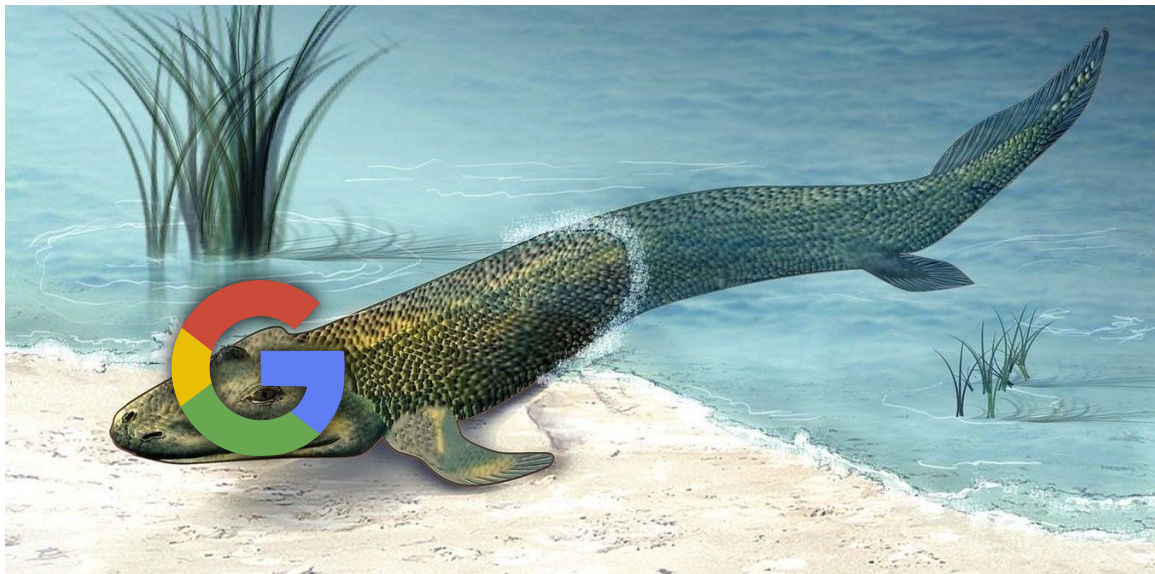


Monitoring in a Serverless World

Colin Douch
Cloudflare Observability

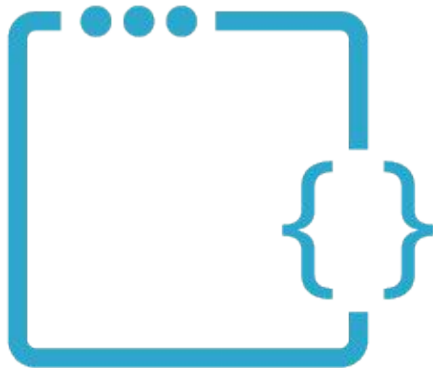
In the beginning, life was good



These SaaS providers decided to invent Serverless Computing one day, and now we have to support the Observability infrastructure

Serverless At Cloudflare

We made a thing



And then we started
using it...

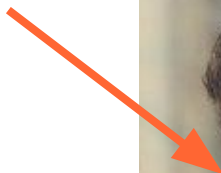
A Quick Introduction

Hi! I'm Colin

Observability Lead @ Cloudflare

Incorrectly thinks that COBOL is
a decent language

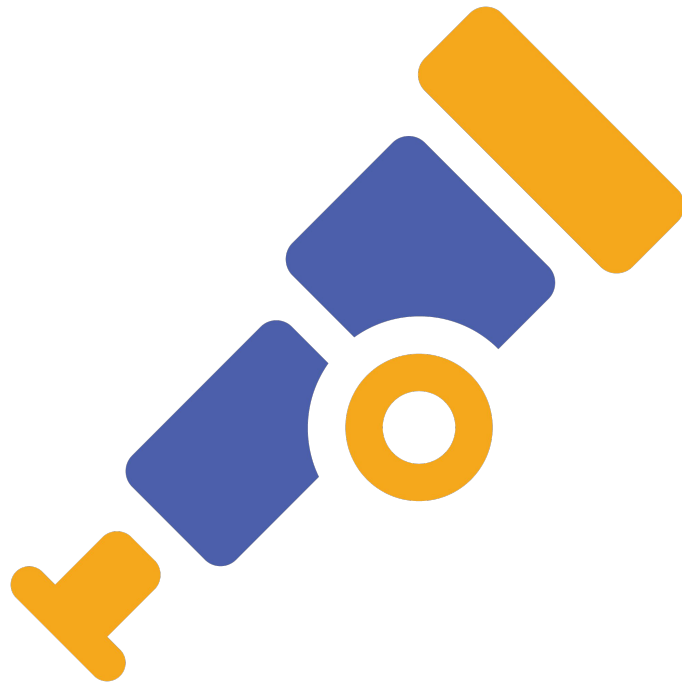
Not British



Metrics

Logs

Traces



Metrics!

Prometheus:
Innovative,
Useful



The downside of Prometheus

Prometheus:
Innovative,
Useful,
Opinionated



Your system can be discovered and scraped



You have to **wait**
for Prometheus to
talk to you

Your system can expose metrics over the network

Prometheus needs to be able to **pull** your metrics

- Your metrics need to be exposed over HTTP (generally...)
- And it needs to be reachable over the network



Your system can do its own aggregation



Credit: Wikipedia

What if your
service only ever
processes one
request?

TYPE slide SUMMARY

We did a survey!

- What sort of metrics do our teams actually use?
- Do teams actually *use* aggregation?
- What sort of semantics do they need?



What We Found: 😱



What we found: Counters



Numbers that sum together

- Request counts
- Error counts

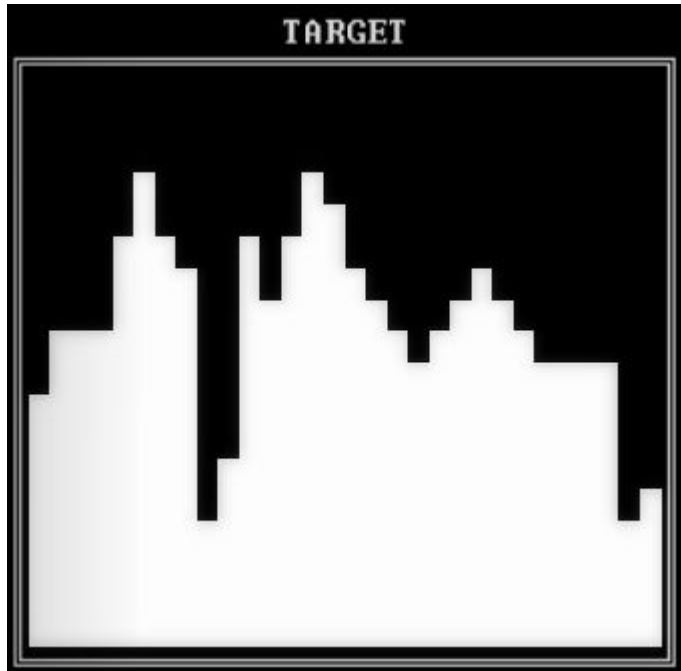
What we found: Gauges

Numbers that go
up and down

- Memory usage
- Request Times



What we found: Histograms / Summaries



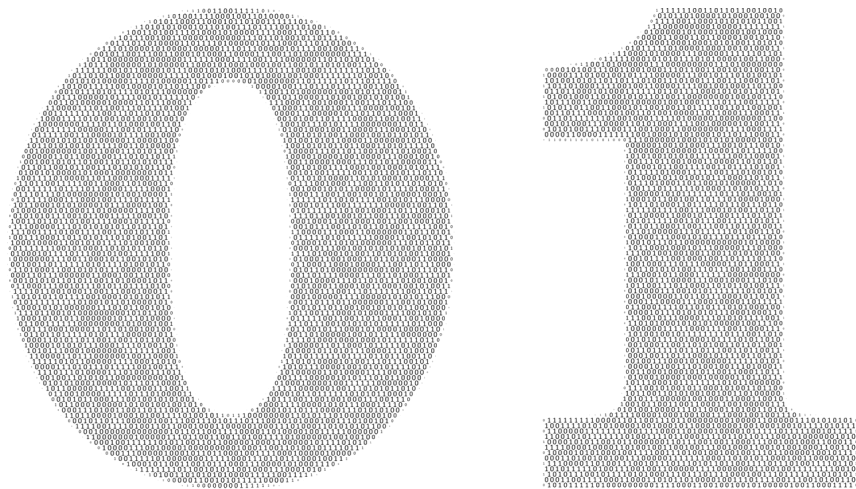
Deceptively hard.

- What happens when you need to merge buckets?
- What happens when you get a *summary*

What we found: Info metrics

Metrics that replace the entire *metric family* when a new label set comes in

- Version metrics
- Enums



Some prior art

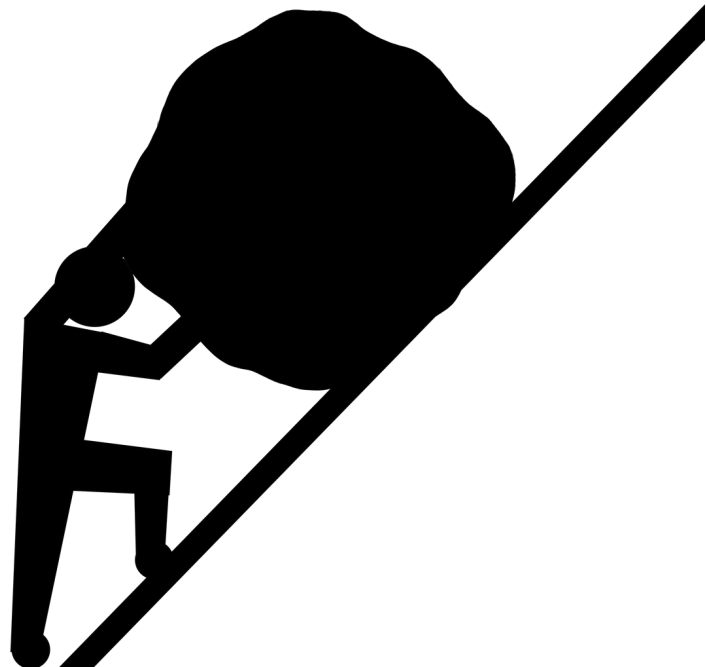


Push Gateway

<https://github.com/prometheus/pushgateway>



Usually, the only valid use case for the Pushgateway is for capturing the outcome of a service-level batch job.



Aggregation Gateway

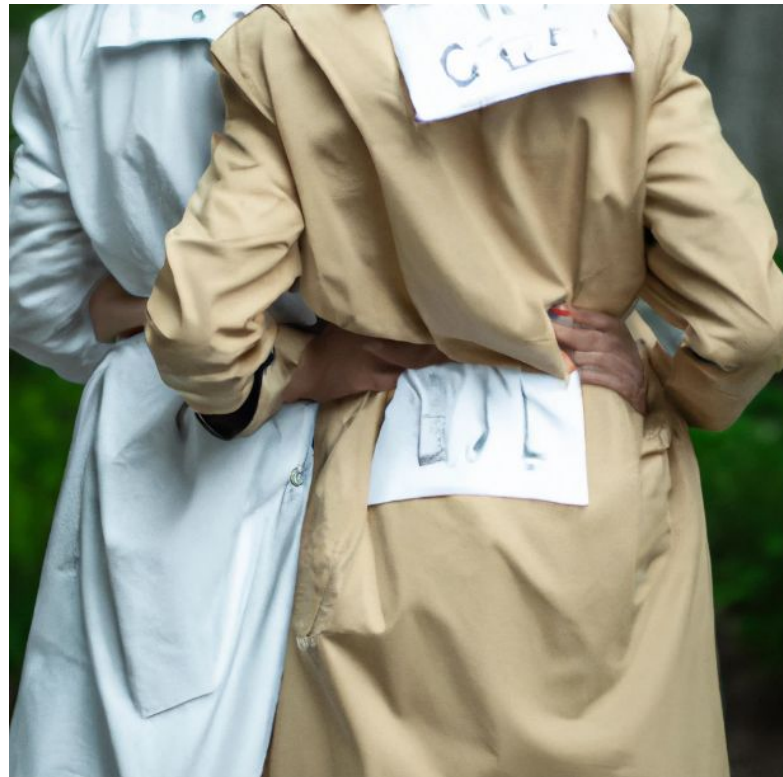


<https://github.com/weaveworks/prom-aggregation-gateway>



An Internal Bespoke thing

Let's just stitch the
two things together



Something Event Based?

We could **leave Text Exposition**
behind *entirely*

Events2prom, Cleodora

What we came up with

The Gravel Gateway



Credit: Wikipedia

<https://github.com/sinkingpoint/gravel-gateway>



Some functional requirements

1. Accept **pushes**, like the Push Gateway
2. **Aggregate** like the Aggregation Gateway
3. Aggregate **with all our identified semantics**

**“How do we
express all the
semantics we
need, in the limited
amount of space
we have”?**

```
# TYPE go_threads gauge  
go_threads 13
```

```
# TYPE build_info  
build_info{go_version="1.18",version="2022.2.1"} 1
```

That's easy: We Cheat

Let's smuggle semantic information
in the labels!



Introducing the “Clear Mode Label”

```
wshim_build_info{go_version="1.18",clear_mode="info"}
```

Summing

```
wshim_request_count{clear_mode="sum"} 1  
wshim_request_count{clear_mode="sum"} 1
```

```
wshim_request_count 2
```

Info Metrics

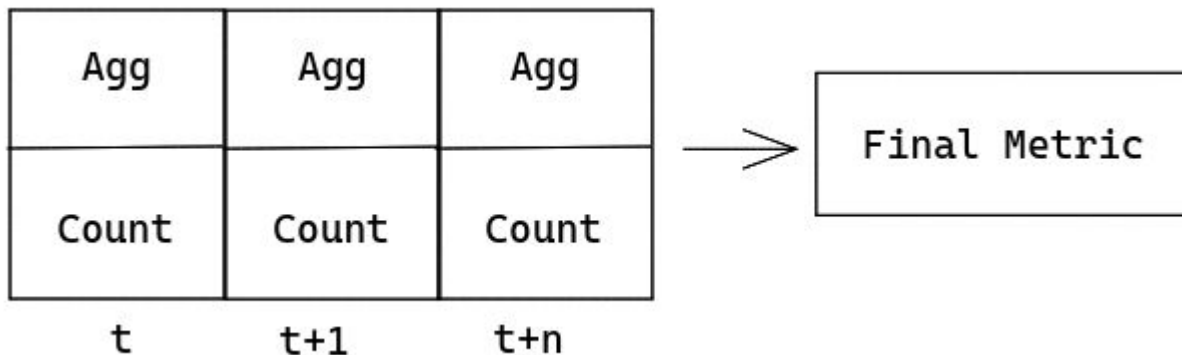
```
wshim_info{version="1.0",go_version="1.17,clear_mode="family"} 1  
wshim_info{version="1.1",go_version="1.18",clear_mode="family"} 1
```

```
wshim_info{version="1.1",go_version="1.18"} 1
```

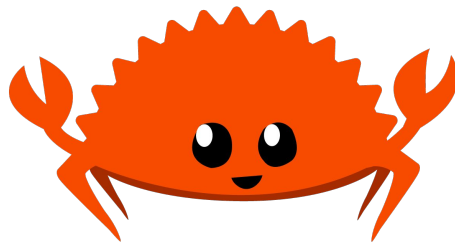
Means/Medians

```
wshim_memory_usage_bytes{clear_mode="mean5m"} 0  
wshim_memory_usage_bytes{clear_mode="mean5m"} 5  
wshim_memory_usage_bytes{clear_mode="mean5m"} 10
```

wshim_memory_usage_bytes 5



Compatibility with existing client libraries



How this looks in practice

```
const { Pushgateway, register } = require('prom-client')
const pushGw = new Pushgateway(
  'https://gravel.sinkingpoint.com',
  register
);

const counter = new client.Counter({name: 'requests'});
counter.inc();

const info = new client.Gauge({name: 'info'});
info.set(1, {'clear_mode': 'family', 'version': '1.0'});

pushGw.push({ jobName: 'test' }, (err, resp, body) => {});
```

Looking Forward to the Future



Where Gravel Gateway goes from here

- More work on scaling
- More interesting aggregations
- Rewriting pebbles



Credit: Wikipedia

Open Metrics



OPEN METRICS

Cleodora and the shift towards Open Telemetry



Thanks!

Slides: <https://colindou.ch/slides>

Code: <https://github.com/sinkingpoint/gravel-gateway>

Want to chat, or tell me how bad of a person I am?

@colind@hachyderm.io

iam@colindou.ch

In the hallway, after this



Slides ^

Code v

