LAZADA TECH HUB

# Prometheus

## мониторинг от души

Evgeny Pak,
Senior Go Developer

DevConf

# Lazada

# Как мы «докатились» до Prometheus или проблемы стороннего сервиса метрик


librato.

- Ограниченный объем метрик
- Ограниченные возможности группировки
- Неудобный язык запросов
- Закрытый код
- Большая зависимость от стороннего сервиса
- Платный сервис

# Prometheus

## Хранение даных: TimeSeries

**Каждая метрика сохраняется в отдельный файл**

| TimeStamp | Float 64 |
|---|---|
| 1.394632178e+09 | 10.00 |
| 1.394632901e+09 | 11.00 |
| 1.394632101e+09 | 12.00 |
| 1.39463201e+09 | 13.00 |

# Структура метрик

librato.

**Имя метрики**

http-requests-total.my-handler.post.500

**Prometheus**

**Имя метрики + Labels(аттрибуты)**

http_requests_total{method="post",handler="my-handler",status="500"}

# Labels Prometheus под капотом

http_requests_total{method="post", handler="my-handler1",status="200"} => 10

http_requests_total{method="post", handler="my-handler1",status="500"} => 11

http_requests_total{method="post", handler="my-handler2",status="200"} => 12

http_requests_total{method="post", handler="my-handler2",status="500"} => 13

# Samples Data

*Fingerprint:*

1a22b471c9c7d493 1d4336d50e89b48d 1eb43aca4aee2daf 1f2cd5dba72e2260




*File:*

1a/22b471c9c7d493.db 1d/4336d50e89b48d.db 1e/b43aca4aee2daf.db 1f/2cd5dba72e2260.db

# Labels Prometheus под капотом

## LevelDB

| Key | Value |
|---|---|
| __name__="http_requests_total" | 1a22b471c9c7d493(1)<br>1d4336d50e89b48d(2)<br>1eb43aca4aee2daf (3)<br>1f2cd5dba72e2260 (4) |
| method="post" | 1a22b471c9c7d493(1)<br>1d4336d50e89b48d(2)<br>1eb43aca4aee2daf (3)<br>1f2cd5dba72e2260 (4) |
| handler="my-handler1" | 1a22b471c9c7d493(1)<br>1d4336d50e89b48d(2) |
| handler="my-handler2" | 1eb43aca4aee2daf(3)<br>1f2cd5dba72e2260(4) |
| status="200" | 1a22b471c9c7d493(1)<br>1eb43aca4aee2daf(3) |
| status="500" | 1d4336d50e89b48d(2)<br>1f2cd5dba72e2260(4) |

# Labels Prometheus под капотом

## Нужно получить данные метрики

http_requests_total{method="post", handler="my-handler1",status="500"}

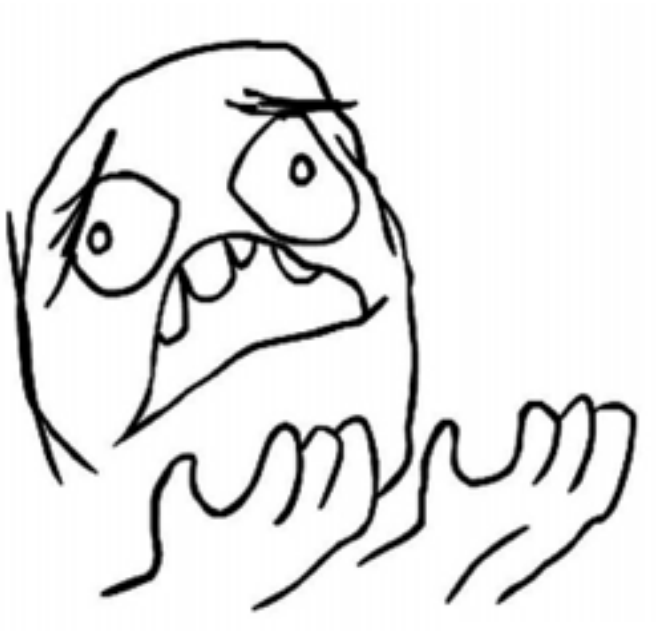
### Запрашиваем LevelDB

| __name__="http_requests_total" | method="post" | handler="my-handler1" | status="500" |
|---|---|---|---|
| 1a22b471c9c7d493 | 1a22b471c9c7d493 | 1a22b471c9c7d493 | 1d4336d50e89b48d |
| 1d4336d50e89b48d | 1d4336d50e89b48d | 1d4336d50e89b48d | 1f2cd5dba72e2260 |
| 1eb43aca4aee2daf | 1eb43aca4aee2daf | | |
| 1f2cd5dba72e2260 | 1f2cd5dba72e2260 | | |

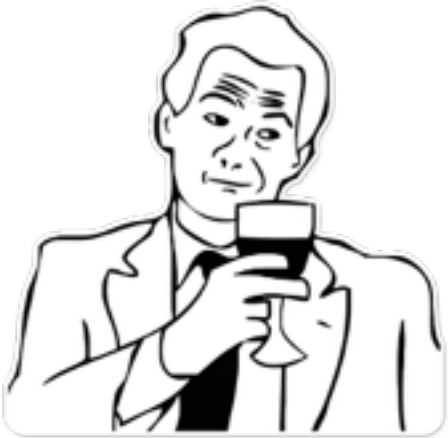
**Пересекаем данные:** 1d4336d50e89b48d

↓

**Конвертируем Fingerprint в файл:** 1d/4336d50e89b48d.db



В чем же фича?



*в агрегации….*

**Пример агрегации данных**

max(http_requests_total{method="post",status="500"} by handler

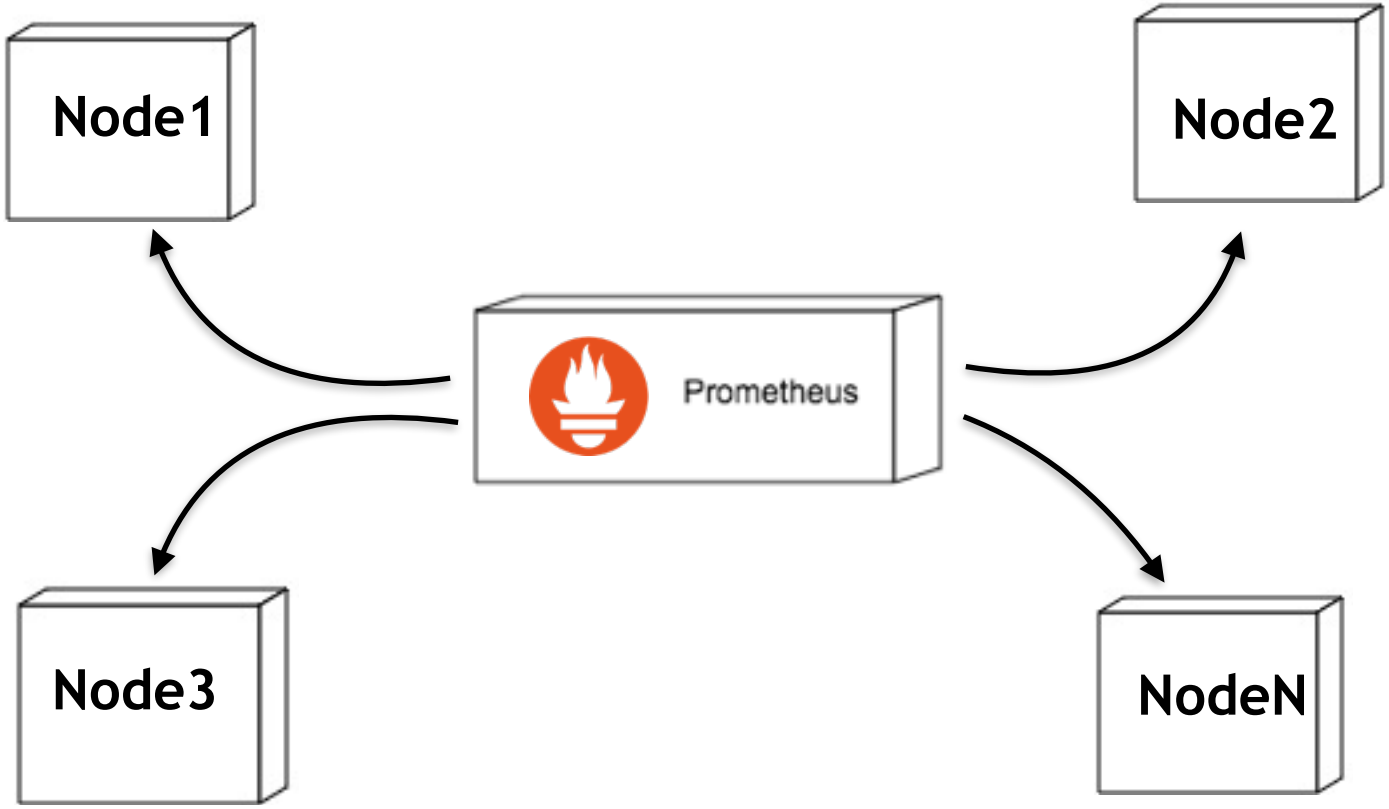
**Пример пересечения метрик**

cache.aerospike{count=«hit»} + on (job,instance) cache.aerospike {count="miss")

**Каждое уникальное значение Label представляет собой отдельную метрику**

# Prometheus killer feature

## Pull scheme




Предсказуема нагрузка

Уменьшение вероятности ошибки

## Graphite


## Prometheus

### *Хранение данных*

TimeSeries

### *Структура метрик*

Только имя метрики

*http-requests-total.my-handler.post.500*

### *Хранение данных*

TimeSeries

### *Структура метрик*

Имя метрики + Атрибуты (Labels)

*http_requests_total{method="post",handler="my-handler",status="500"}*

## InfluxDB


## Prometheus

***Хранение данных***

TimeSeries FullMetaData

***Хранение данных***

TimeSeries

# TimeSeries FullMetaData

**Каждая метрика хранит в себе информацию обо всех Labels**

| Series | time | Seq_num | handler | Status | Value |
|---|---|---|---|---|---|
| http_request_count | 1.3978e+09 | 9.2641e+09 | my_handler1 | 500 | 10 |
| http_request_count | 1.2178e+09 | 9.2642e+09 | my_handler2 | 200 | 11 |
| http_request_count | 1.3178e+09 | 9.2643e+09 | my_handler1 | 500 | 12 |
| http_request_count | 1.2178e+09 | 9.2644e+09 | my_handler2 | 200 | 15 |

**Примерная структура хранения**

## InfluxDB 


## Prometheus 

***Хранение данных***

TimeSeries FullMetaData

***Структура метрик***

Имя метрики + Атрибуты (Labels)

***Горизонтальное масштабирование***

Платное и закрытое из коробки

***Хранение данных***

TimeSeries (экономнее)

***Структура метрик***

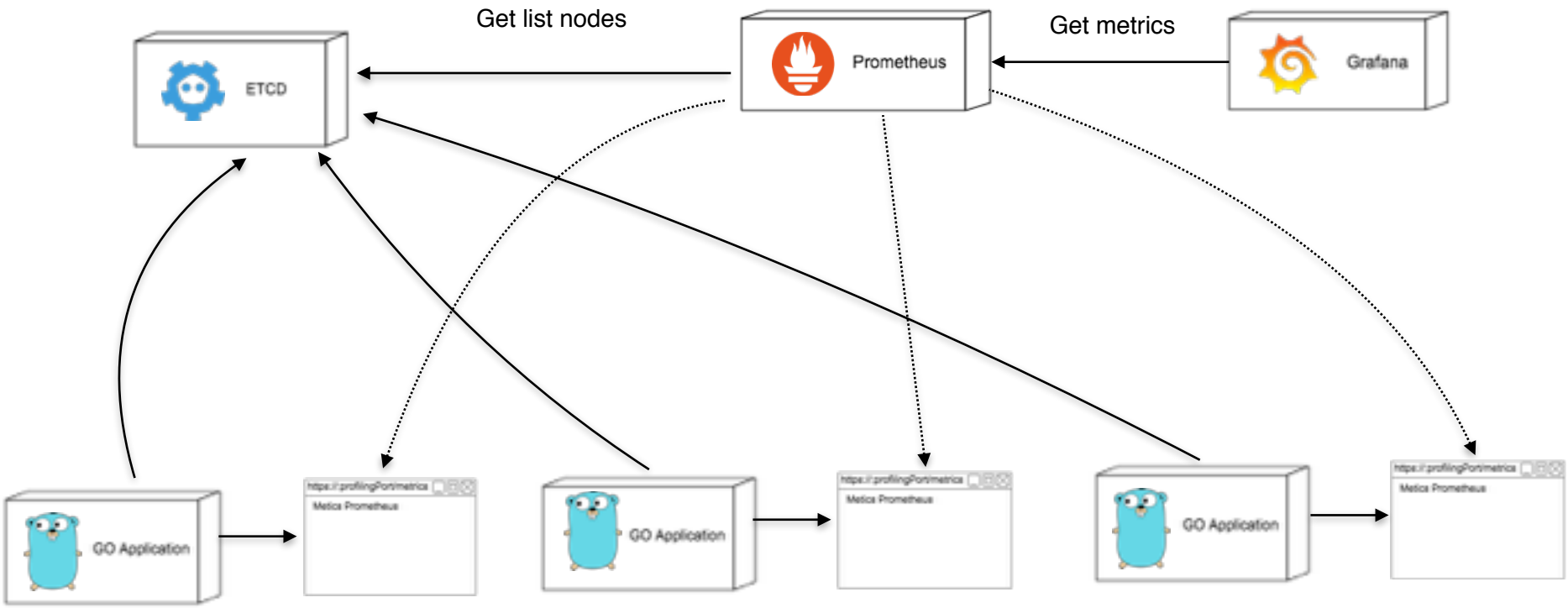
Имя метрики + Атрибуты (Labels)

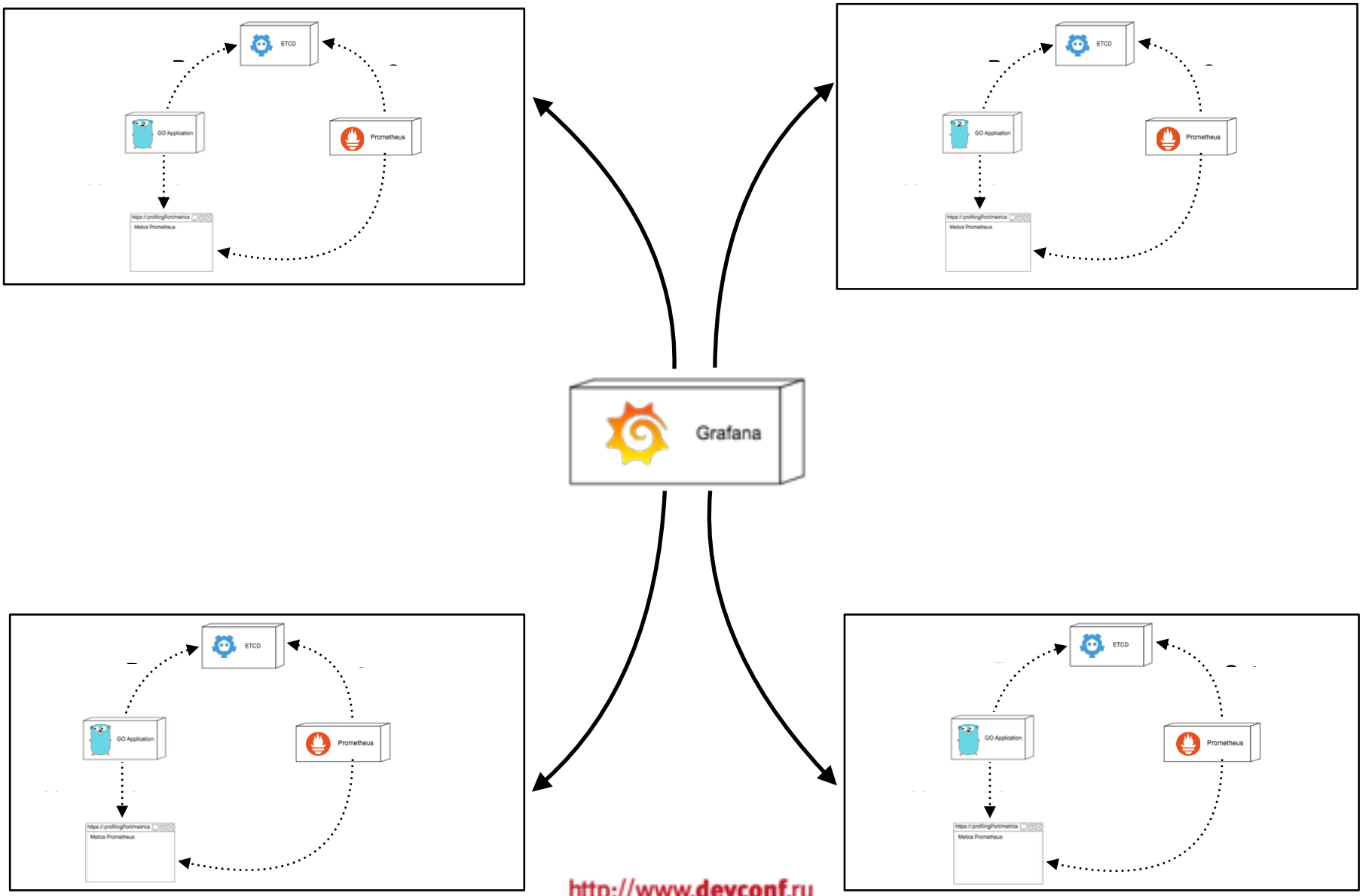
***Горизонтальное масштабирование***

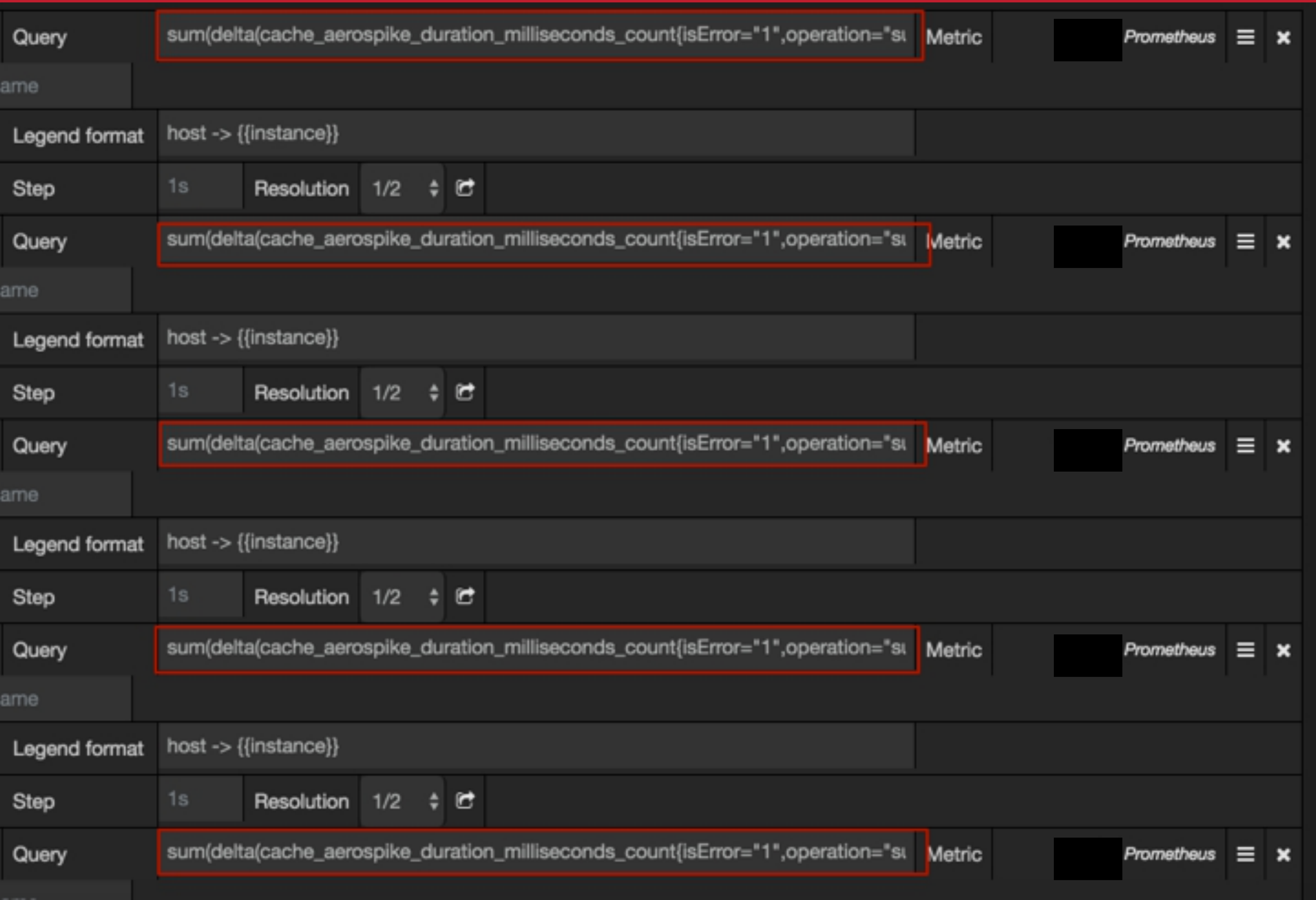
Federation

# Переезд на Prometheus

# Наша схема работы




ETCD
GO Application
Prometheus
Metics Prometheus
Grafana



Query sum(delta(cache_aerospike_duration_milliseconds_count{isError="1",operation="s Metric [redacted] *Prometheus* ☰ ✖

ame

Legend format host -> {{instance}}

Step 1s Resolution 1/2

Query sum(delta(cache_aerospike_duration_milliseconds_count{isError="1",operation="s Metric [redacted] *Prometheus* ☰ ✖

ame

Legend format host -> {{instance}}

Step 1s Resolution 1/2

Query sum(delta(cache_aerospike_duration_milliseconds_count{isError="1",operation="s Metric [redacted] *Prometheus* ☰ ✖

ame

Legend format host -> {{instance}}

Step 1s Resolution 1/2

Query sum(delta(cache_aerospike_duration_milliseconds_count{isError="1",operation="s Metric [redacted] *Prometheus* ☰ ✖

ame

Legend format host -> {{instance}}

Step 1s Resolution 1/2

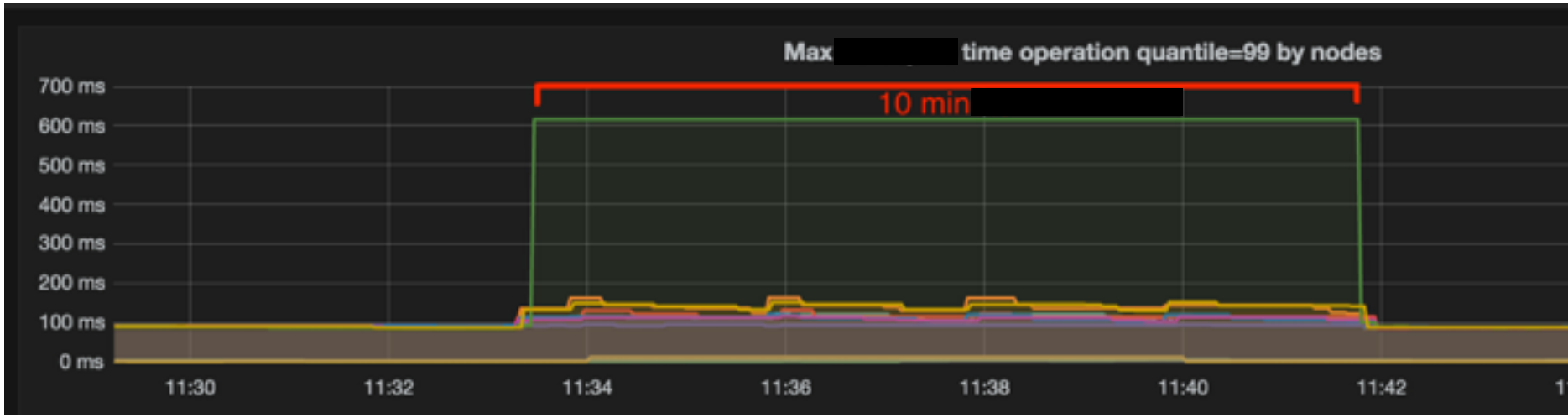
Query sum(delta(cache_aerospike_duration_milliseconds_count{isError="1",operation="s Metric [redacted] *Prometheus* ☰ ✖

# Конструктор дашбордов

Alexander I.Grafov

https://github.com/grafov/autograf

# Prometheus summary problems




Проблемы:

- Почему 10 минут значение постоянное?
- Почему прямоугольные графики?

**Квантиль** значение, которое заданная случайная величина не превышает с фиксированной вероятностью

$$Q(0.9) = \mathbf{y}$$

С вероятностью в **90%** значение будет **не больше** чем y

Нужно посчитать **0.5 квантиль**

на множестве значений:

| | | | | | |
|---|---|---|---|---|---|
| 25 | 10 | 13 | 17 | 30 | 21 |

1. **Сортируем**:

| | | | | | |
|---|---|---|---|---|---|
| 25 | 10 | 13 | 17 | 30 | 21 |

→

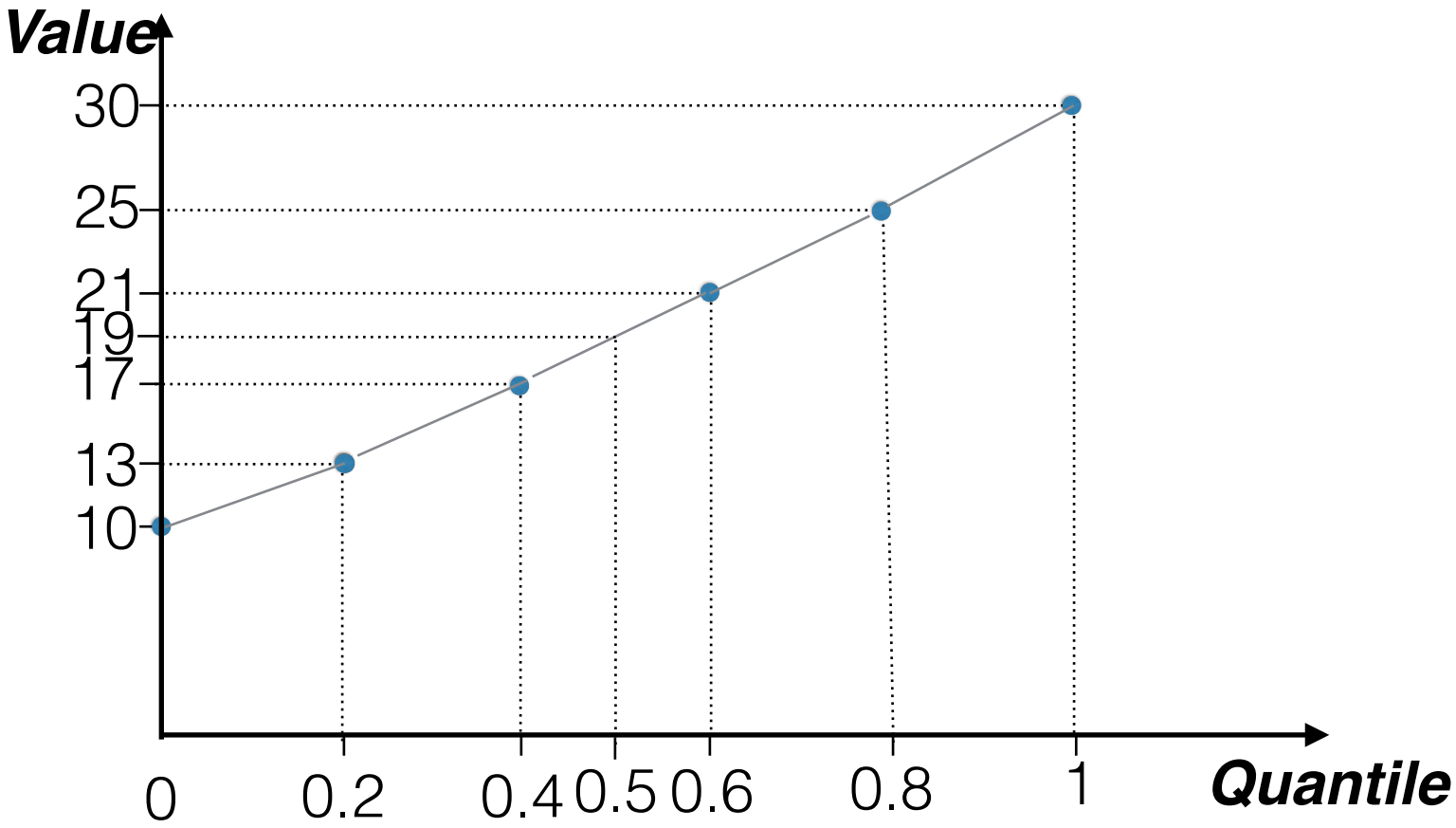
| | | | | | |
|---|---|---|---|---|---|
| 10 | 13 | 17 | 21 | 25 | 30 |

2. **Вычисляем квантили**:

$$quantile\_step = \frac{1}{len(values) - 1}$$

| **Quantile** | 0 | 0.2 | 0.4 | 0.6 | 0.8 | 1 |
|---|---|---|---|---|---|---|
| **Value** | 10 | 13 | 17 | 21 | 25 | 30 |

# Вычисление квантили



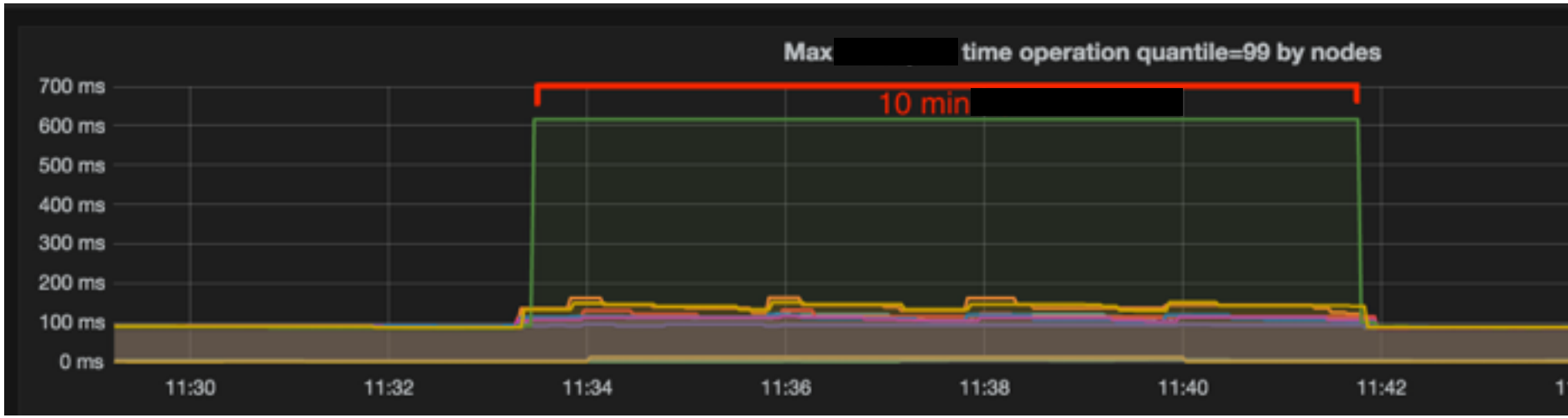
# Summary вычисление кванитили

**Active**

| bucket 1 | bucket 2 | bucket 3 | bucket 4 | bucket 5 |
|---|---|---|---|---|
| 0 | | | | |
| ~~1008000~~ | ~~1008000~~ | ~~1008000~~ | ~~1008000~~ | ~~1008000~~ |
| 2 | ~~2~~ | ~~2~~ | ~~2~~ | ~~2~~ |
| 3 | 3 | ~~3~~ | ~~3~~ | ~~3~~ |
| 4 | 4 | 4 | ~~4~~ | ~~4~~ |
| 5 | 5 | 5 | 5 | ~~5~~ |
| 6 | 6 | 6 | 6 | 6 |

10 minutes

# Prometheus summary problems




Проблемы:

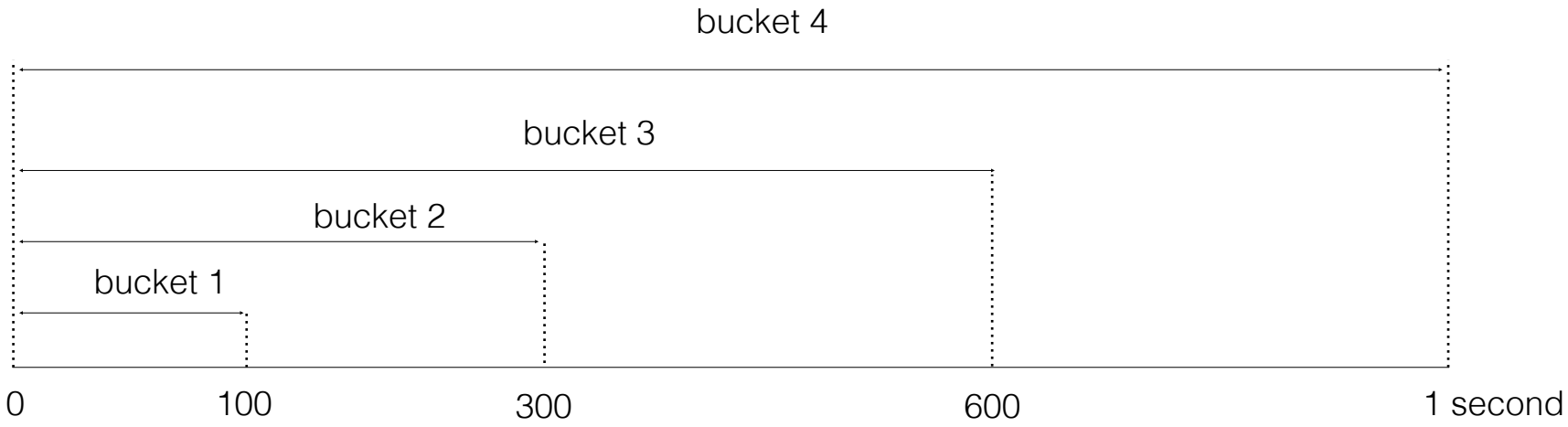
- Почему 10 минут значение постоянное?

  Так настроены по дефолту интервалы обновления данных

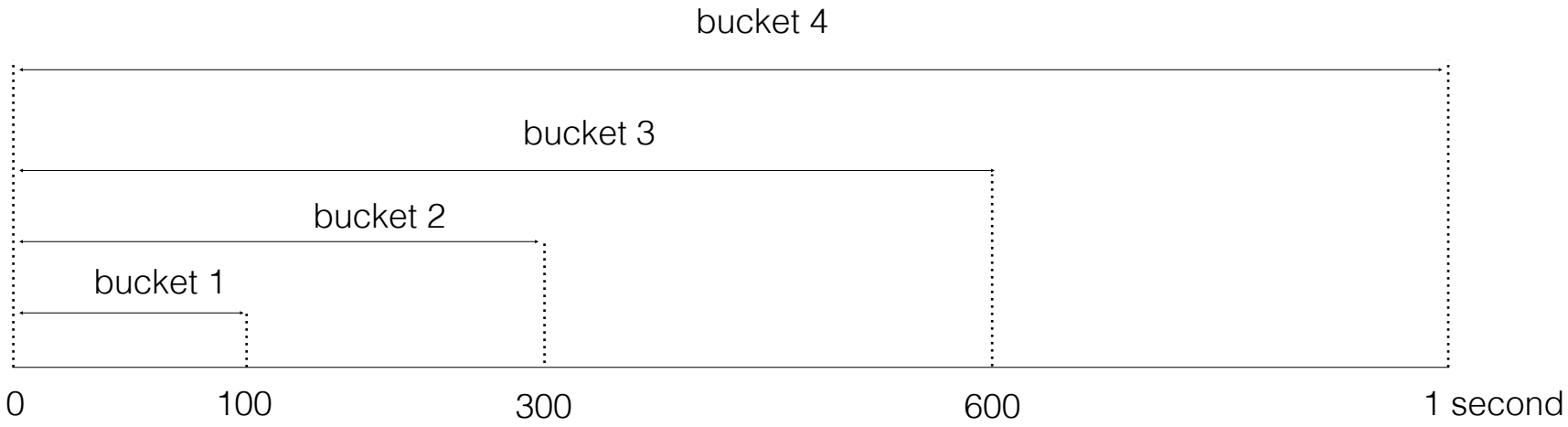
- Почему прямоугольные графики?

  Не используется линейная интерполяция

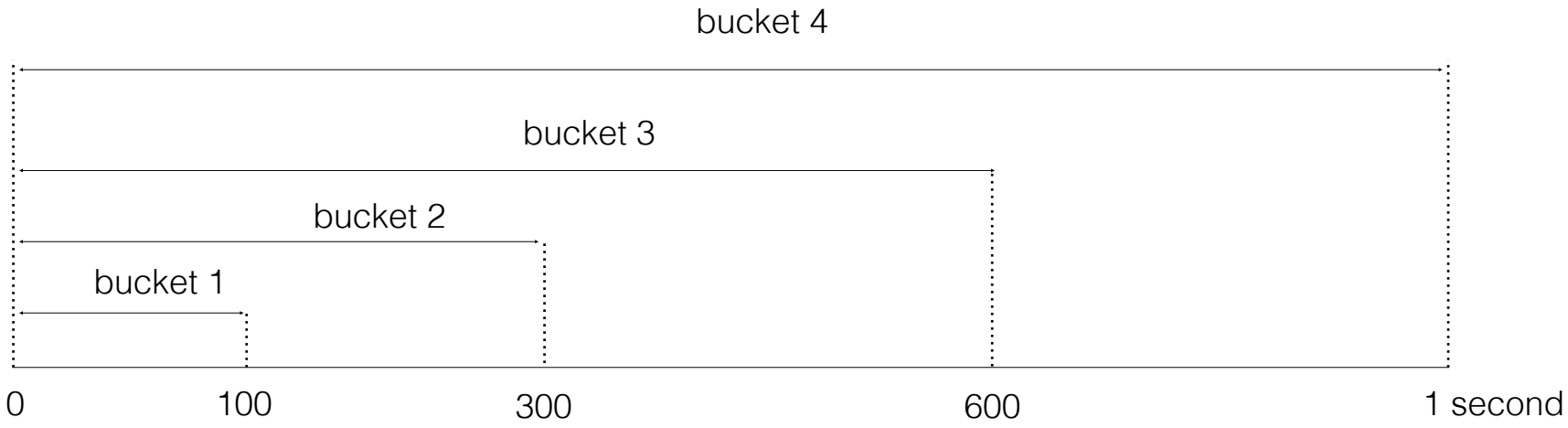
# Histogram Prometheus



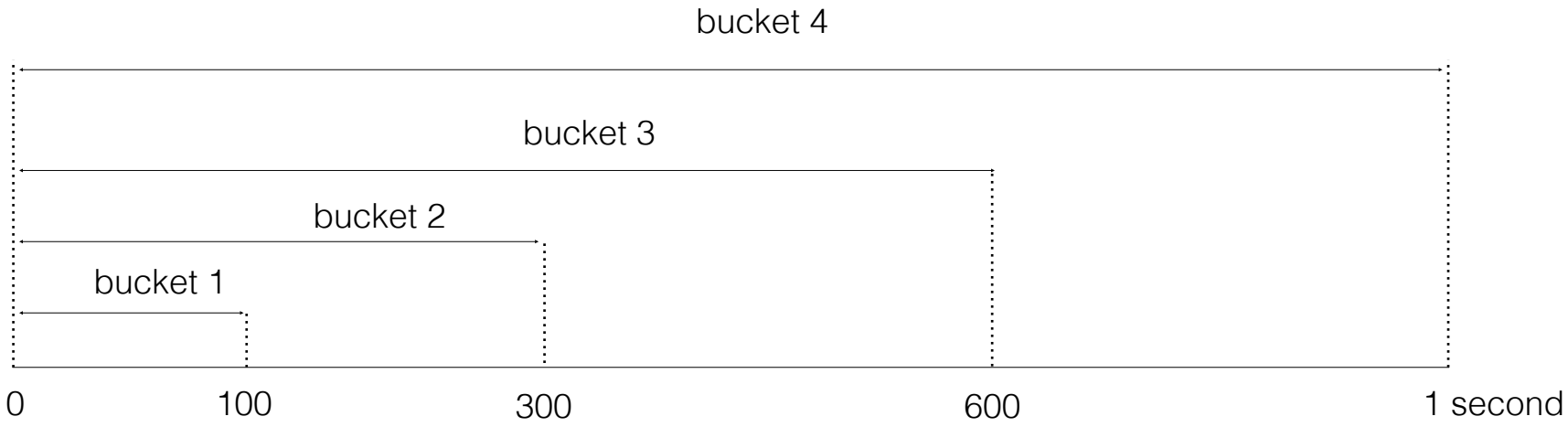

| **Bucket** | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| **Count** | 10 | 20 | 30 | 80 |



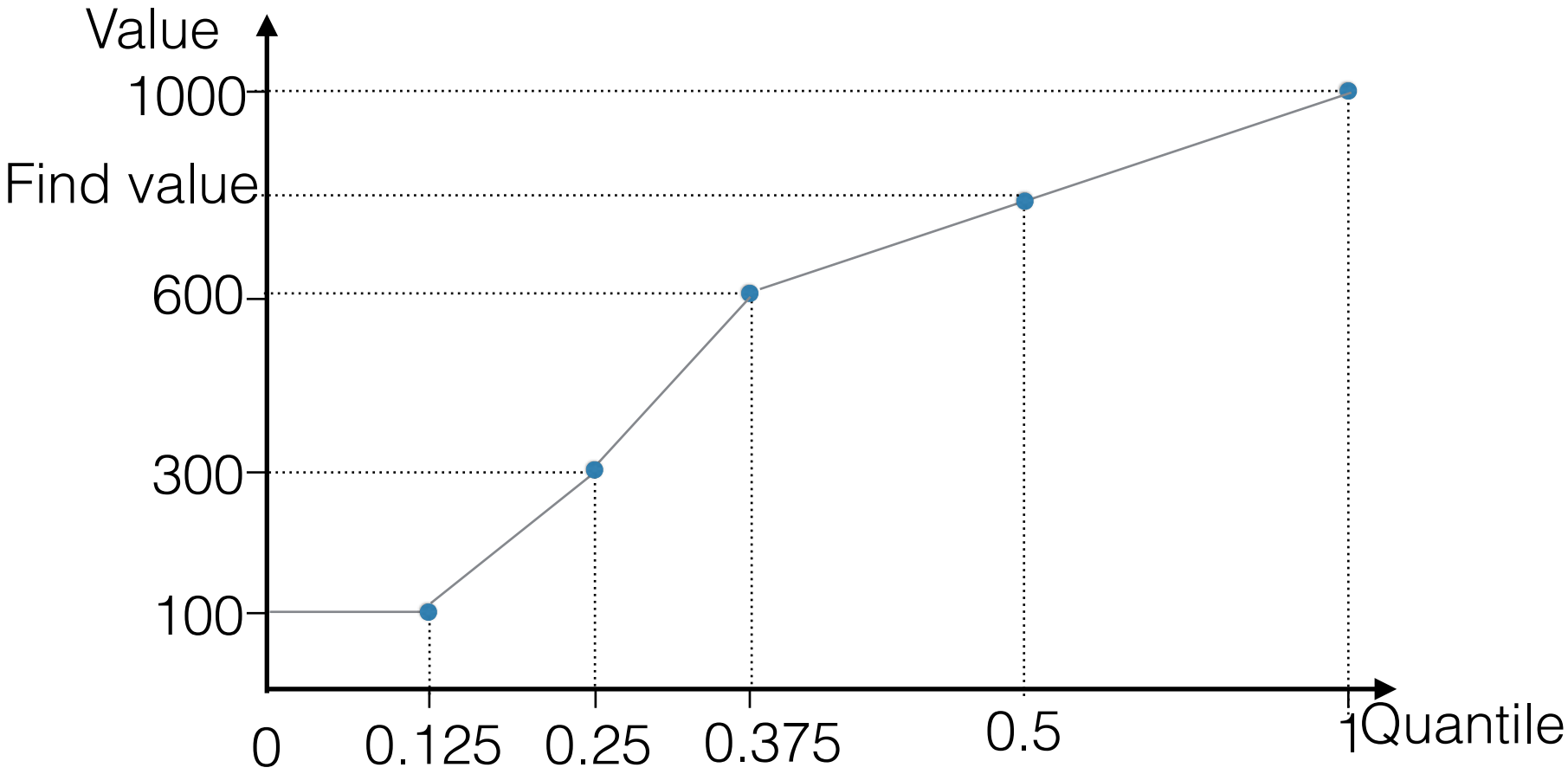

| **Bucket** | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| **Probability** | 10/80 | 20/80 | 30/80 | 80/80 |




| **Bucket** | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| **Probability** | 0,125 | 0,25 | 0,375 | 1 |




| **Bucket** | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| **Quantile** | 0,125 | 0,25 | 0,375 | 1 |



# Histogram Prometheus

***Плюсы***

- Вычисляет любую квантиль за любой интервал времени
- Уходит нагрузка с клиента
- Вычисляет квантиль для агрегируемых данных с разных node

***Минусы***

- Нужно заранее задавать бакеты
- При неправильно подобранных бакетах идет большая погрешность вычисления квантиль

# Вы узнали:

- Преимущества Prometheus(TimeSeries,Label,Pull Scheme)
- Как в Prometheus устроено хранение лайблов и метрик
- Нашу схему работы с Prometheus
- Как можно решить проблему с дублированием данных в Grafana
- Как вычисляются квантили на стороне клиента и сервера

# Prometheus

- Стали загружать в 10-ки раз больше метрик
- Расширились возможности агрегации данных
- Открытый код написанный на GO
- Бесплатное решение
- Независимость

LAZADA TECH HUB

# PROMETHEUS TEAM

LAZADA TECH HUB

# Вопросы?

# Спасибо)


**Evgeny Pak, Senior Go Developer**
**Компания Lazada**

justgoodman@yandex.ru
skype: justgoodman84


# Спасибо :)

**Evgeny Pak, Senior Go Developer**
**Компания Lazada**

justgoodman@yandex.ru
skype: justgoodman84