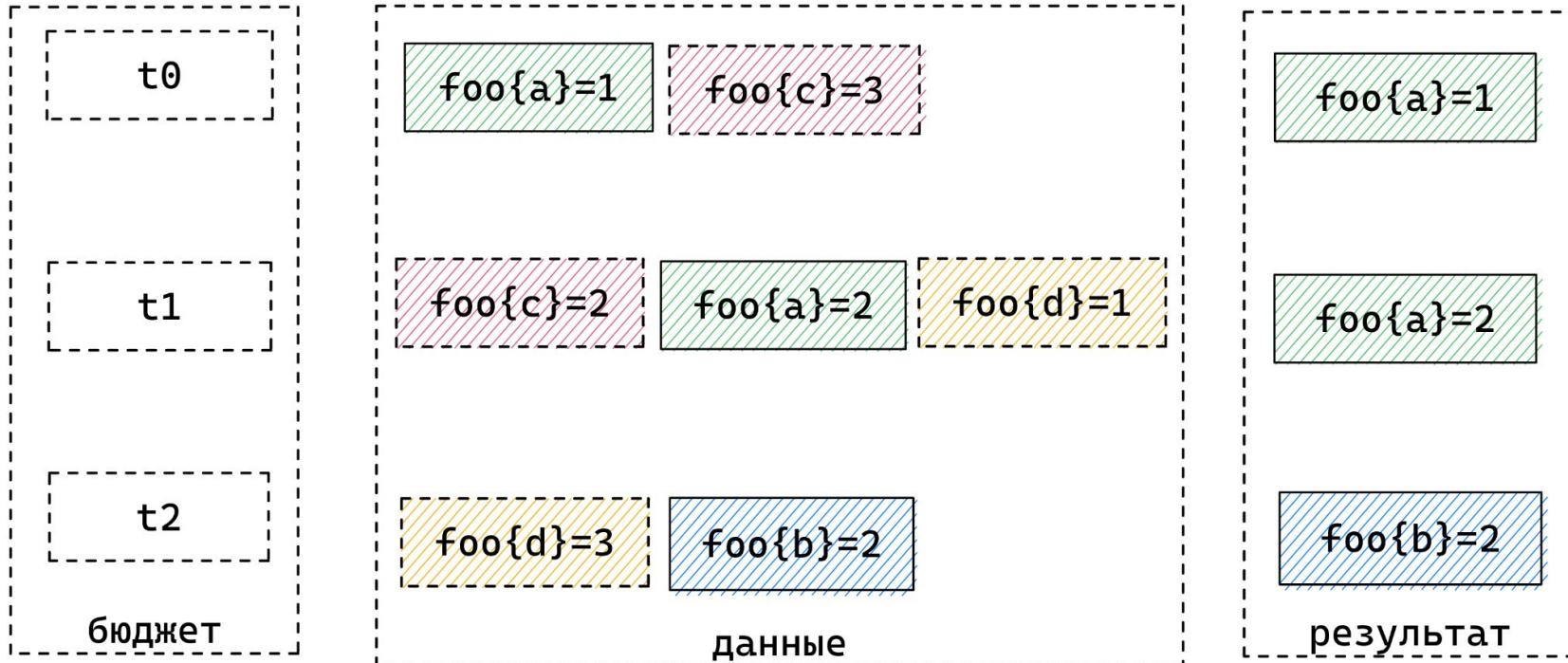
# Kafka не спасет



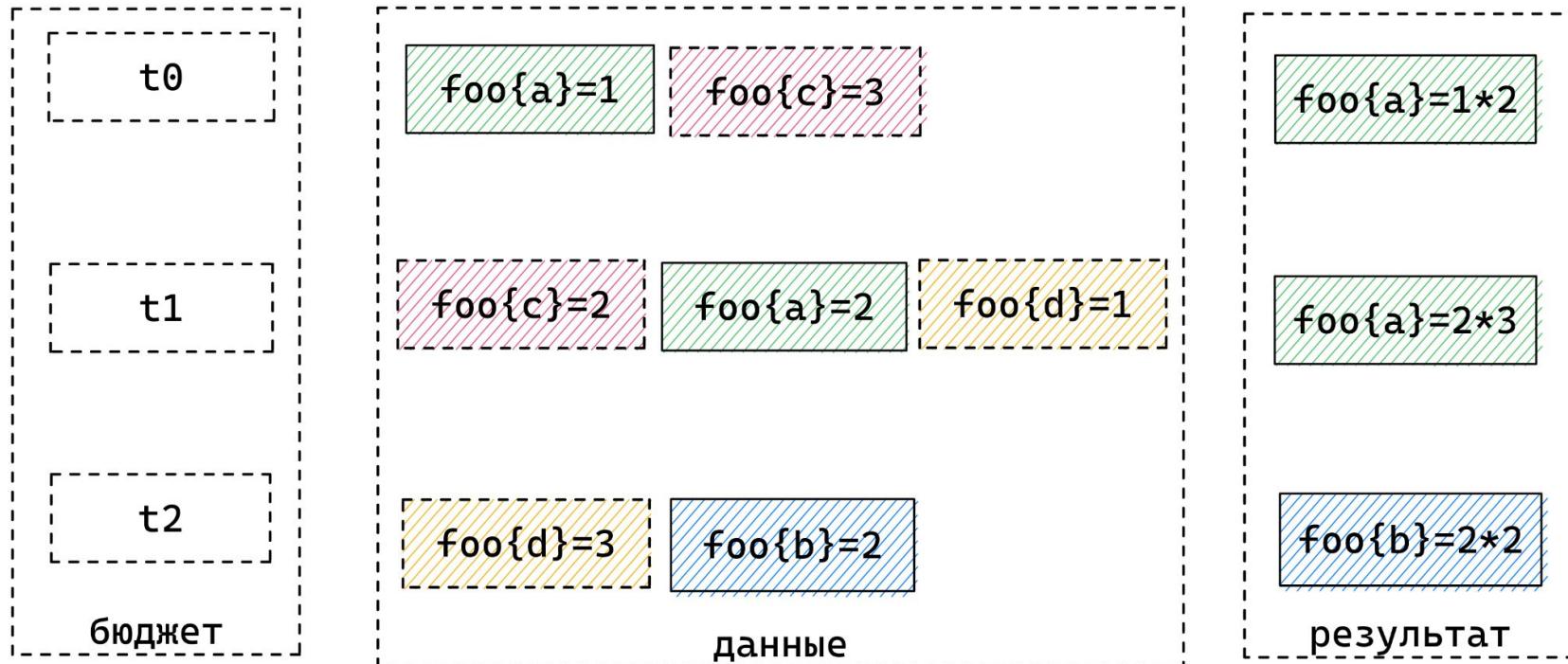
# Боремся с перегрузкой



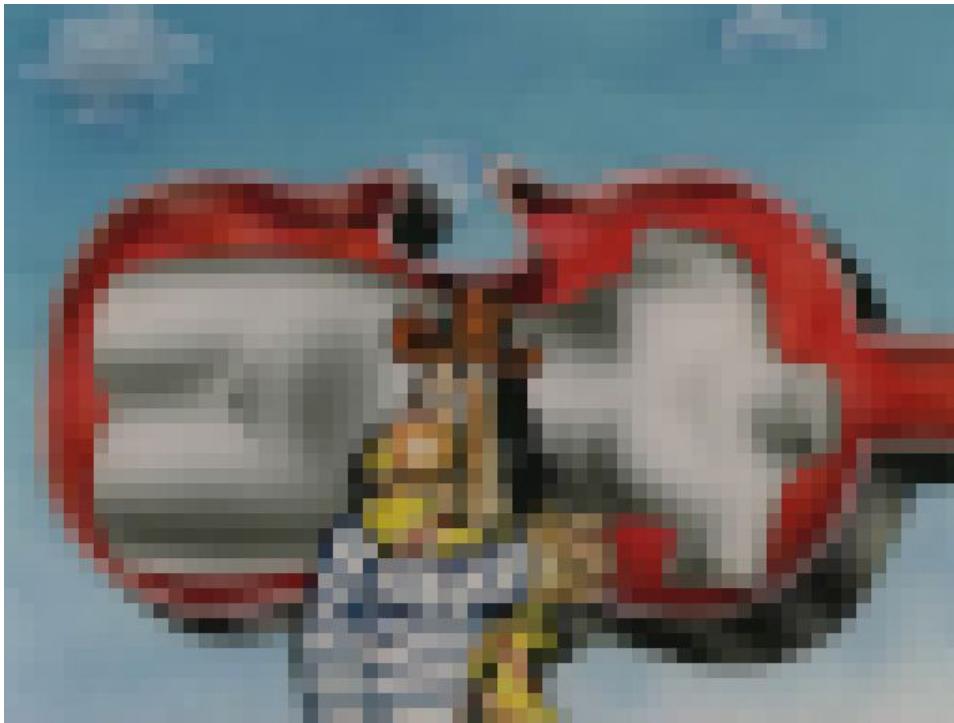
# cat data > /dev/null



# Семплирование



# Семплирование работает



# Не дать сломать мониторинг ничего

1. Бюджетируем ресурсы
  - Сеть
  - Диск
  - CPU
2. Честно распределяем между пользователями
3. Авто-семплируем при превышении бюджета

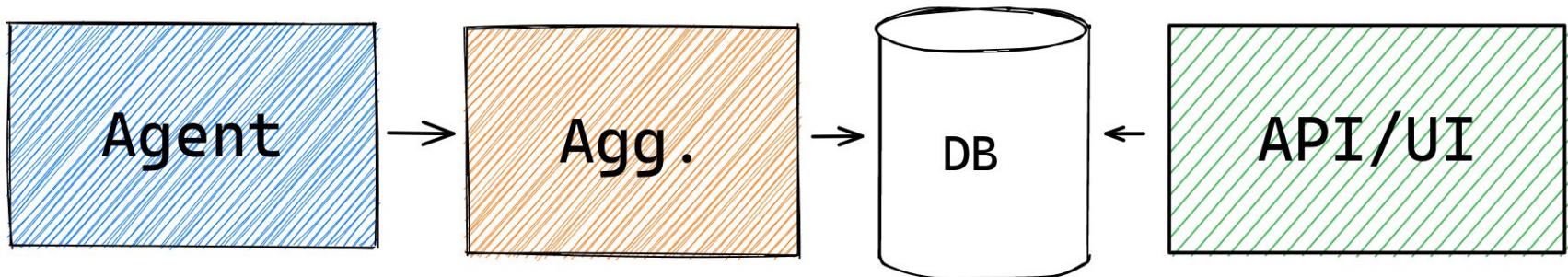
*“Look Ma,  
a Highly-Available  
Multi-Tenant System!”*



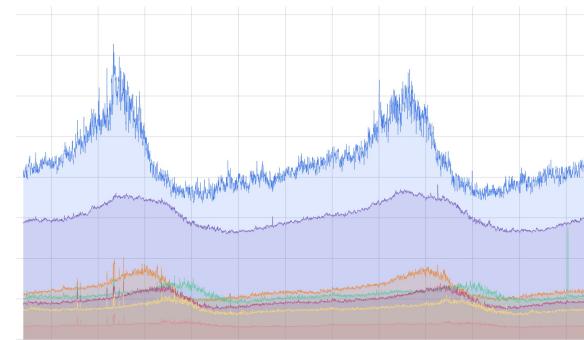
# Как устроен StatsHouse



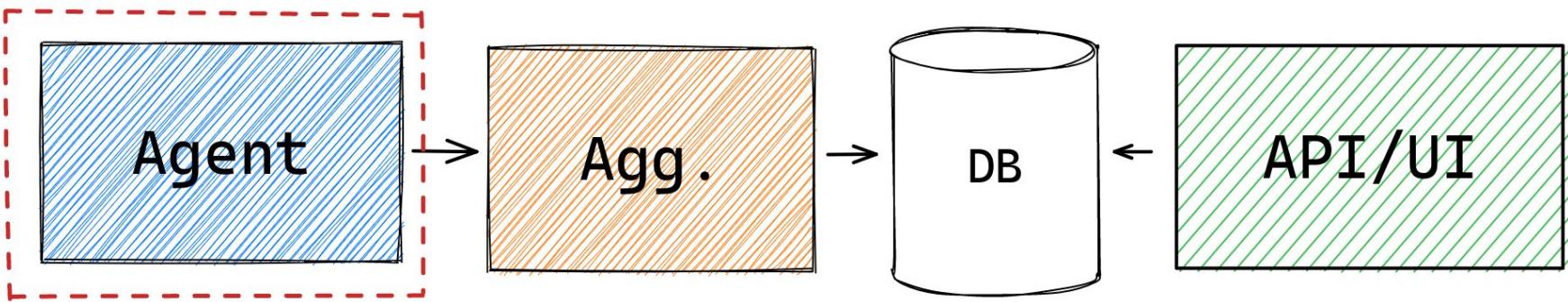
# StatsHouse



```
cdn_ttfb {  
    country = ru  
    region = msk  
    os = android  
} = 0.239
```

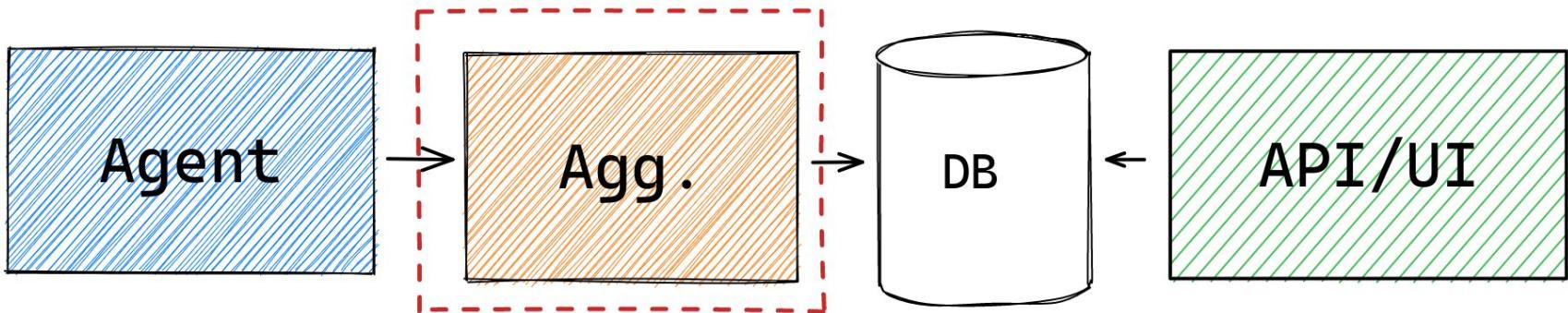


# StatsHouse: агент



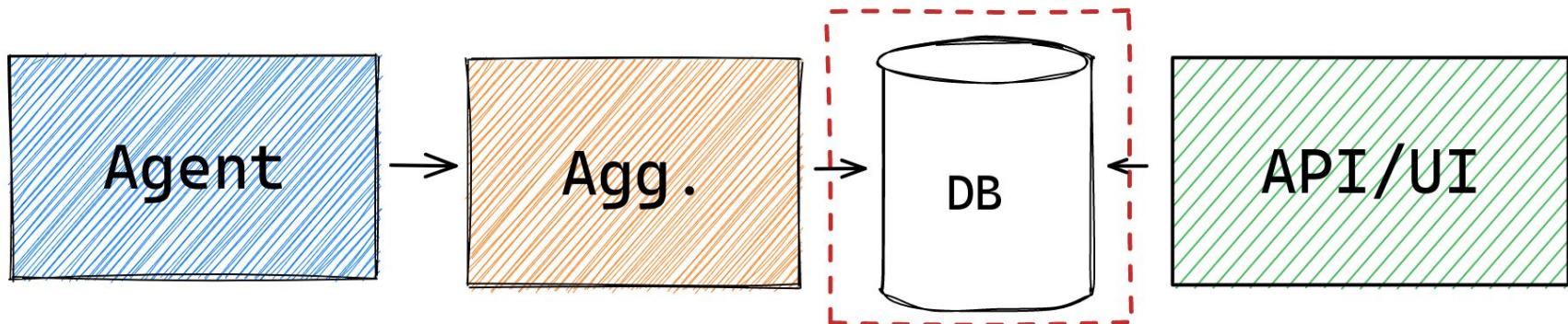
- Агрегация между процессами
- Бюджетирование + авто-семплирование
- Шардирование + Failover
- Локальное хранение

# StatsHouse: агрегатор



- Агрегация между агентами
- Бюджетирование + авто-семплирование
- Приоритизация
- Backpressure

# StatsHouse: база



- Репликация
- Downsampling
- Хранение + TTL
- Выполнение запросов

# ClickHouse как основное хранилище

- Эксплуатация умеет в ClickHouse
- ClickHouse масштабируется
- ClickHouse не тормозит™\*

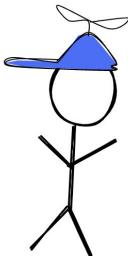
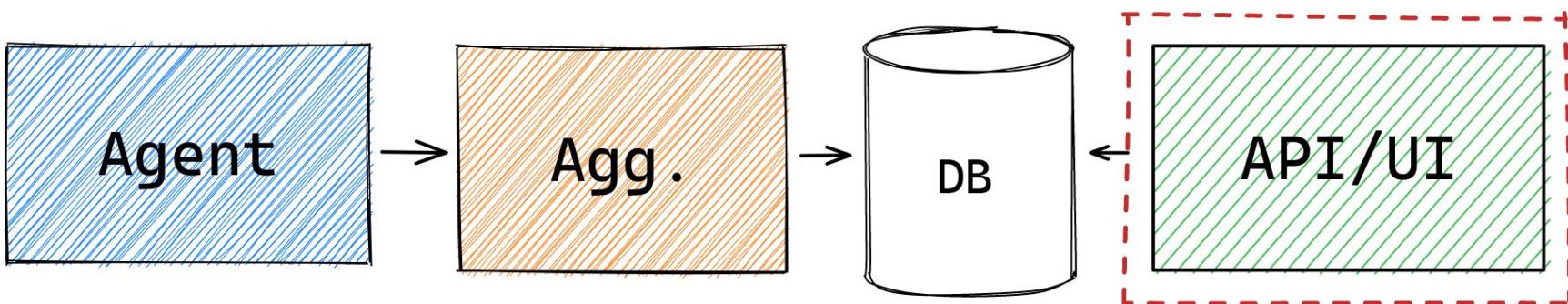


# Схема хранения

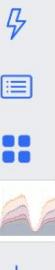
time	metric	tag1	tag2	count
01:14:21	cdn_ttfb	msk	android	3015682
01:14:21	cdn_ttfb	msk	ios	936581
01:14:21	cdn_ttfb	nsk	android	1025718
01:14:21	cdn_ttfb	nsk	ios	315981
01:14:21	cdn_ttfb	spb	android	2390001

- Только 3 таблицы
  - Секундная
  - Минутная
  - Часовая
- Только 16 тегов
- Числа вместо строк

# StatsHouse: API + UI



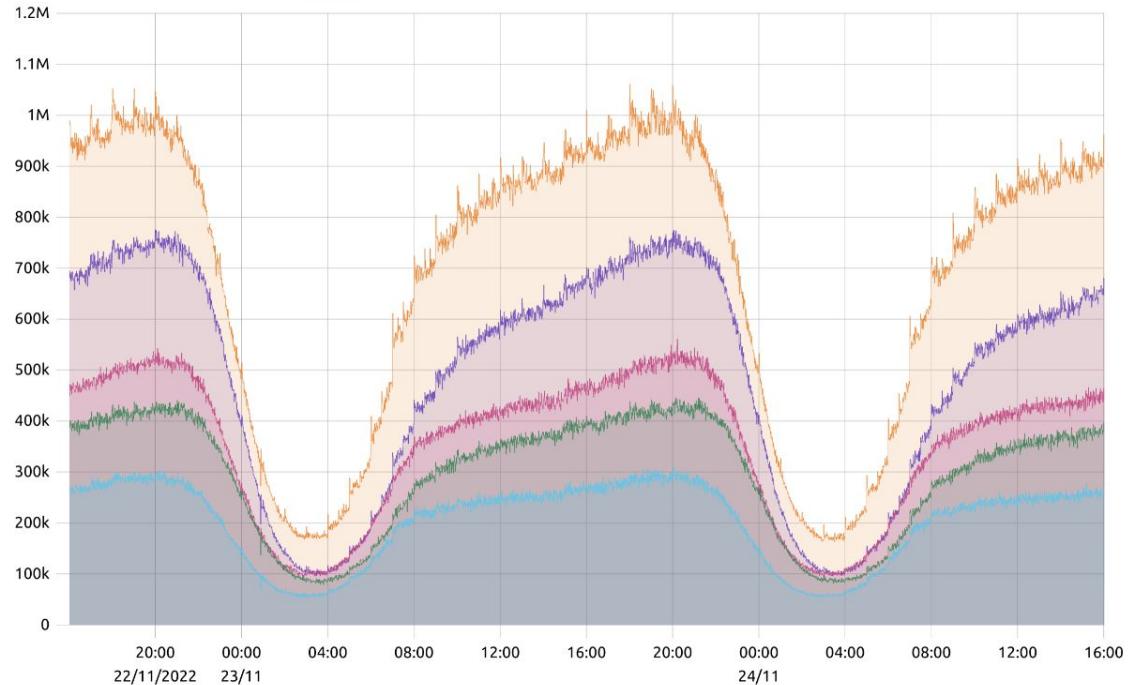
- Контроль доступа
- Ограничение нагрузки
- Кеширование



api\_methods: count/sec

Вызовы API (без учета execute.\* вызовов)

Receive status Sampling: **source** / aggregator Cardinality Mapping status CSV Edit



Time —

account —

messages —

stats —

< > + - ⌂ 🔍 🔍 ⌂ 🔒 🔒 ▶

api\_methods

count/sec

Last 48 hours

Today Week

24/11/2022

16:01:43



-24h

-48h

-1w

-2w

-3w

-4w

-365d

Auto

Top 5



environment



production

Группа методов



метод



версия

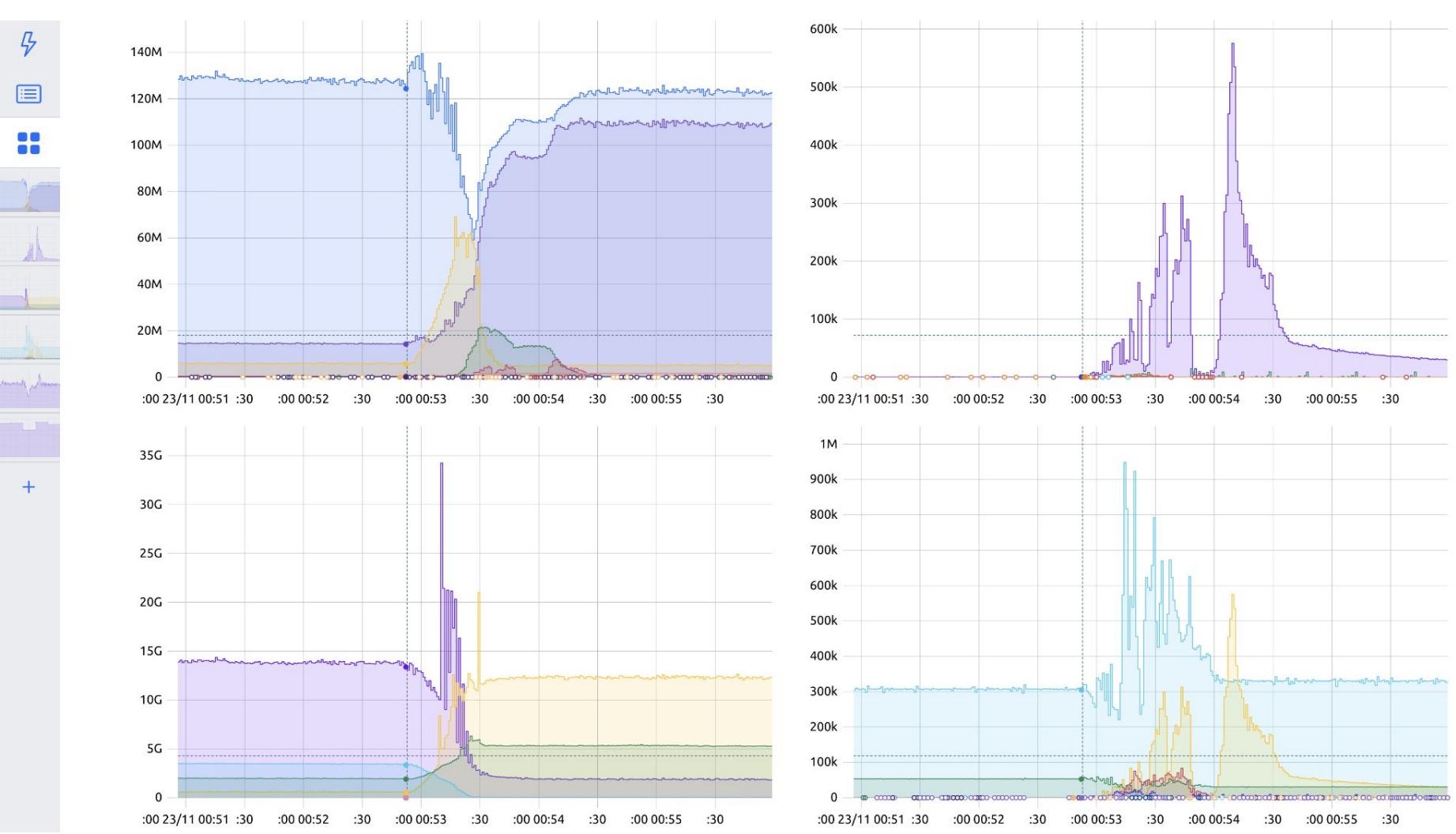


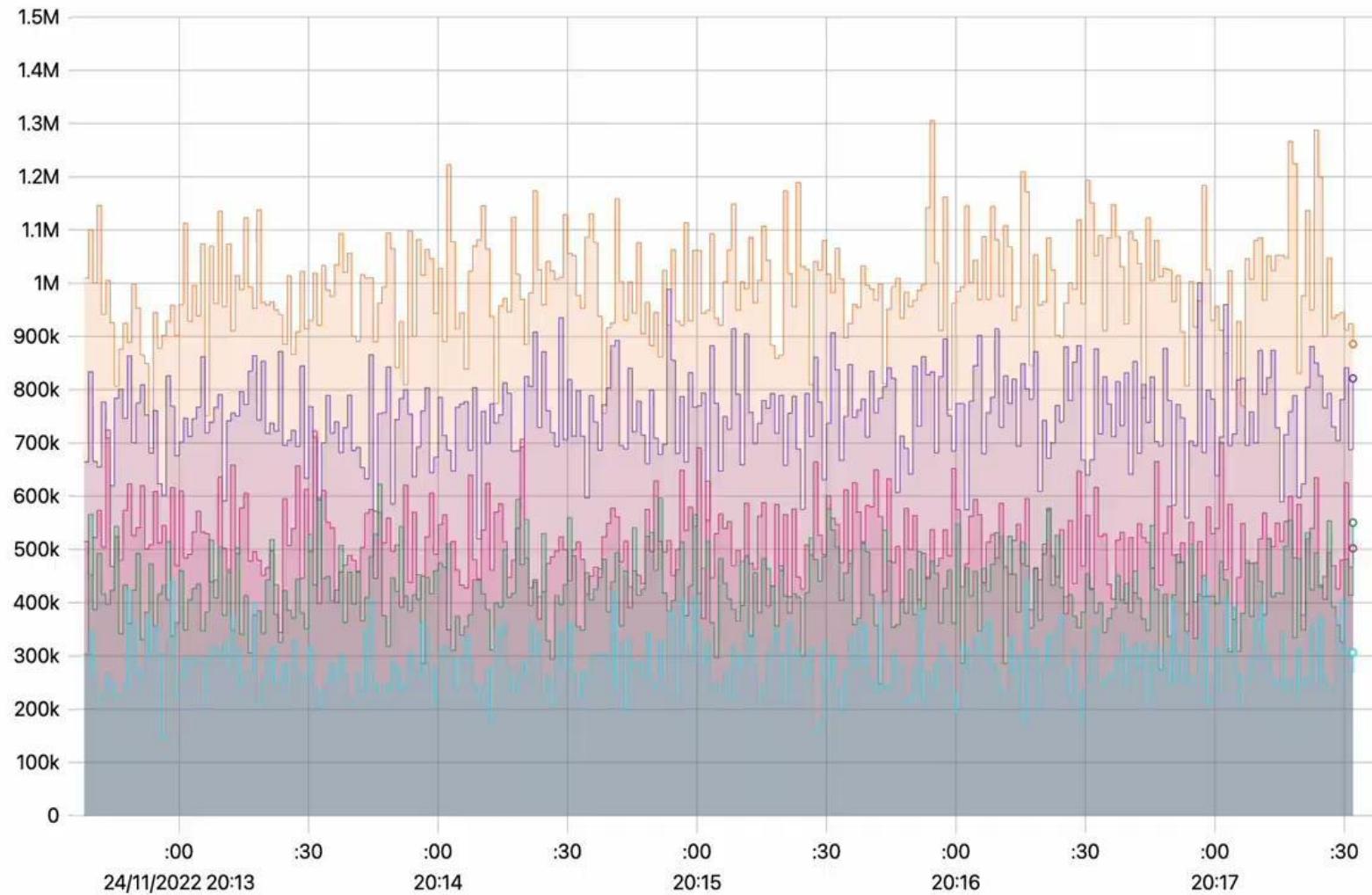
версия minor



платформа







# **StatsHouse сегодня**

**1 секунда**

разрешение

**5 секунд**

задержка

**4 года**

данных

**350 000 000**

метрик в секунду

**15 000**

серверов-источников

# Анонс!



# StatsHouse — Open Source

[github.com/vkcom/  
statshouse](https://github.com/vkcom/statshouse)

MPL 2.0



# StatsHouse

[github.com/vkcom/statshouse](https://github.com/vkcom/statshouse)



Григорий Петросян  
[vk.com/gp](https://vk.com/gp)  
[t.me/pgregory](https://t.me/pgregory)

