Prometheus 3.0

Bryan Boreham, Grafana

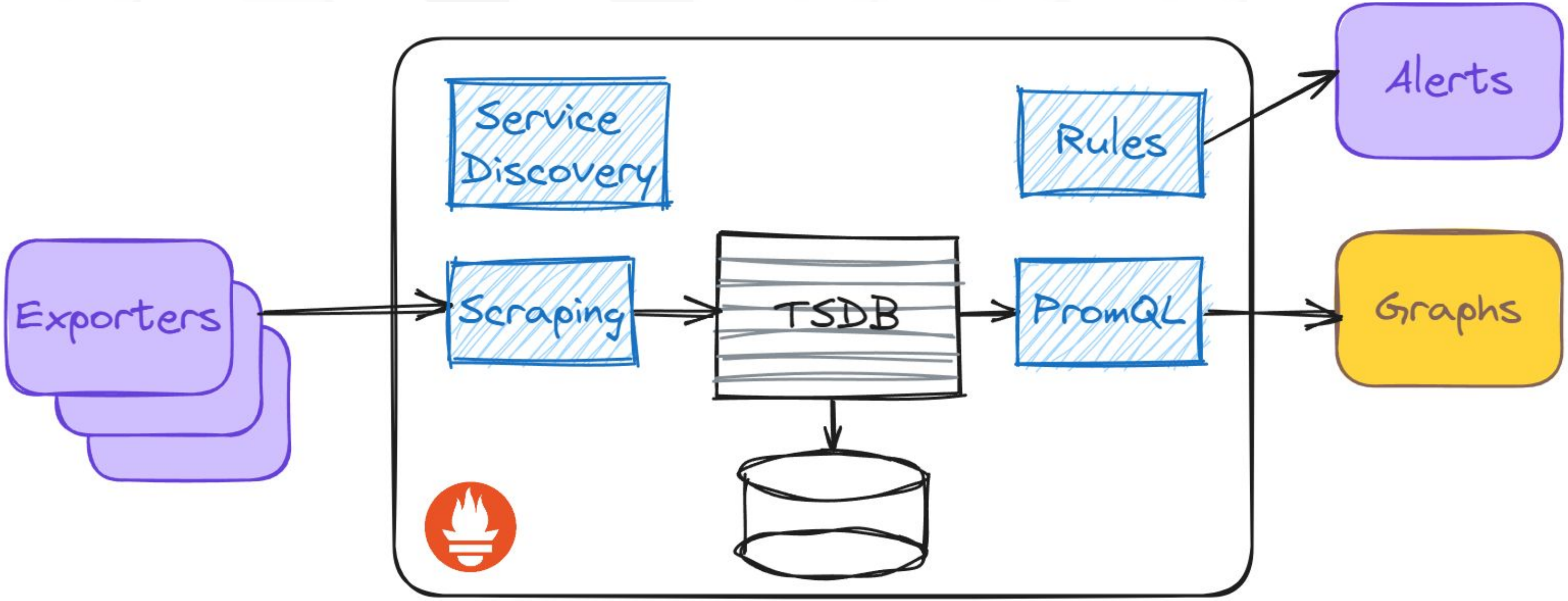
Jan Fajerski, Red Hat

FOSDEM 25

Show of hands

Intro

https://youtu.be/rT4fJNbfe14

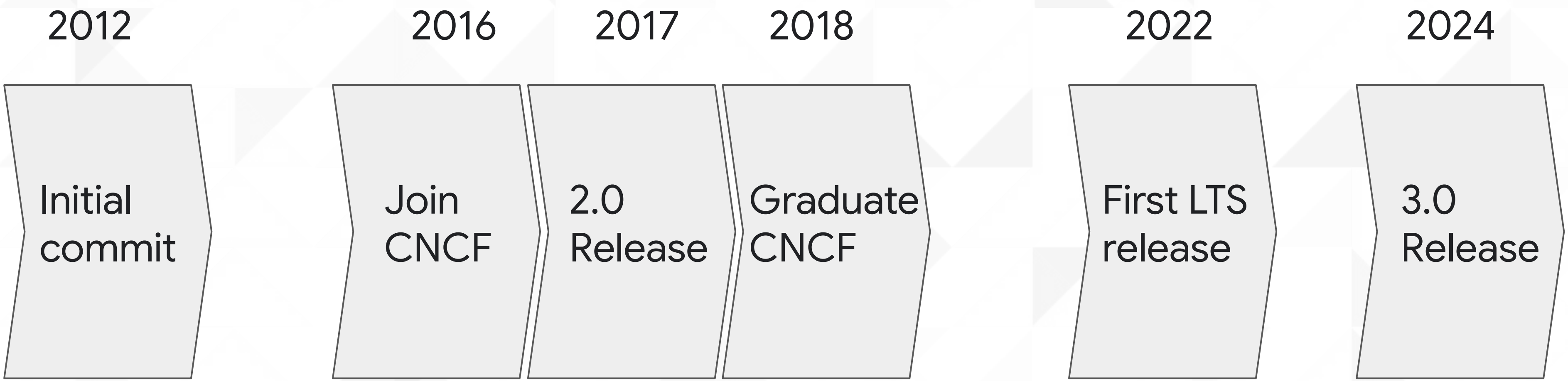


History

Prometheus started at Soundcloud

Second project to join and graduate CNCF

Active exporter ecosystem, widely used



2012 Initial commit

2016 Join CNCF

2017 2.0 Release

2018 Graduate CNCF

2022 First LTS release

2024 3.0 Release

What is Prometheus 3.0 ?

First Major release in 7 years

Brand new UI

OTEL Compatibility

UTF-8 support

Remote Write 2.0

Native Histogram support

Breaking changes

Checkout the Migration guide

Previous talks

https://www.youtube.com/results?search_query=Prometheus+3.0

Changes

Regex change

What does this selector do? **`my_series{id=~".*"}`**

A: No work, just fetches all **`my_series`**

B: Scans every possible value of the **`id`** label

Answer: B in Prometheus 1.x and 2.x, A in Prometheus 3.x

Once you've made one breaking change...

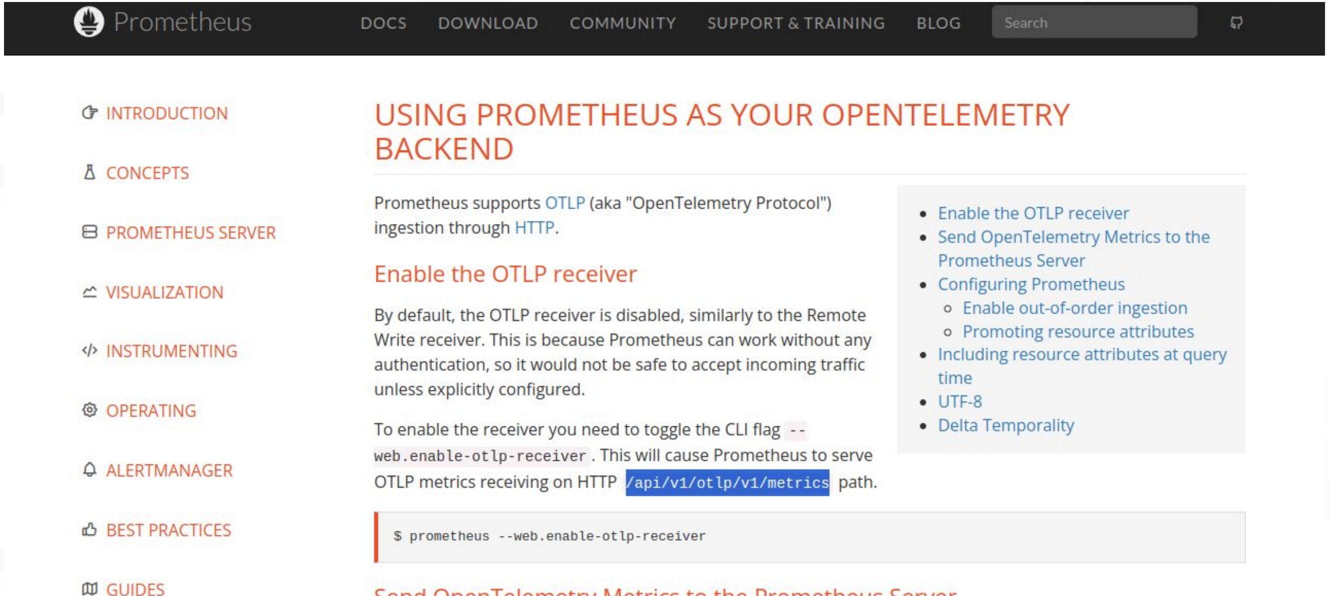
- [CHANGE] Set the `GOMAXPROCS` variable automatically to match the Linux CPU quota. Use `--no-auto-gomaxprocs` to disable it. The `auto-gomaxprocs` feature flag was removed. #15376
- [CHANGE] Set the `GOMEMLIMIT` variable automatically to match the Linux container memory limit. Use `--no-auto-gomemlimit` to disable it. The `auto-gomemlimit` feature flag was removed. #15373
- [CHANGE] Scraping: Remove implicit fallback to the Prometheus text format in case of invalid/missing Content-Type and fail the scrape instead. Add ability to specify a `fallback_scrape_protocol` in the scrape config. #15136
- [CHANGE] Remote-write: default enable_http2 to false. #15219
- [CHANGE] Scraping: normalize "le" and "quantile" label values upon ingestion. #15164
- [CHANGE] Scraping: config `scrape_classic_histograms` was renamed to `always_scrape_classic_histograms`. #15178
- [CHANGE] Config: remove expand-external-labels flag, expand external labels env vars by default. #14657
- [CHANGE] Disallow configuring AM with the v1 api. #13883
- [CHANGE] regexp `.` now matches all characters (performance improvement). #14505
- [CHANGE] `holt_winters` is now called `double_exponential_smoothing` and moves behind the experimental-promql-functions feature flag. #14930
- [CHANGE] API: The OTLP receiver endpoint can now be enabled using `--web.enable-otlp-receiver` instead of `--enable-feature=otlp-write-receiver`. #14894
- [CHANGE] Prometheus will not add or remove port numbers from the target address. `no-default-scrape-port` feature flag removed. #14160
- [CHANGE] Logging: the format of log lines has changed a little, along with the adoption of Go's Structured Logging package. #14906
- [CHANGE] Don't create extra `_created` timeseries if feature-flag `created-timestamp-zero-ingestion` is enabled. #14738
- [CHANGE] Float literals and time durations being the same is now a stable fetaure. #15111
- [CHANGE] UI: The old web UI has been replaced by a completely new one that is less cluttered and adds a few new features (PromLens-style tree view, better metrics explorer, "Explain" tab). However, it is still missing some features of the old UI (notably, exemplar display and heatmaps). To switch back to the old UI, you can use the feature flag `--enable-feature=old-ui` for the time being. #14872
- [CHANGE] PromQL: Range selectors and the lookback delta are now left-open, i.e. a sample coinciding with the lower time limit is excluded rather than included. #13904
- [CHANGE] Kubernetes SD: Remove support for `discovery.k8s.io/v1beta1` API version of EndpointSlice. This version is no longer served as of Kubernetes v1.25. #14365
- [CHANGE] Kubernetes SD: Remove support for `networking.k8s.io/v1beta1` API version of Ingress. This version is no longer served as of Kubernetes v1.22. #14365
- [CHANGE] UTF-8: Enable UTF-8 support by default. Prometheus now allows all UTF-8 characters in metric and label names. The corresponding `utf8-name` feature flag has been removed. #14705
- [CHANGE] Console: Remove example files for the console feature. Users can continue using the console feature by supplying their own JavaScript and templates. #14807
- [CHANGE] SD: Enable the new service discovery manager by default. This SD manager does not restart unchanged discoveries upon reloading. This makes reloads faster and reduces pressure on service discoveries' sources. The corresponding `new-service-discovery-manager` feature flag has been removed. #14770
- [CHANGE] Agent mode has been promoted to stable. The feature flag `agent` has been removed. To run Prometheus in Agent mode, use the new `--agent` cmdline arg instead. #14747
- [CHANGE] Remove deprecated `remote-write-receiver`,`promql-at-modifier`, and `promql-negative-offset` feature flags. #13456, #14526
- [CHANGE] Remove deprecated `storage.tsdb.allow-overlapping-blocks`, `alertmanager.timeout`, and `storage.tsdb.retention` flags. #14640, #14643

OTEL support

`/api/v1/otlp/v1/metrics` Endpoint, disabled by default

Accepts POST request via `OTLP/HTTP` protocol

Requires additional configuration to work well



USING PROMETHEUS AS YOUR OPENTELEMETRY BACKEND

Prometheus supports OTLP (aka "OpenTelemetry Protocol") ingestion through HTTP.

- Enable the OTLP receiver
- Send OpenTelemetry Metrics to the Prometheus Server
- Configuring Prometheus
  - Enable out-of-order ingestion
  - Promoting resource attributes
- Including resource attributes at query time
- UTF-8
- Delta Temporality

Enable the OTLP receiver

By default, the OTLP receiver is disabled, similarly to the Remote Write receiver. This is because Prometheus can work without any authentication, so it would not be safe to accept incoming traffic unless explicitly configured.

To enable the receiver you need to toggle the CLI flag `--web.enable-otlp-receiver`. This will cause Prometheus to serve OTLP metrics receiving on HTTP `/api/v1/otlp/v1/metrics` path.

```
$ prometheus --web.enable-otlp-receiver
```

Send OpenTelemetry Metrics to the Prometheus Server

Support all characters ("UTF-8")

This change allows Prometheus to accept OpenTelemetry Semantic Conventions.

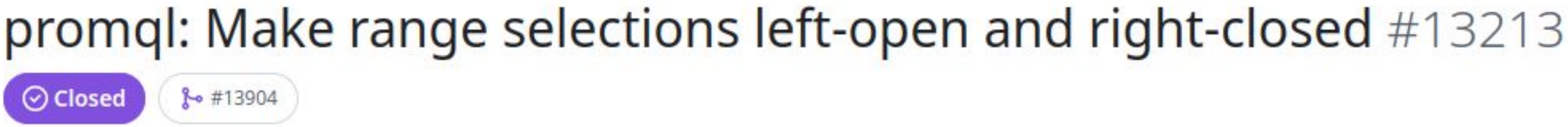
Example: **`service.name`**

Characters outside of traditional Prometheus label names need quoting.

```
    {"service.name"="nginx"}
```

https://prometheus.io/docs/guides/utf8/

Range selector [t1, t2] to (t1, t2]



promql: Make range selections left-open and right-closed #13213

Closed

#13904

> “This is, however, a breaking change, although the **impact is mostly academic**. Therefore, we should implement this change with the upcoming 3.0.0 release.”

https://github.com/prometheus/prometheus/issues/13213

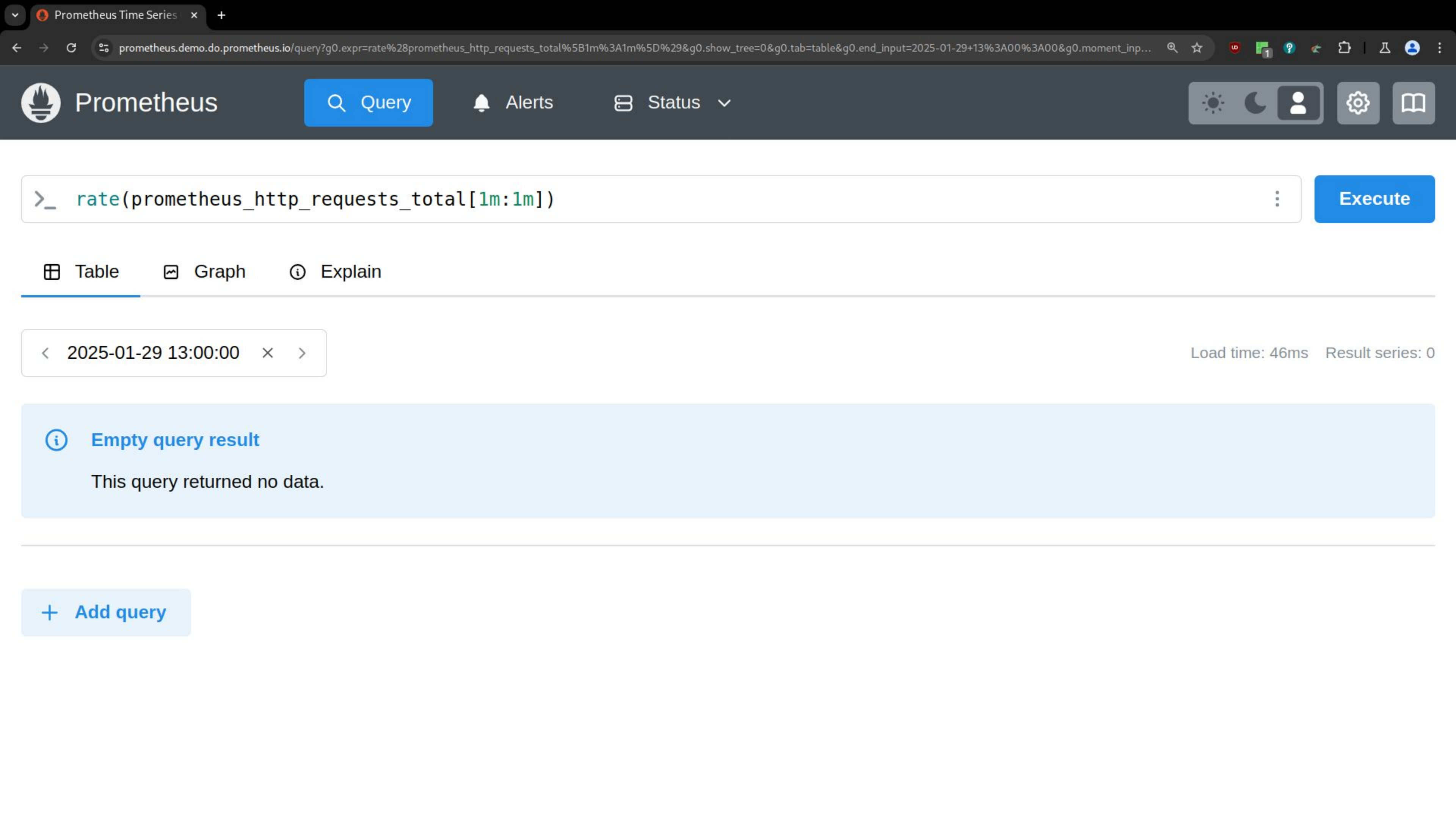
Range selector [t1, t2] to (t1, t2]

“This change has likely few effects for everyday use, except for **some** subquery use cases.
Query front-ends that **align queries** usually align subqueries to multiples of the step size. These subqueries will likely be affected.
Tests are more likely to affected. To fix those either adjust the expected number of samples or extend the range by less than one sample interval.”

```
rate(prometheus_http_requests_total[1m:1m])
```

https://prometheus.io/docs/prometheus/latest/migration/#promql

New UI



prometheus.demo.do.prometheus.io/query?g0.expr=rate%28prometheus_http_requests_total%5B1m%3A1m%5D%29&g0.show_tree=0&g0.tab=table&g0.end_input=2025-01-29+13%3A00%3A00&g0.moment_inp...

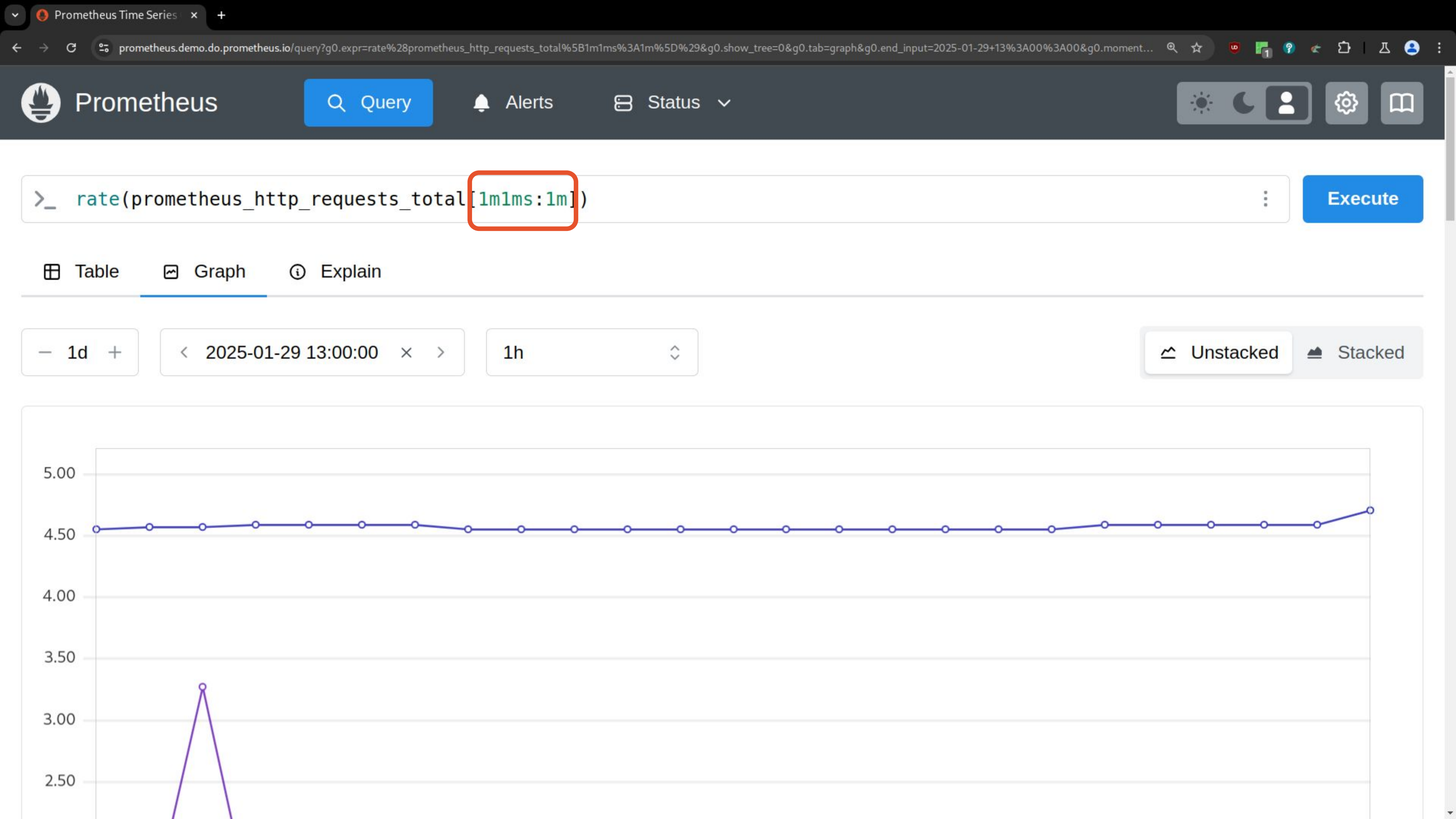
```
rate(prometheus_http_requests_total[1m:1m])
```

2025-01-29 13:00:00

Load time: 46ms Result series: 0

Empty query result

This query returned no data.



prometheus.demo.do.prometheus.io/query?g0.expr=rate%28prometheus_http_requests_total%5B1m1ms%3A1m%5D%29&g0.show_tree=0&g0.tab=graph&g0.end_input=2025-01-29+13%3A00%3A00&g0.moment...

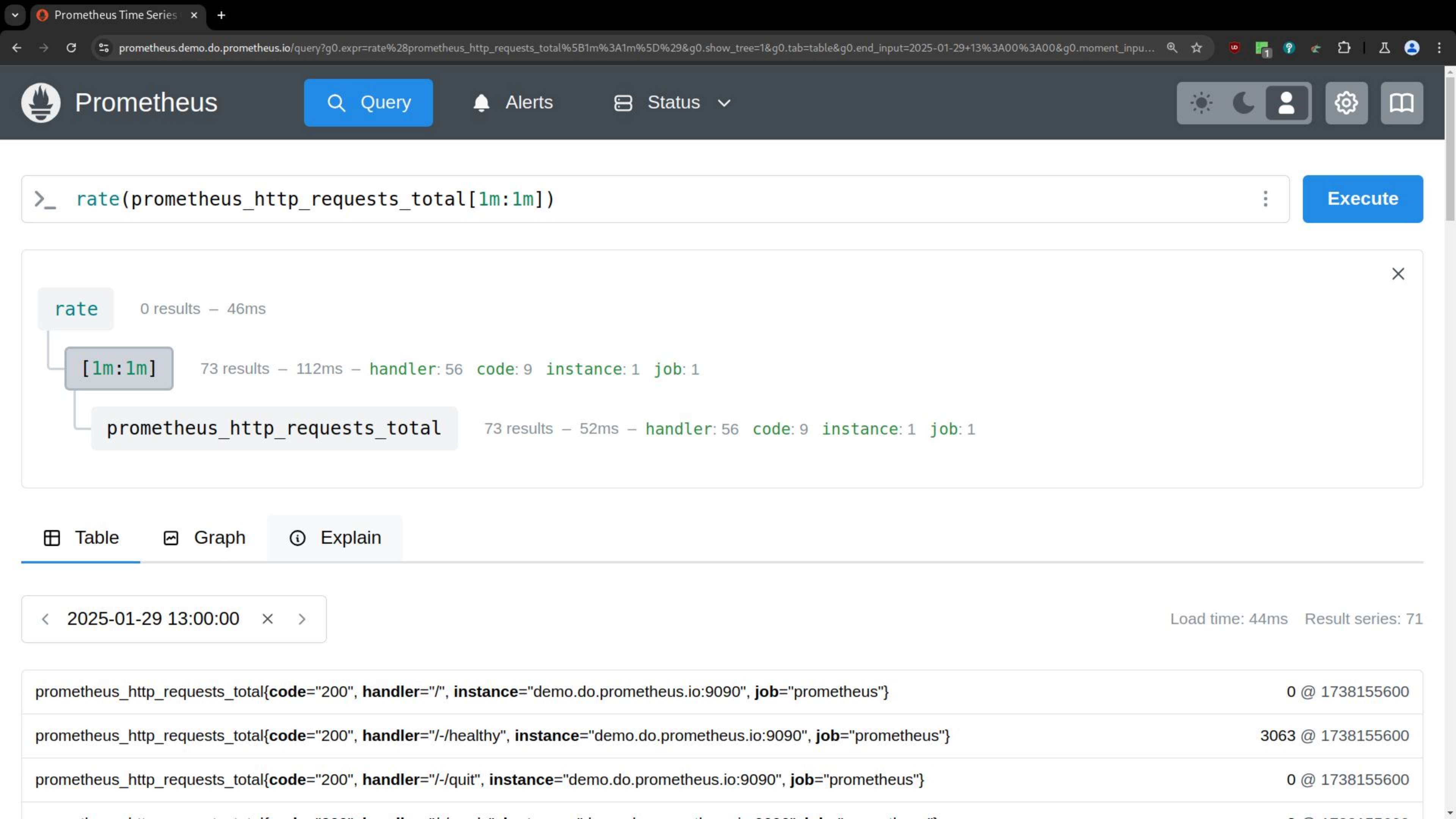
```
rate(prometheus_http_requests_total[1m1ms:1m])
```

1d

2025-01-29 13:00:00

1h

Unstacked


Prometheus Time Series
prometheus.demo.do.prometheus.io/query?g0.expr=rate%28prometheus_http_requests_total%5B1m%3A1m%5D%29&g0.show_tree=1&g0.tab=table&g0.end_input=2025-01-29+13%3A00%3A00&g0.moment_inpu...
Prometheus
Query
Alerts
Status
Execute
Table
Graph
Explain

```
rate(prometheus_http_requests_total[1m:1m])
```

rate 0 results – 46ms

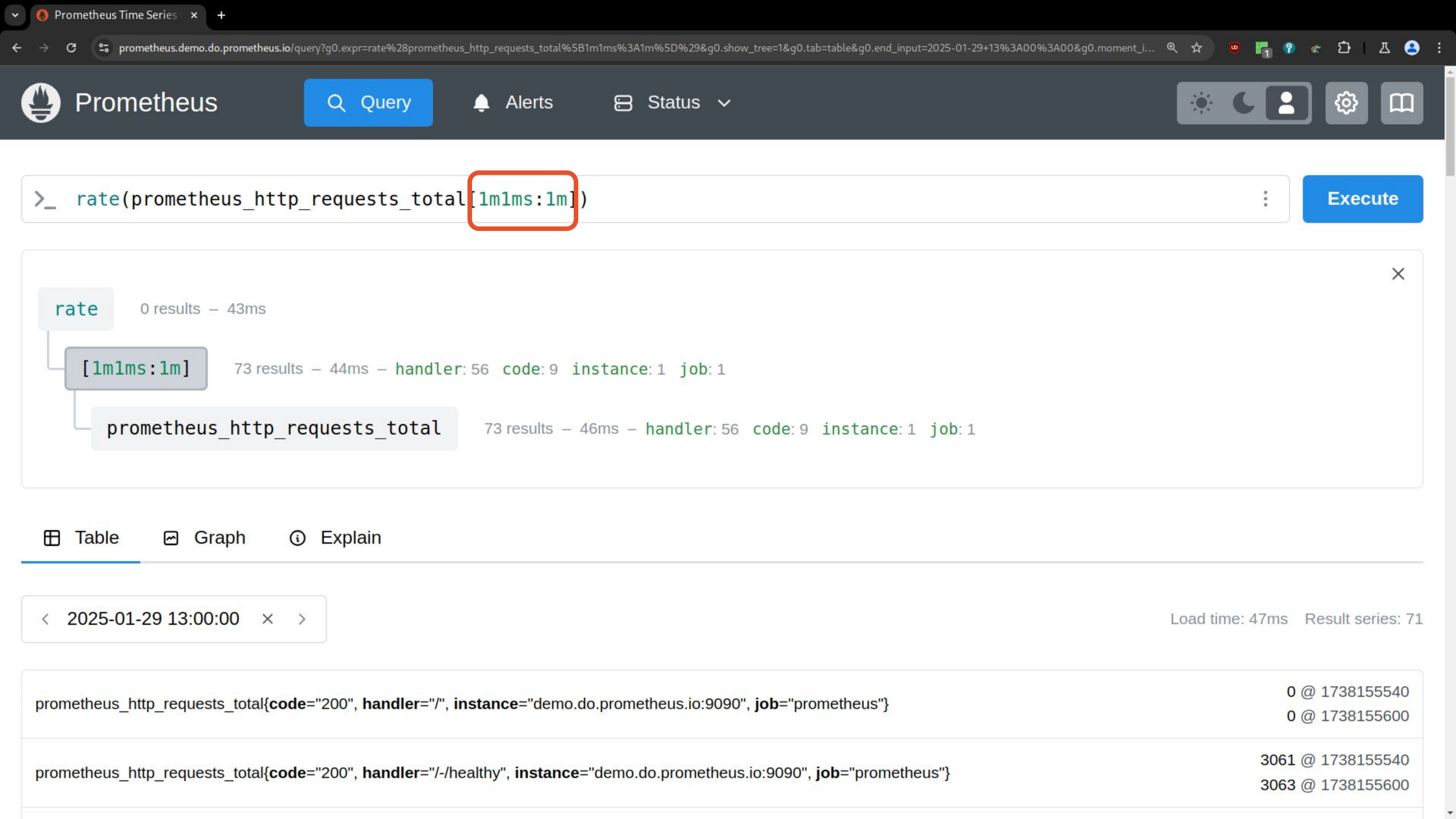
[1m:1m] 73 results – 112ms – handler: 56 code: 9 instance: 1 job: 1

prometheus_http_requests_total 73 results – 52ms – handler: 56 code: 9 instance: 1 job: 1

2025-01-29 13:00:00

Load time: 44ms Result series: 71

prometheus_http_requests_total{**code**="200", **handler**="/", **instance**="demo.do.prometheus.io:9090", **job**="prometheus"}	0 @ 1738155600
prometheus_http_requests_total{**code**="200", **handler**="/-/healthy", **instance**="demo.do.prometheus.io:9090", **job**="prometheus"}	3063 @ 1738155600
prometheus_http_requests_total{**code**="200", **handler**="/-/quit", **instance**="demo.do.prometheus.io:9090", **job**="prometheus"}	0 @ 1738155600



Prometheus Time Series
prometheus.demo.do.prometheus.io/query?g0.expr=rate%28prometheus_http_requests_total%5B1m1ms%3A1m%5D%29&g0.show_tree=1&g0.tab=table&g0.end_input=2025-01-29+13%3A00%3A00&g0.moment_i...
Prometheus
Query
Alerts
Status
Execute
Table
Graph
Explain

```
rate(prometheus_http_requests_total[1m1ms:1m])
```

rate 0 results – 43ms

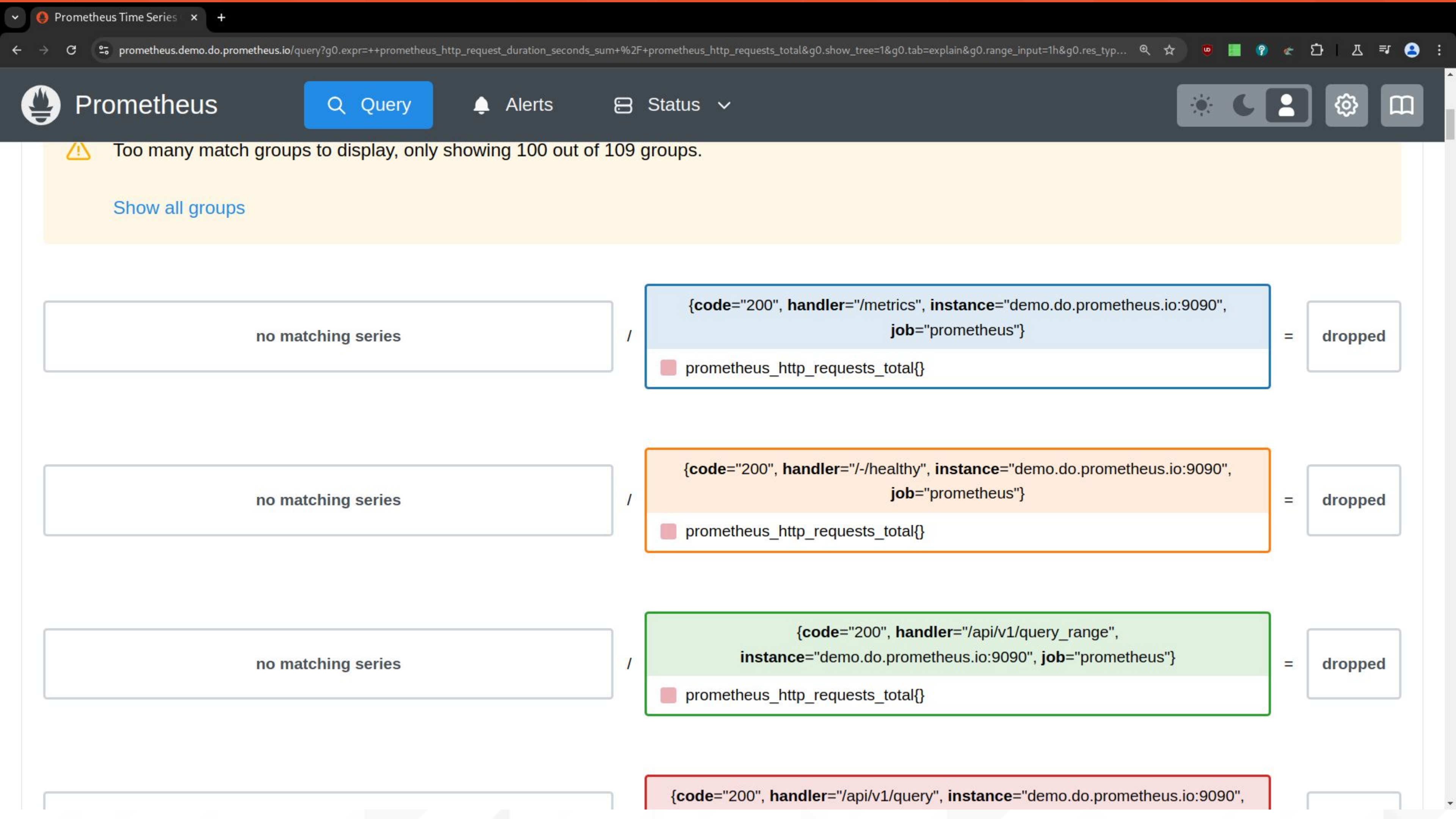
[1m1ms:1m] 73 results – 44ms – handler: 56 code: 9 instance: 1 job: 1

prometheus_http_requests_total 73 results – 46ms – handler: 56 code: 9 instance: 1 job: 1

2025-01-29 13:00:00

Load time: 47ms Result series: 71

```
prometheus_http_requests_total{code="200", handler="/", instance="demo.do.prometheus.io:9090", job="prometheus"}            0 @ 1738155540
                                                                                                                            0 @ 1738155600
prometheus_http_requests_total{code="200", handler="/-/healthy", instance="demo.do.prometheus.io:9090", job="prometheus"} 3061 @ 1738155540
                                                                                                                         3063 @ 1738155600
```


Prometheus Time Series
prometheus.demo.do.prometheus.io/query?g0.expr=++prometheus_http_request_duration_seconds_sum+%2F+prometheus_http_requests_total&g0.show_tree=1&g0.tab=explain&g0.range_input=1h&g0.res_typ…
Prometheus
Query
Alerts
Status

⚠ Too many match groups to display, only showing 100 out of 109 groups.

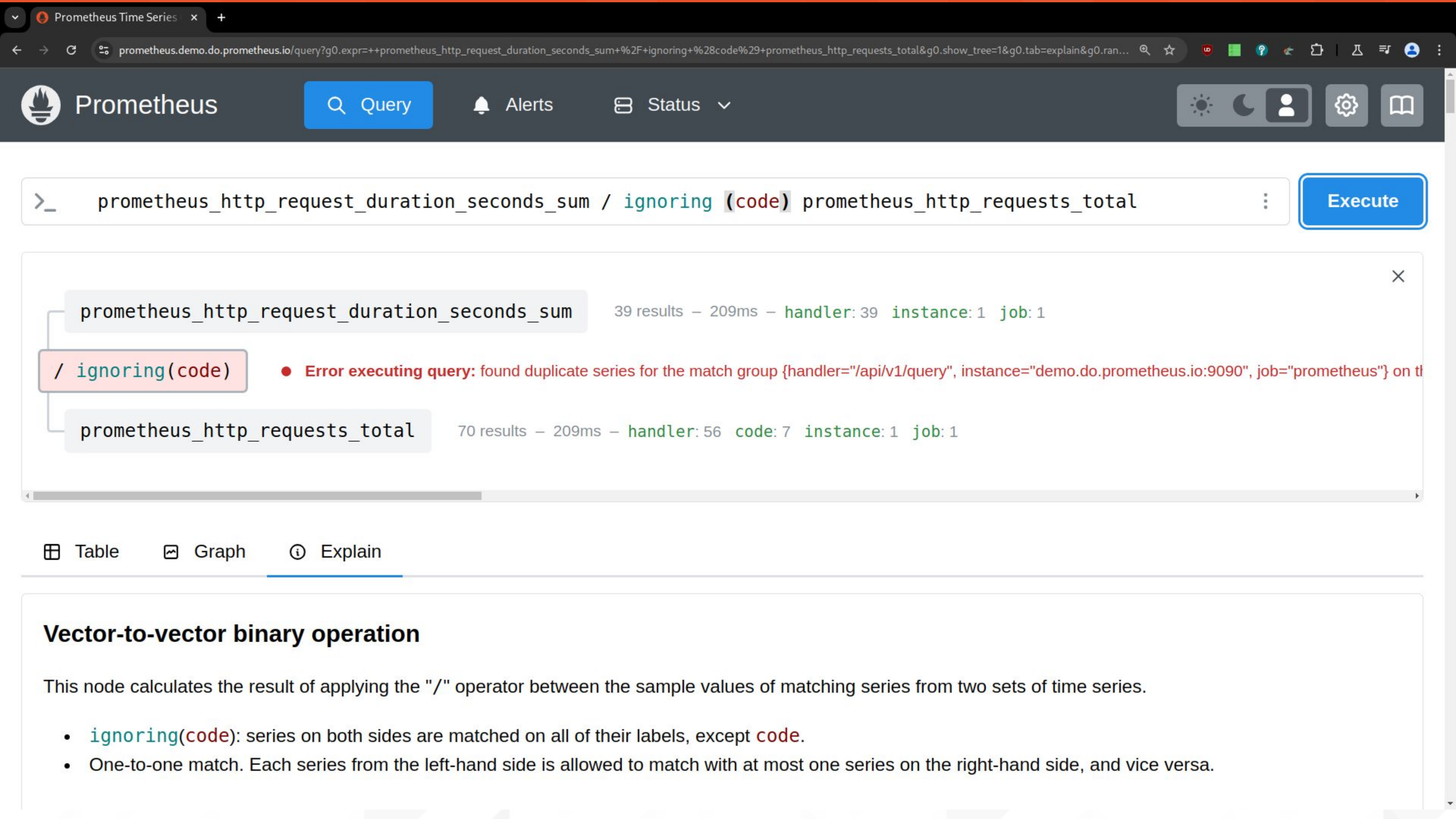
Show all groups

no matching series	/	{**code**="200", **handler**="/metrics", **instance**="demo.do.prometheus.io:9090", **job**="prometheus"}
		prometheus_http_requests_total{}
			=	dropped

no matching series	/	{**code**="200", **handler**="/-/healthy", **instance**="demo.do.prometheus.io:9090", **job**="prometheus"}
		prometheus_http_requests_total{}
			=	dropped

no matching series	/	{**code**="200", **handler**="/api/v1/query_range", **instance**="demo.do.prometheus.io:9090", **job**="prometheus"}
		prometheus_http_requests_total{}
			=	dropped

		{**code**="200", **handler**="/api/v1/query", **instance**="demo.do.prometheus.io:9090",		


Prometheus Time Series
prometheus.demo.do.prometheus.io/query?g0.expr=++prometheus_http_request_duration_seconds_sum+%2F+ignoring+%28code%29+prometheus_http_requests_total&g0.show_tree=1&g0.tab=explain&g0.ran...
Prometheus
Query
Alerts
Status
Execute
Table
Graph
Explain

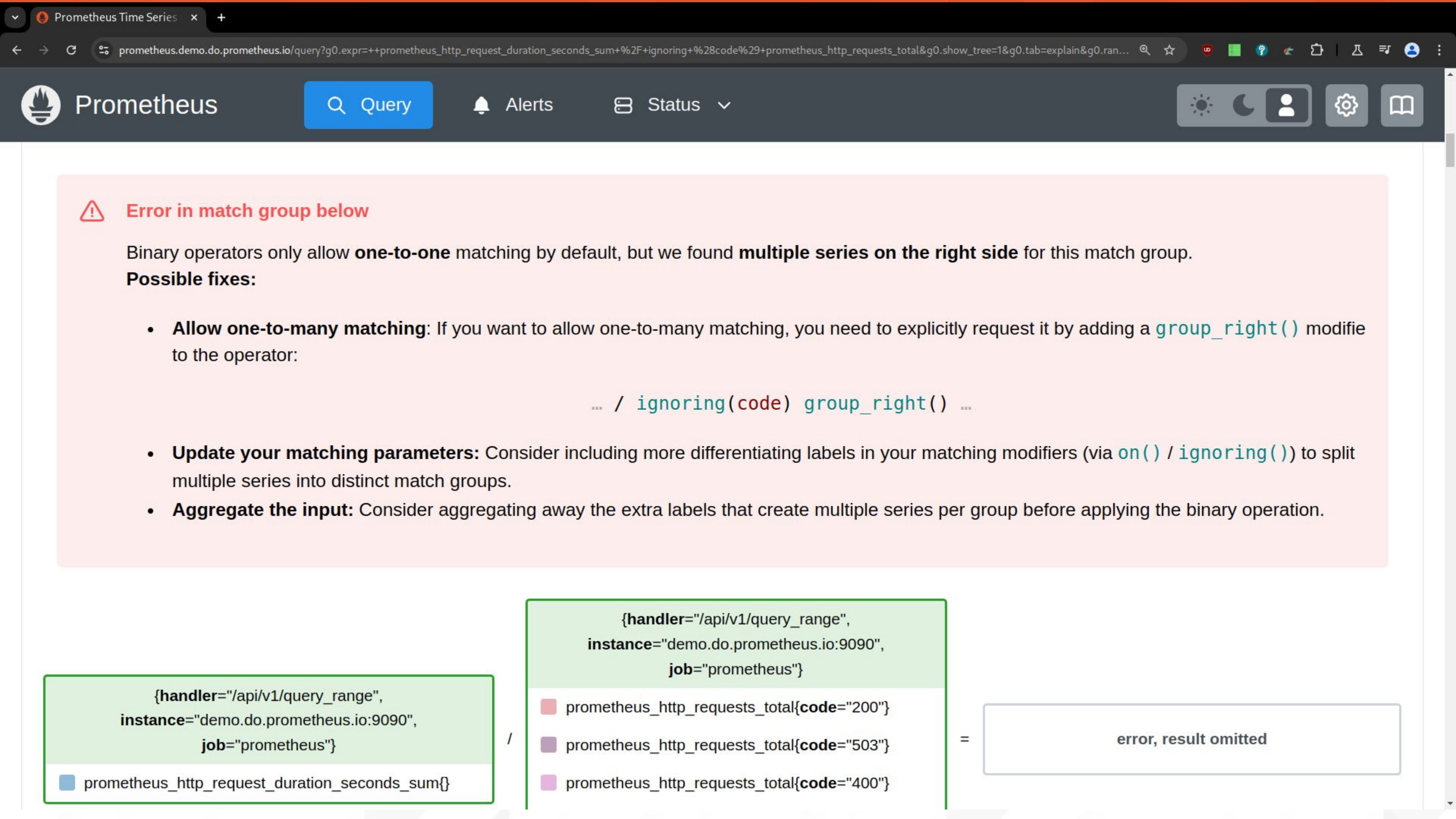
```
prometheus_http_request_duration_seconds_sum / ignoring (code) prometheus_http_requests_total
```

```
prometheus_http_request_duration_seconds_sum    39 results – 209ms – handler: 39  instance: 1  job: 1
/ ignoring(code)    ● Error executing query: found duplicate series for the match group {handler="/api/v1/query", instance="demo.do.prometheus.io:9090", job="prometheus"} on th
prometheus_http_requests_total    70 results – 209ms – handler: 56  code: 7  instance: 1  job: 1
```

Vector-to-vector binary operation

This node calculates the result of applying the "/" operator between the sample values of matching series from two sets of time series.

- ignoring(code): series on both sides are matched on all of their labels, except code.
- One-to-one match. Each series from the left-hand side is allowed to match with at most one series on the right-hand side, and vice versa.


Prometheus Time Series
prometheus.demo.do.prometheus.io/query?g0.expr=++prometheus_http_request_duration_seconds_sum+%2F+ignoring+%28code%29+prometheus_http_requests_total&g0.show_tree=1&g0.tab=explain&g0.ran...
Prometheus
Query
Alerts
Status

Error in match group below

Binary operators only allow **one-to-one** matching by default, but we found **multiple series on the right side** for this match group.
Possible fixes:

- **Allow one-to-many matching**: If you want to allow one-to-many matching, you need to explicitly request it by adding a `group_right()` modifie to the operator:

```
… / ignoring(code) group_right() …
```

- **Update your matching parameters:** Consider including more differentiating labels in your matching modifiers (via `on()` / `ignoring()`) to split multiple series into distinct match groups.
- **Aggregate the input:** Consider aggregating away the extra labels that create multiple series per group before applying the binary operation.

{**handler**="/api/v1/query_range", **instance**="demo.do.prometheus.io:9090", **job**="prometheus"}

■ prometheus_http_request_duration_seconds_sum{}

/

{**handler**="/api/v1/query_range", **instance**="demo.do.prometheus.io:9090", **job**="prometheus"}

■ prometheus_http_requests_total{**code**="200"}

■ prometheus_http_requests_total{**code**="503"}

■ prometheus_http_requests_total{**code**="400"}

=

error, result omitted

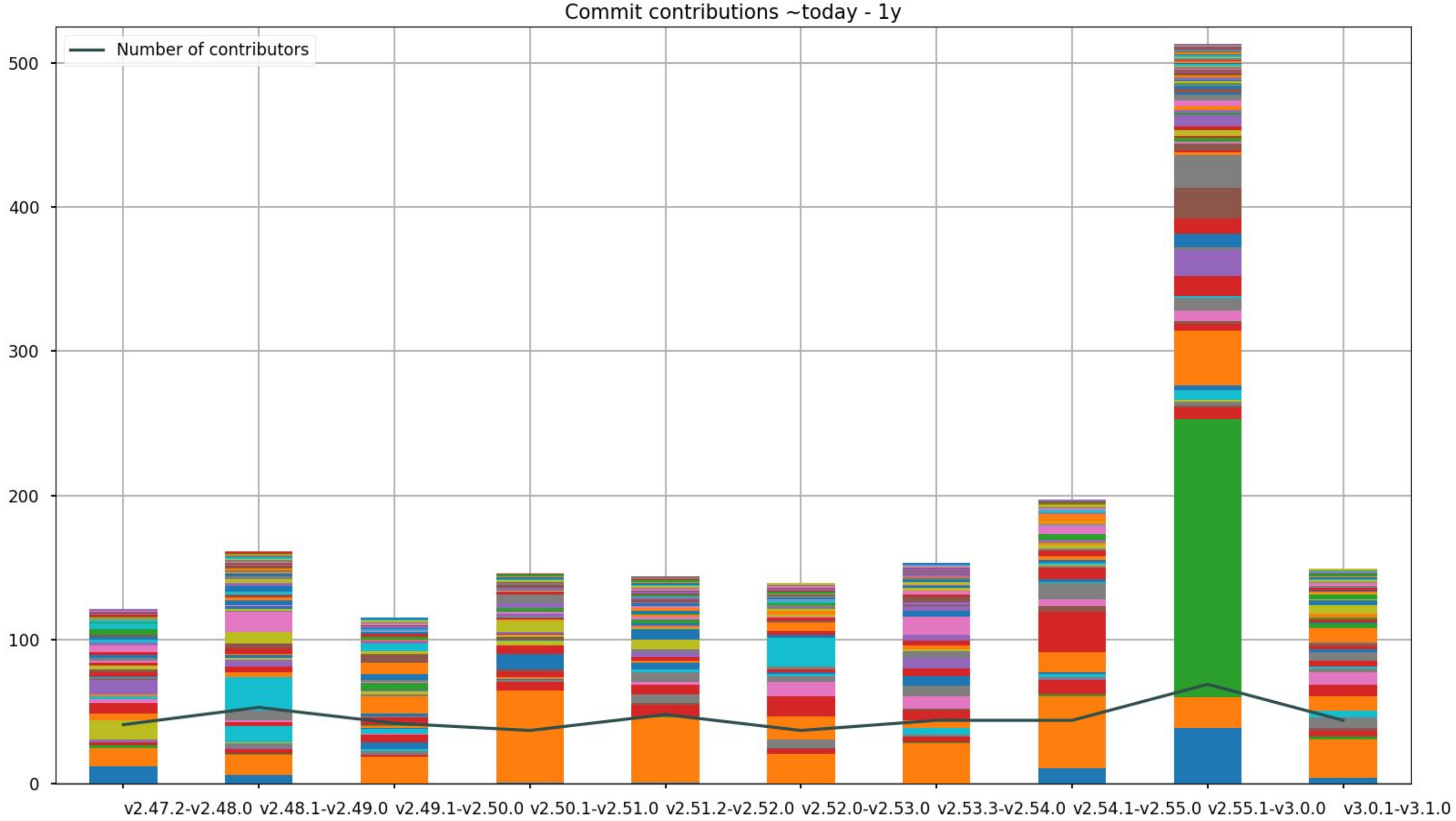
What went wrong?

What went wrong in 3.0?

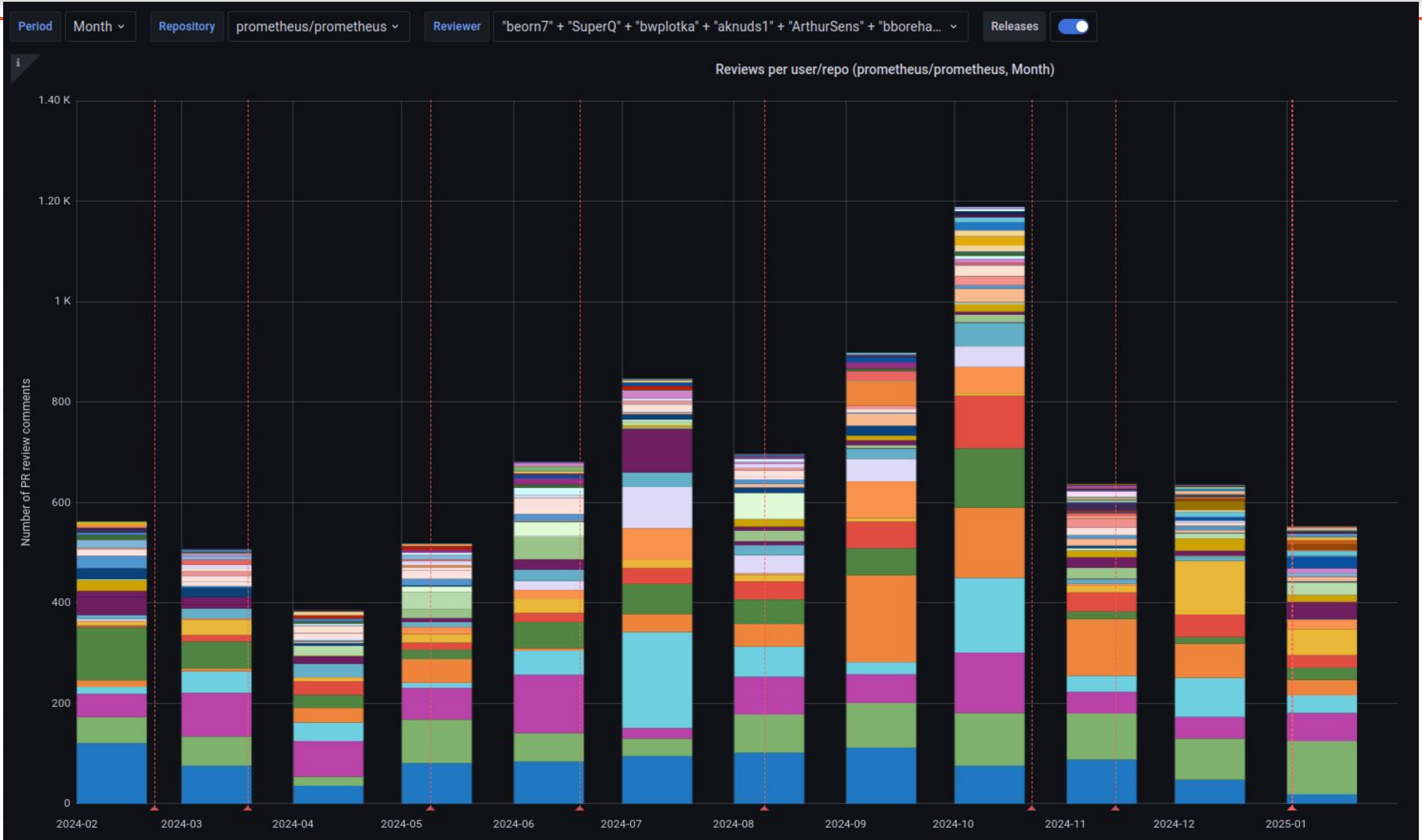
Just three things needed fixing quickly:

- [BUGFIX] Promql: Make subqueries left open. #15431
- [BUGFIX] Fix memory leak when query log is enabled. #15434
- [BUGFIX] Support utf8 names on /v1/label/:name/values endpoint. #15399

Contribution development


Commit contributions ~today - 1y
Number of contributors
500
400
300
200
100
0
v2.47.2-v2.48.0
v2.48.1-v2.49.0
v2.49.1-v2.50.0
v2.50.1-v2.51.0
v2.51.2-v2.52.0
v2.52.0-v2.53.0
v2.53.3-v2.54.0
v2.54.1-v2.55.0
v2.55.1-v3.0.0
v3.0.1-v3.1.0

Review contributions per month



Period: Month

Repository: prometheus/prometheus

Reviewer: "beorn7" + "SuperQ" + "bwplotka" + "aknuds1" + "ArthurSens" + "bboreha...

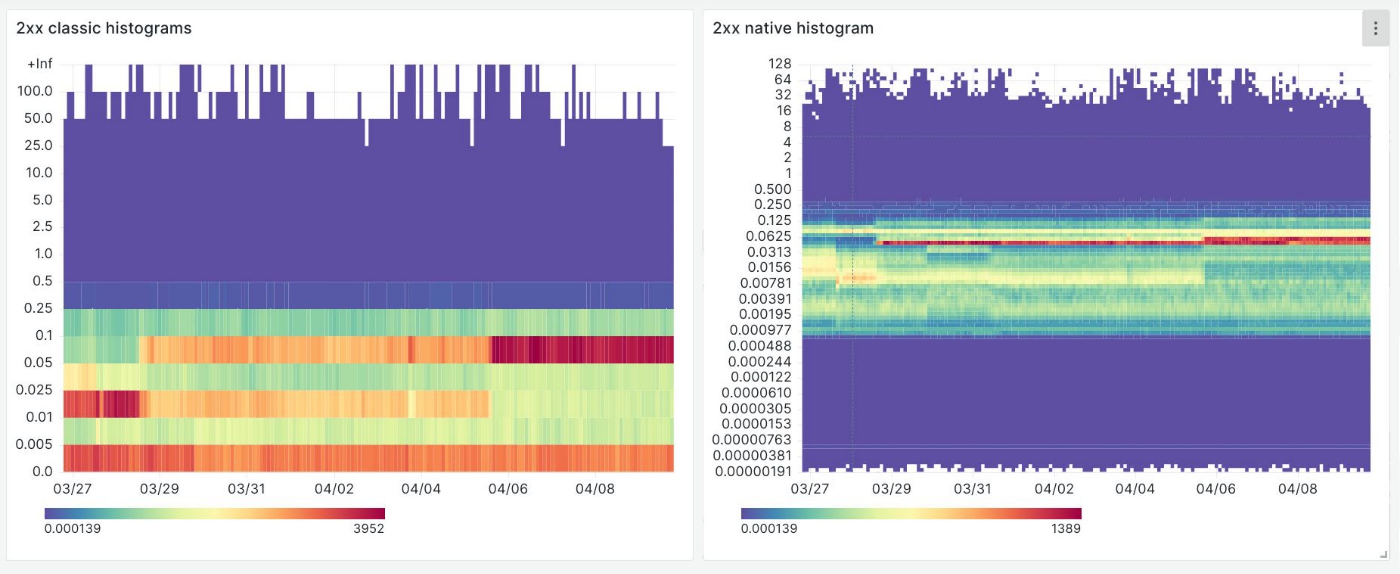
Releases

Join Us!

Questions?

What did we not quite get to?

Native histograms



2xx classic histograms

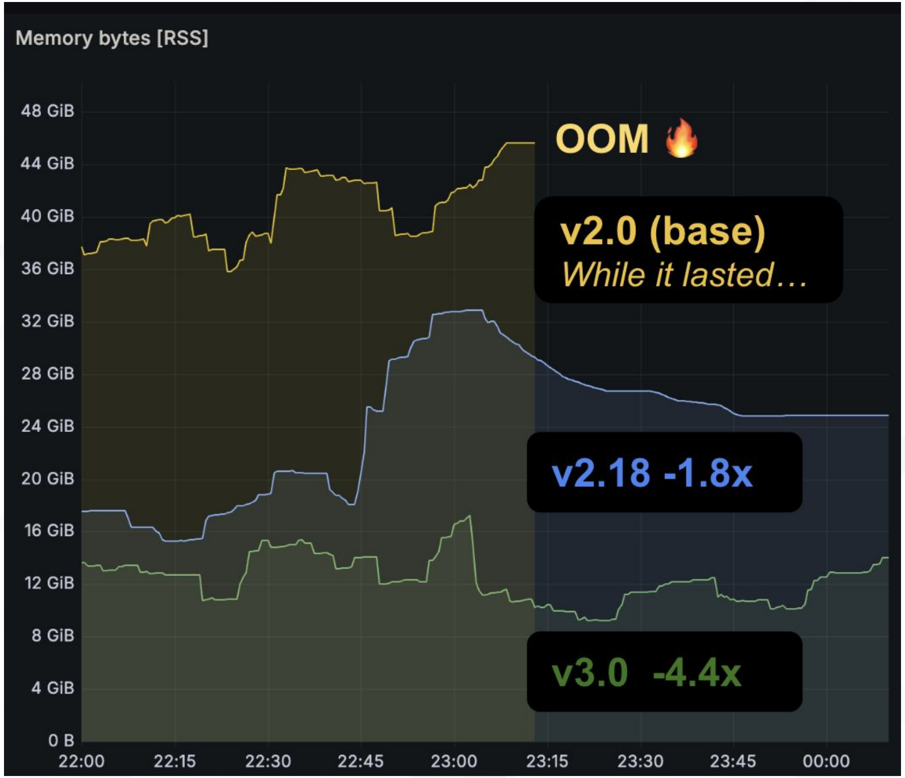
2xx native histogram

What did we not quite get to?

- Better OTEL support
  - Delta temporality
  - Resource attributes

Performance

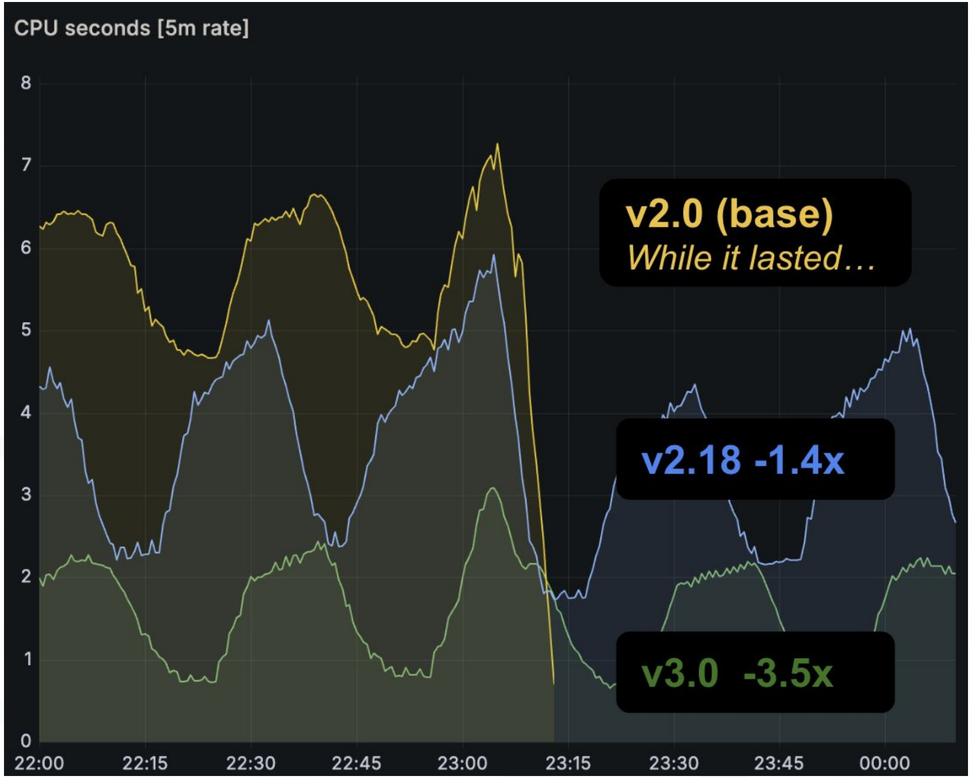
Prometheus Performance



v2.0 (base)
While it lasted…

v2.18 -1.8x

v3.0 -4.4x

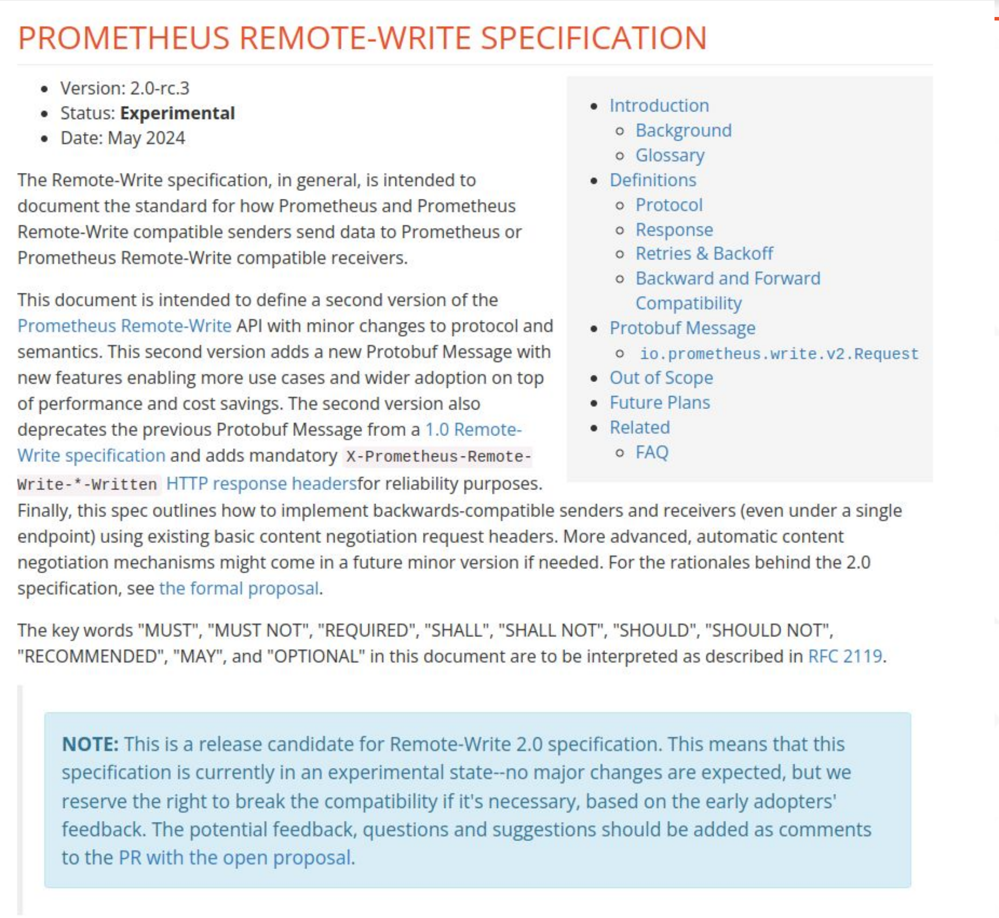


v2.0 (base)
While it lasted…

v2.18 -1.4x

v3.0 -3.5x

Remote Write 2.0



PROMETHEUS REMOTE-WRITE SPECIFICATION

- Version: 2.0-rc.3
- Status: **Experimental**
- Date: May 2024

The Remote-Write specification, in general, is intended to document the standard for how Prometheus and Prometheus Remote-Write compatible senders send data to Prometheus or Prometheus Remote-Write compatible receivers.

This document is intended to define a second version of the Prometheus Remote-Write API with minor changes to protocol and semantics. This second version adds a new Protobuf Message with new features enabling more use cases and wider adoption on top of performance and cost savings. The second version also deprecates the previous Protobuf Message from a 1.0 Remote-Write specification and adds mandatory `X-Prometheus-Remote-Write-*-Written` HTTP response headersfor reliability purposes. Finally, this spec outlines how to implement backwards-compatible senders and receivers (even under a single endpoint) using existing basic content negotiation request headers. More advanced, automatic content negotiation mechanisms might come in a future minor version if needed. For the rationales behind the 2.0 specification, see the formal proposal.

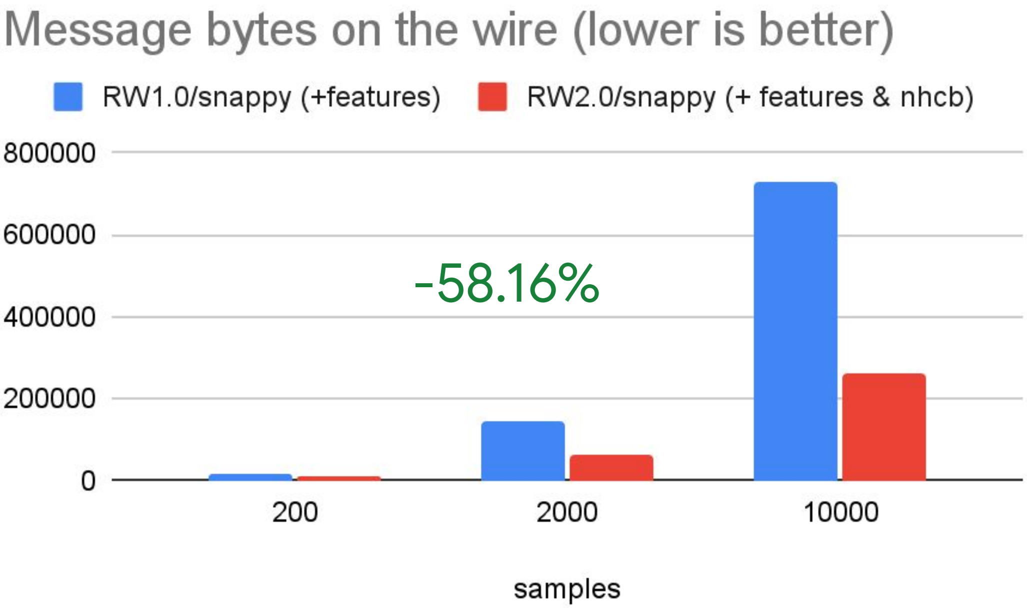
The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119.

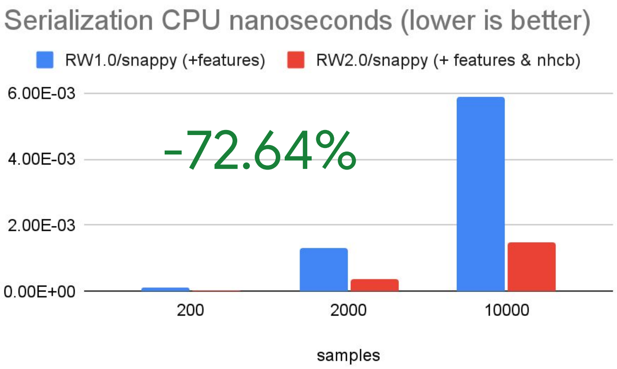
> **NOTE:** This is a release candidate for Remote-Write 2.0 specification. This means that this specification is currently in an experimental state--no major changes are expected, but we reserve the right to break the compatibility if it's necessary, based on the early adopters' feedback. The potential feedback, questions and suggestions should be added as comments to the PR with the open proposal.

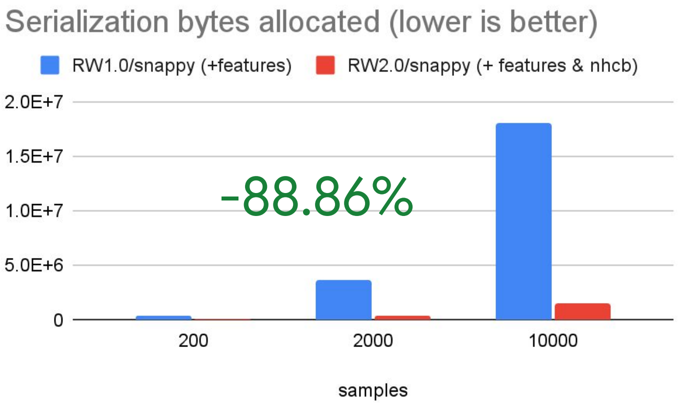
- PRW 2.0 Spec has native support for
  - Native Histograms
  - Created Timestamp
  - Exemplars
  - UTF-8 support
- Structure and Transactionality
- Enables partial write stats

https://www.youtube.com/watch?v=o5HpeMtpsTg&pp=ygUYa3ViZWNvbiByZW1vdGUgd3JpdGUgMi4w

Remote Write 2.0 Performance







Thanks to cstyan and bwplotka https://sched.co/1i7kO