Prometheus

Maintainer Session

Kemal Akkoyun
Polar Signals, @kakkoyun

Bryan Boreham
Grafana Labs, @bboreham

Prometheus team members

Show of hands

What is Prometheus?

Metrics-based monitoring & alerting stack.

- Instrumentation for applications and systems
- Metrics collection and storage
- Querying, alerting, dashboarding
- For all levels of the stack!

Made for dynamic cloud environments.

History

https://youtu.be/rT4fJNbfe14

- Started 2012 at SoundCloud
- Fully open-sourced in 2015
- Joined CNCF, Prometheus v1.0.0 released in 2016
- Prometheus v2.0.0 released in 2017
- Graduated as a CNCF project in 2018

Growth in adoption

2016	2017	2018	2019	2020	2022
2.8k	16k	54k	242k	571k	774k

(Figures from Grafana Labs: count of Grafana installations with a Prometheus datasource)

Community

prometheus / prometheus 11,093 commits Starred 47.7k

Julien Pivotto

to prometheus-develop

Welcomin

Dear community,

Welcoming Chris Sinjakli to the Prometheus team 118 views

Matthias Loibl (MetalMatze)

to Prometheus Developers

Hey everyone,

The Prometheus team is growing: I am happy to announce that Chris Sinjakli is jo

Chris is 3rd in contributions to client_ruby and has been active in client_ruby last

https://github.com/Sinjo

Welcome Chris and thanks for your continuous work!

Cheers,
Matthias / metalmatze

Dear Prometheans,

The Prometheus team is growing: I am happy to announce that Kemal Akko

Kemal has been active as a maintainer in prometheus/client_golang.
He has published releases such as v1.11 (June 2021), v1.12 (Jan 2022), and
Next to these contributions he is also a maintainer of projects like Promethe

https://github.com/kakkoyun

Welcome Kemal and thanks for your continuous work!

Cheers,
Matthias / metalmatze

[prometheus-developers] Welcome Jesús Vázquez to the Prometheus team Inbox ×

Ganesh Vernekar <ganeshvern@gmail.com> Mon, Apr 3, 2:55 PM

to Prometheus Unsubscribe

Hi everyone,

Please join the Prometheus team in welcoming Jesús Vázquez (GH: jesusvazquez) as a new team member.

Jesús worked on adding out-of-order sample support to the Prometheus TSDB and has been helping contributors without their contributions. He plans to continue his efforts on the TSDB and I intend to welcome him as a TSDB maintainer as well :)

Welcome, Jesús!

Thanks,
Ganesh

Daniel is 2nd in contributions to client_ruby and has been active in client_ruby.

https://github.com/dmagliola

Welcome Daniel and thanks for your continuous work!

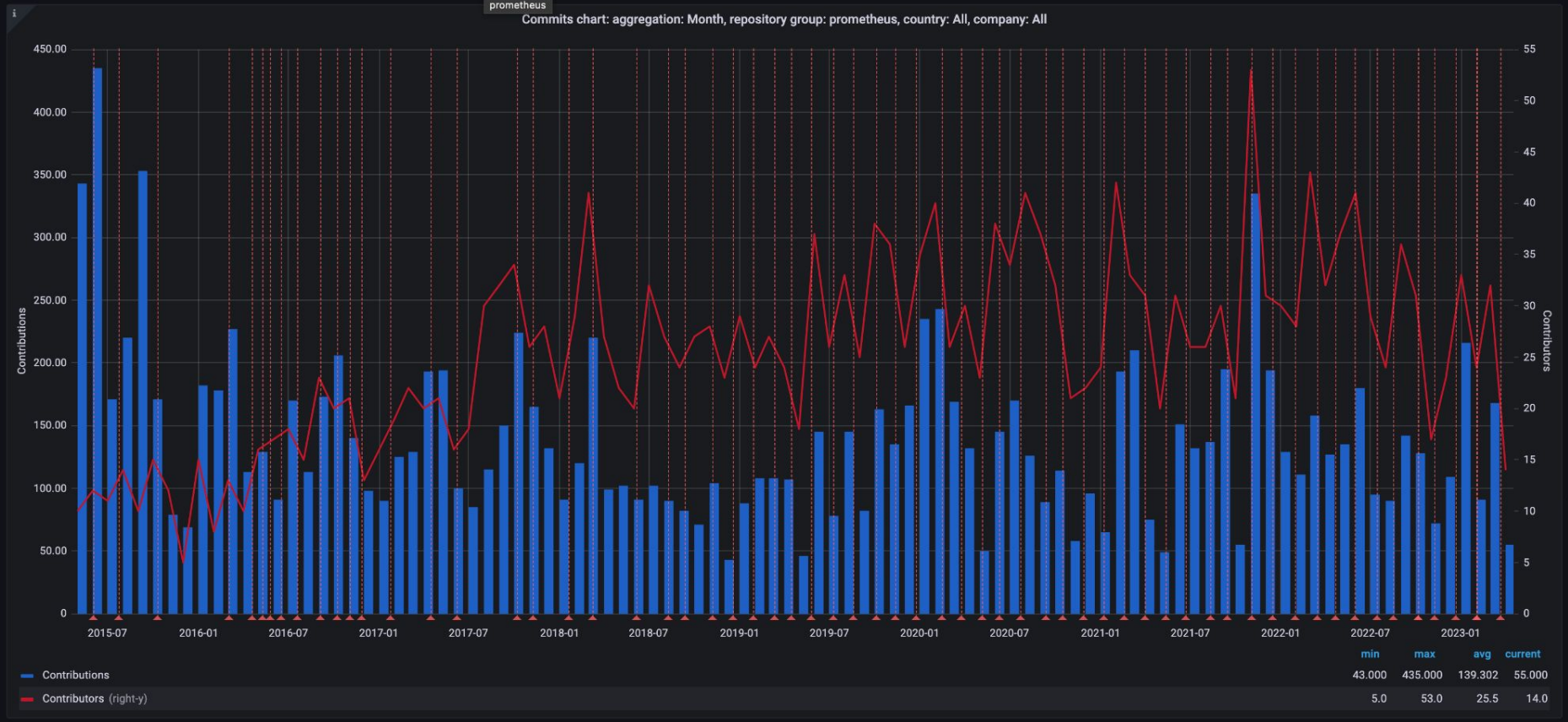
Cheers,
Matthias / metalmatze

General / Contributions chart

now-8y to now-2d

Period: Month | Metric: Commits | Repository group: prometheus | Country: All | Company: All | Releases

prometheus



What is it not?

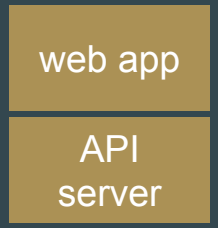
Prometheus does not do:

- Logging or tracing
- Automatic anomaly detection

Architecture

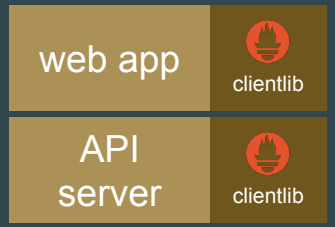
Architecture

Targets



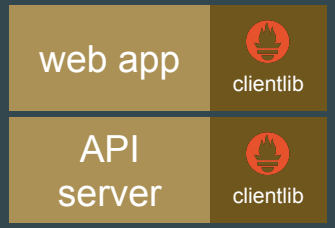
Architecture

Targets



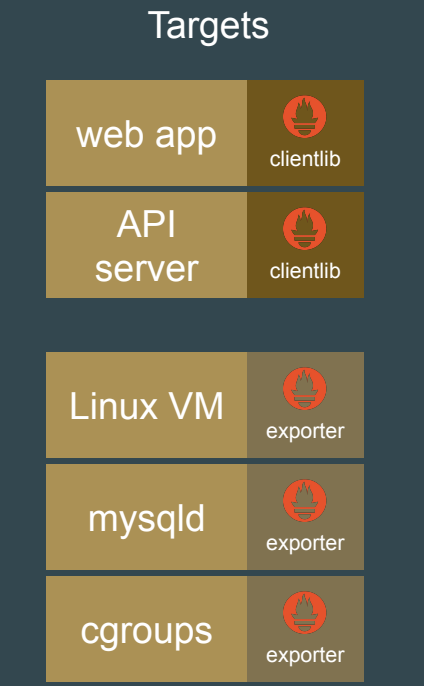
Architecture

Targets

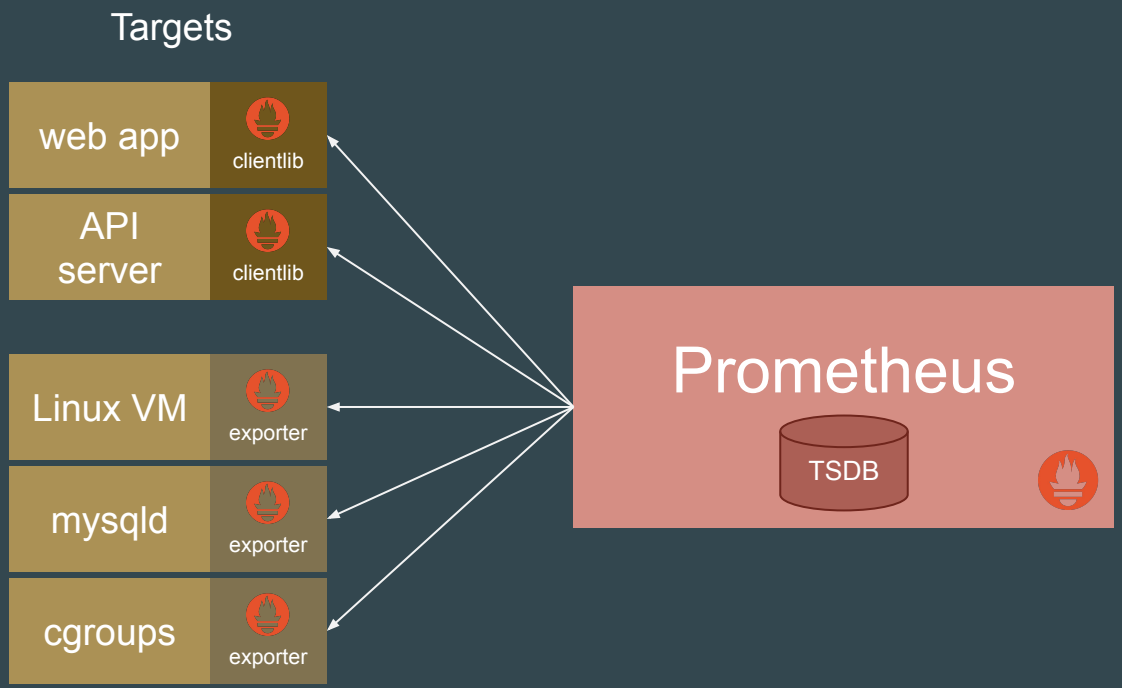


Instrumentation & Exposition

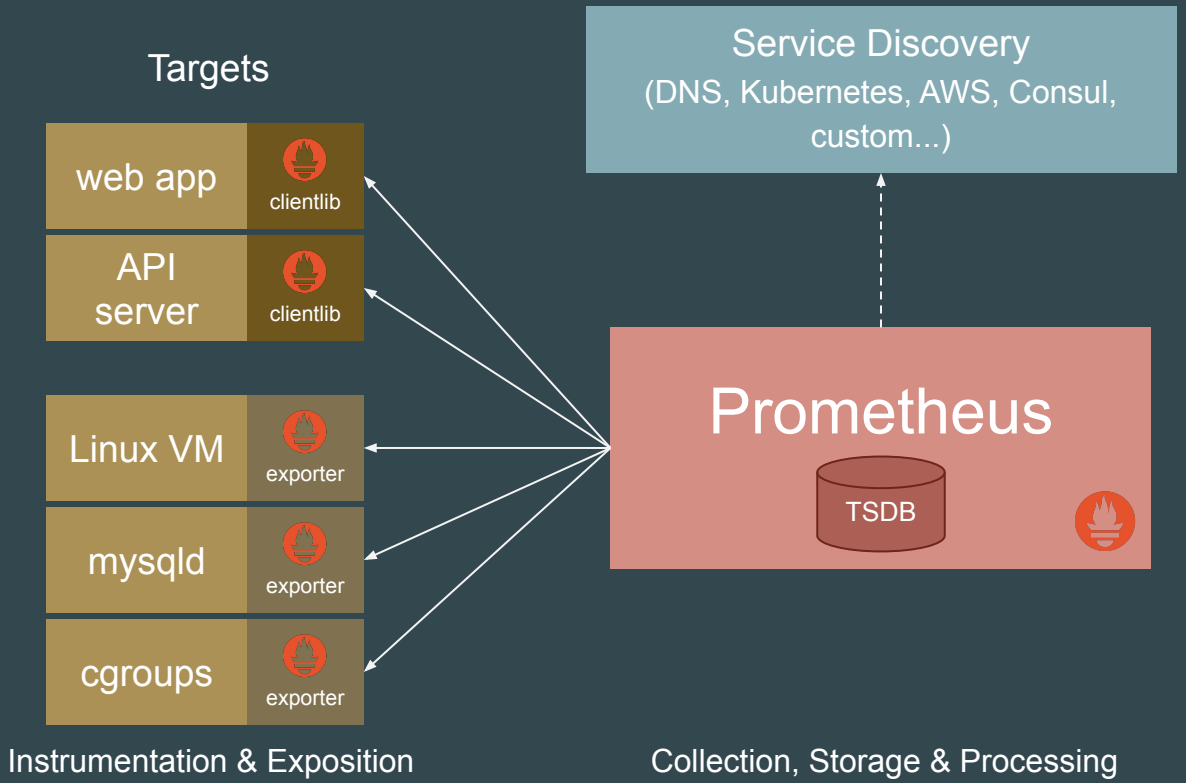
Architecture



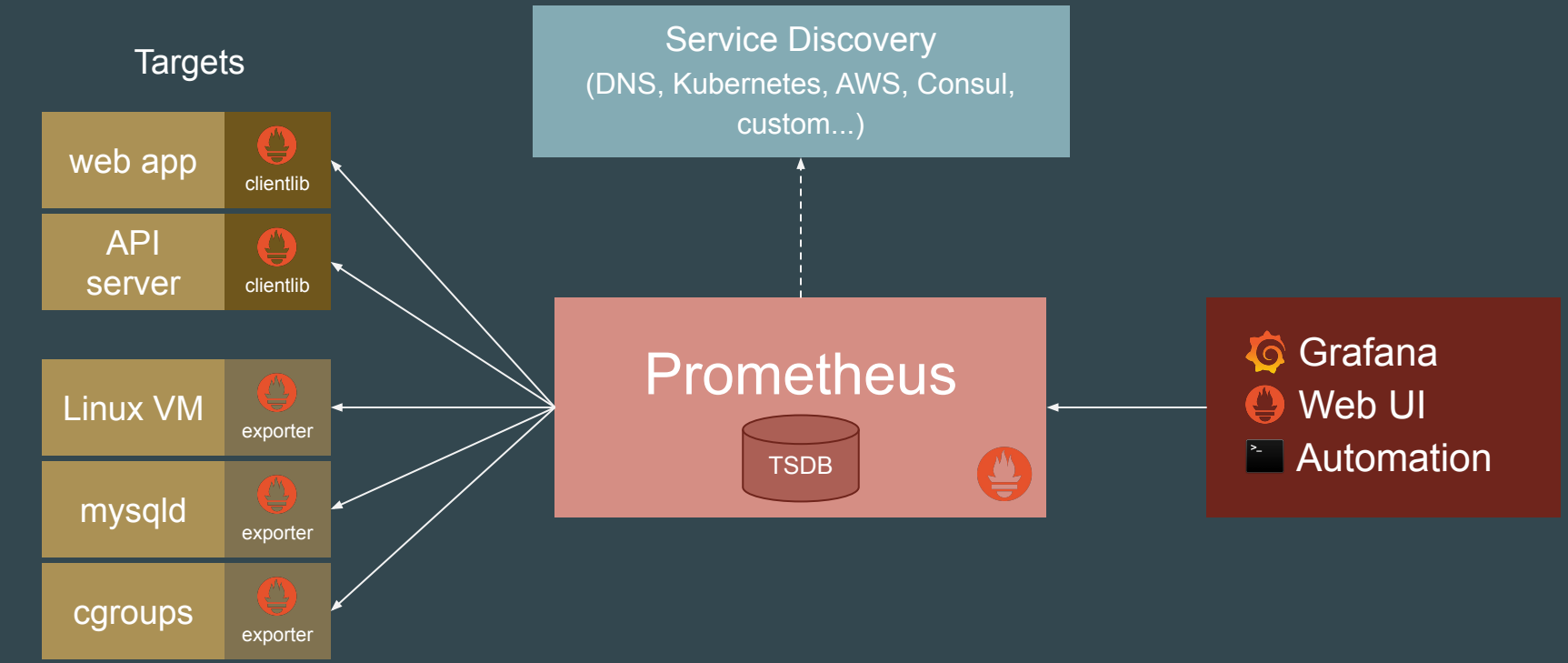
Architecture



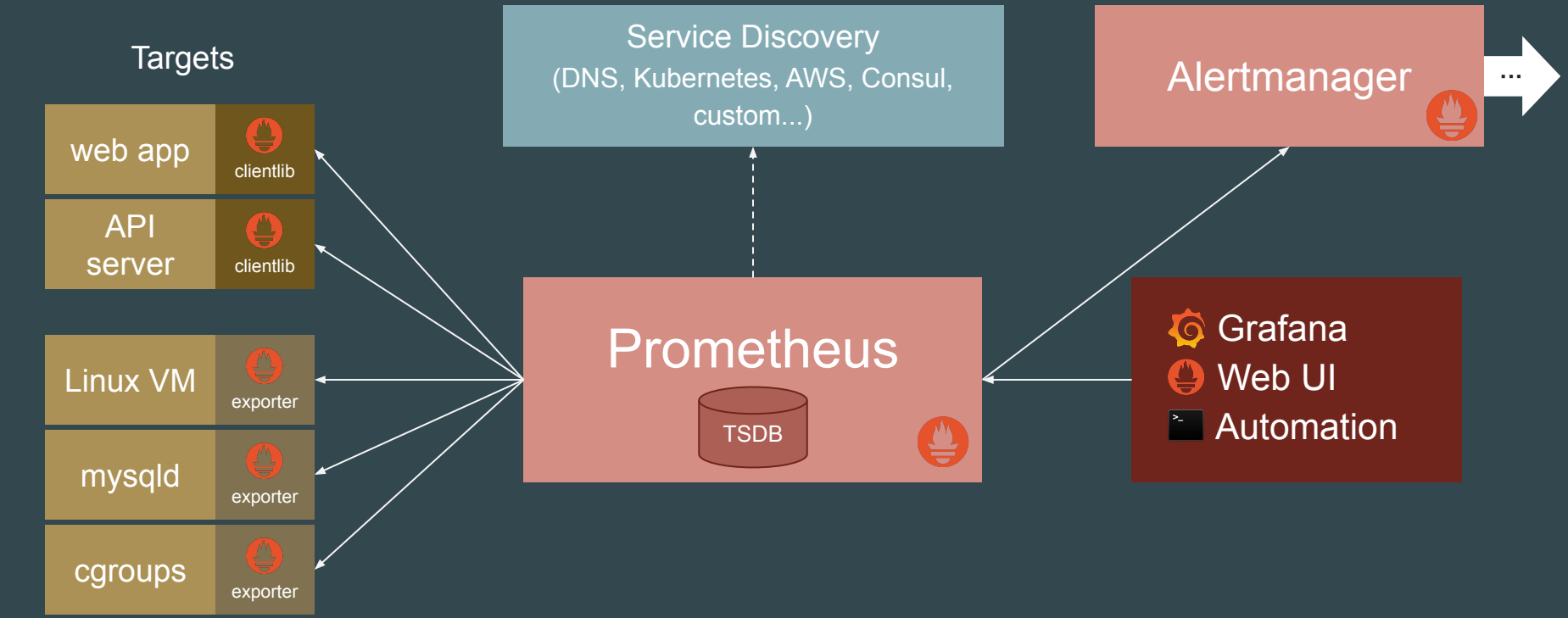
Architecture



Architecture



Architecture



What's new

v2.39 - 2.43

Some of the new features released in the last ~6 months only looking at prometheus/prometheus

Native Histograms

v2.40 + 2.42

Add Support for Native Histograms #11447

Merged **codesome** merged 228 commits into `main` from `sparsehistogram` on Oct 26. 2022

Conversation

codesome

This PR m
This PR is
branch to

Design do
https://doc
Some snea

We would

6

Disclaimer:

Native Histograms are an experimental feature!

Everything described here can still change!

Things might break or behave weirdly!

HAZARD WARNING

```
$ prometheus --enable-feature=native-histograms
```

rams and breaking changes.

e the `wal` directory when upgrading.
mpatible in v2.42.0, this will lead to some data loss

native histograms in v2.42.0 (which writes float

ams.

experimental native histograms. #11783

DB dump. #11872

rules. #11838

rgets for Kubernetes. #11844

ery. #11650

0

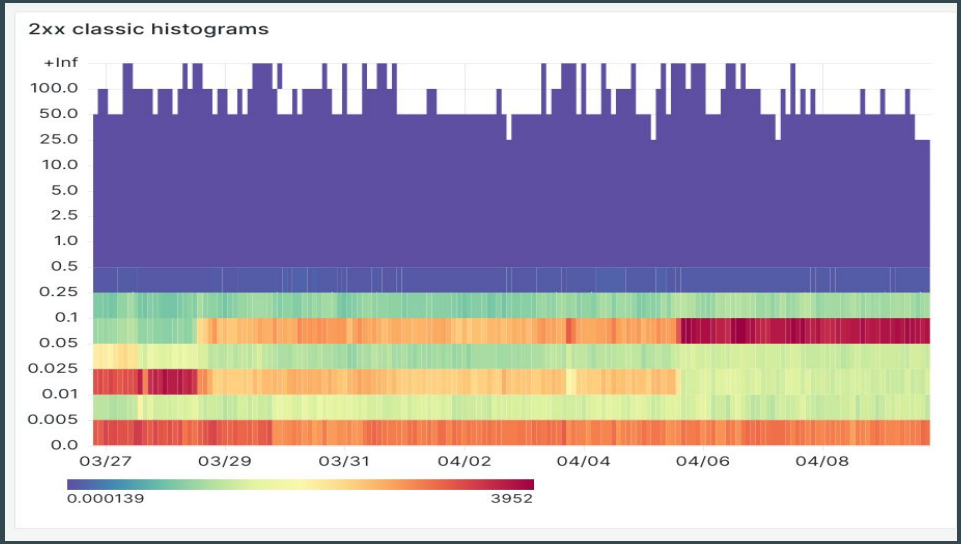
#11840 #11814

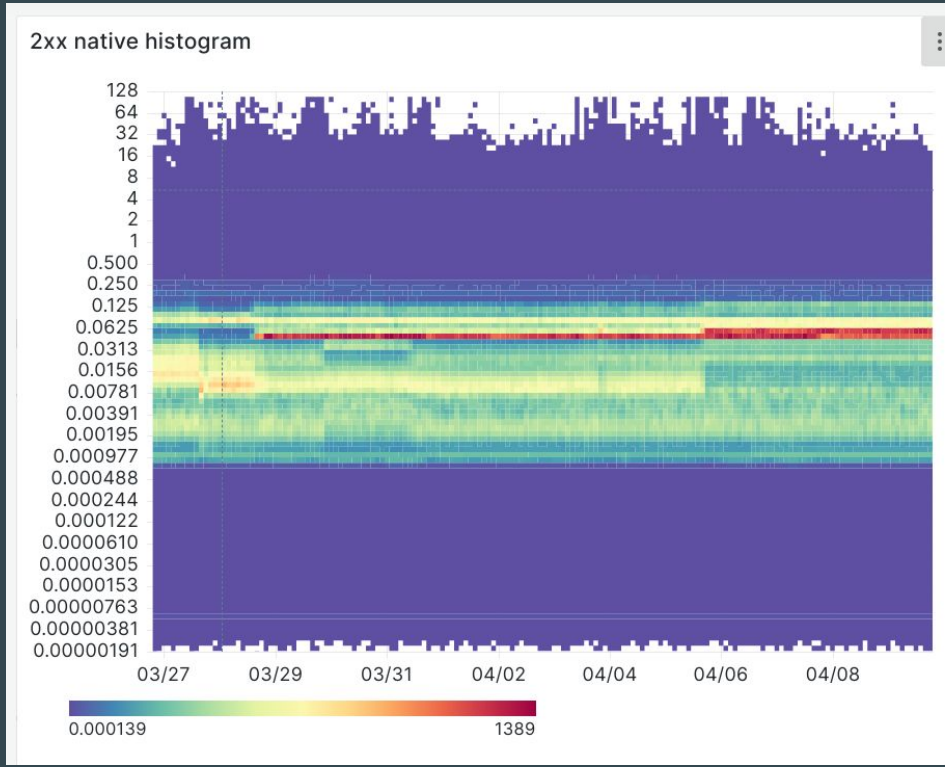
presents buckets as float64 values. #11522 #11817

#11716

Native Histograms

v2.40 + 2.42

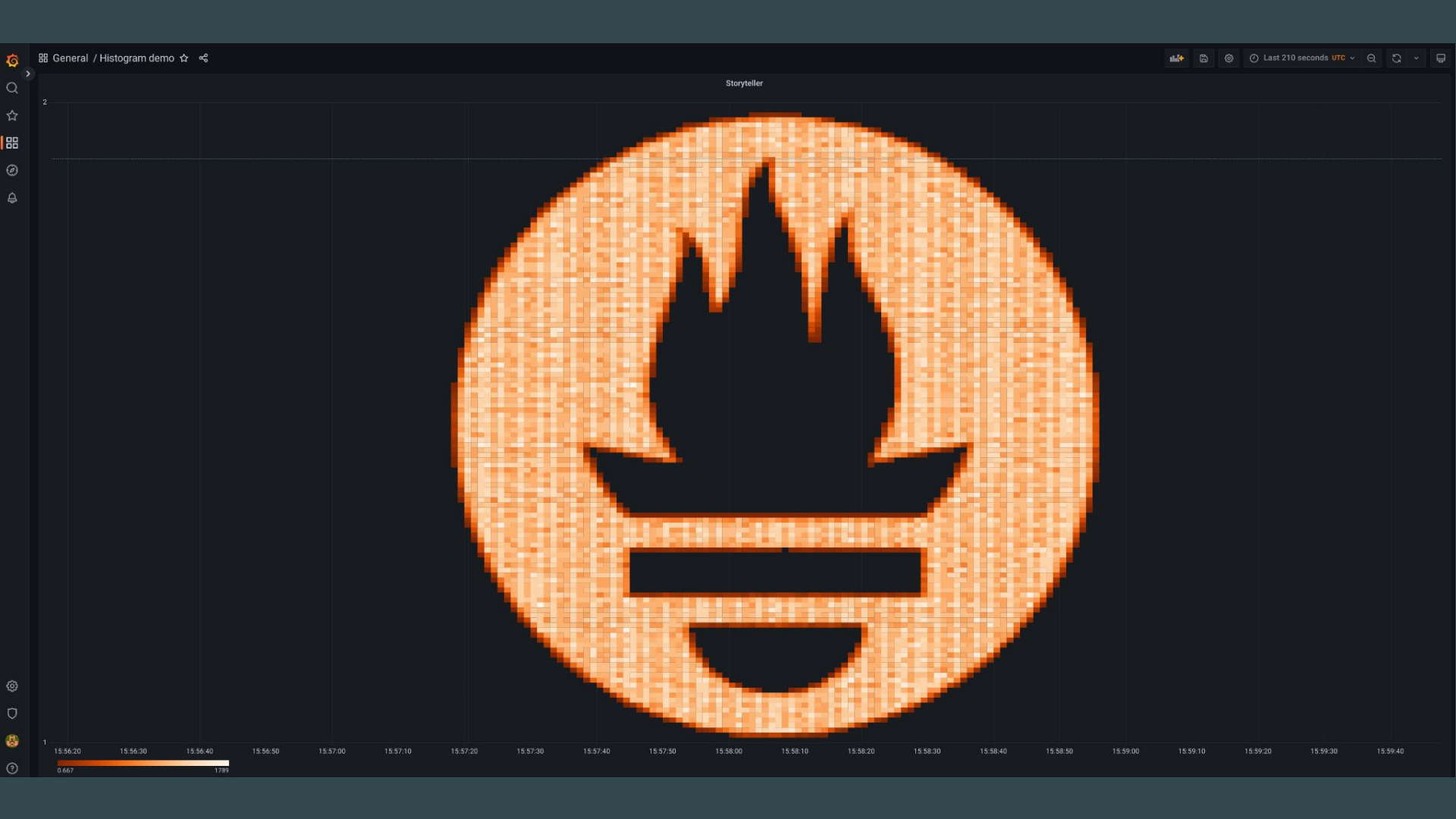




https://colocatedeventseu2023.sched.com/event/1Jo82/prometheus-native-histograms-in-production-bjorn-rabenstein-grafana-labs

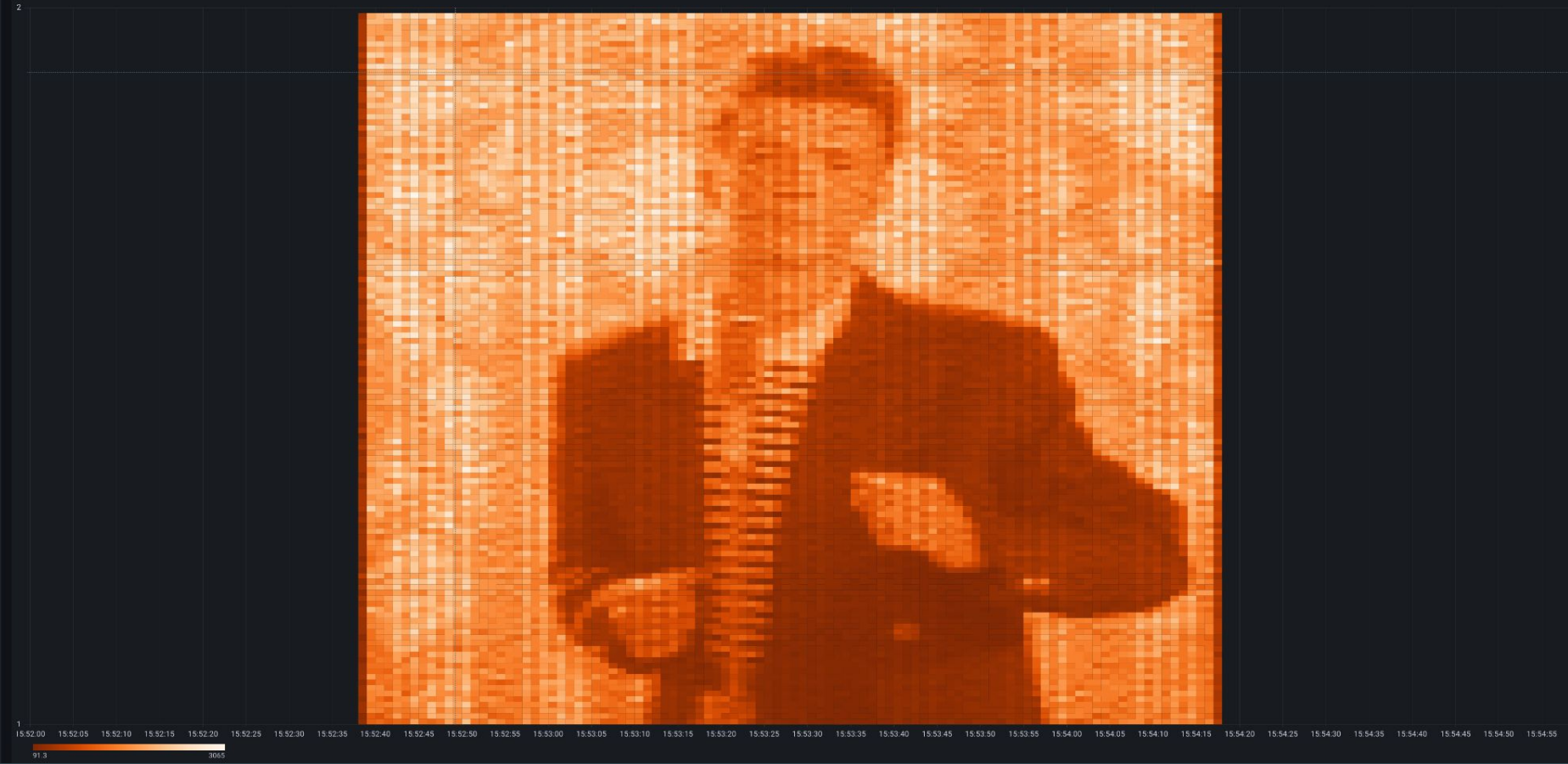
https://promcon.io/2022-munich/talks/native-histograms-in-prometheus

https://promcon.io/2022-munich/talks/promql-for-native-histograms

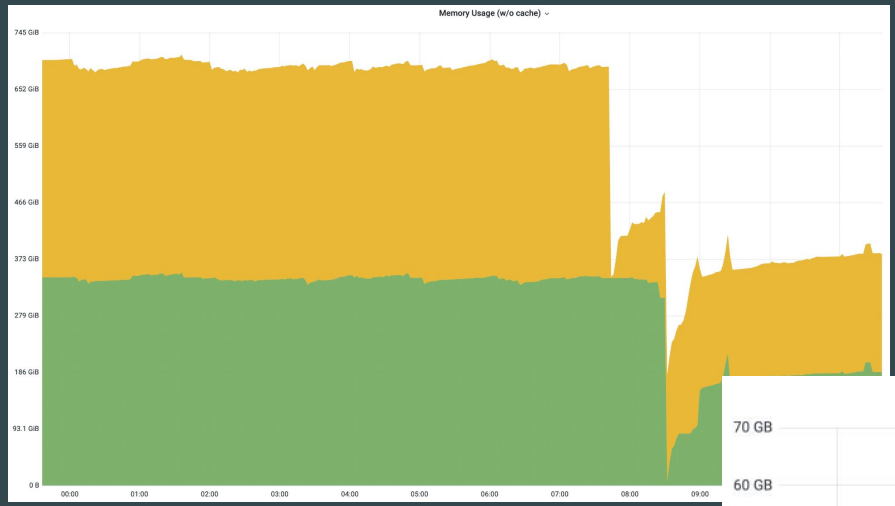

General / Histogram demo
Last 210 seconds UTC
Storyteller
2
1
15:56:20
15:56:30
15:56:40
15:56:50
15:57:00
15:57:10
15:57:20
15:57:30
15:57:40
15:57:50
15:58:00
15:58:10
15:58:20
15:58:30
15:58:40
15:58:50
15:59:00
15:59:10
15:59:20
15:59:30
15:59:40
0.667
1789

Last 3 minutes UTC

Storyteller

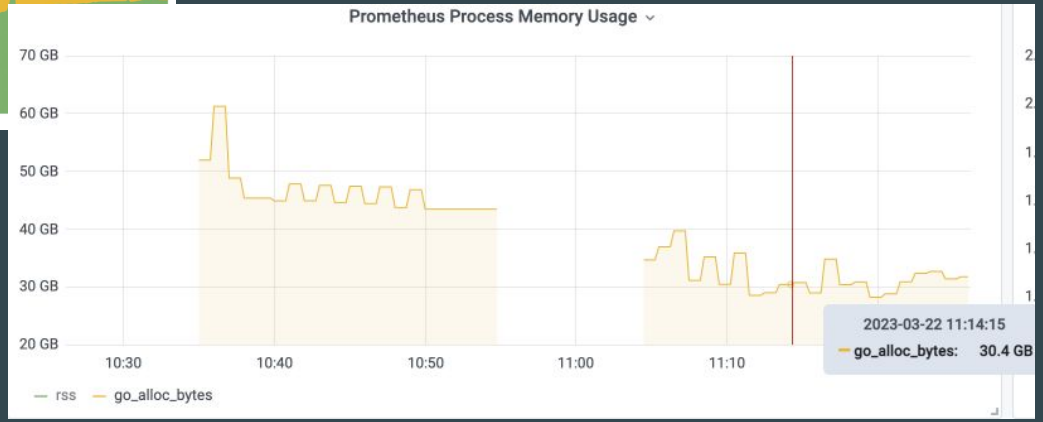


Memory Reductions



v2.39

v2.44



keep_firing_for v2.42

Add 'keep_firing_for' field to alerting rules

```
...
# How long an alert will continue firing after
the condition that triggered it
# has cleared.
[ keep_firing_for: <duration> | default = 0s ]
...
```

What's Coming

- New Alertmanager UI
  - New React UI to align with the rest of the Prometheus Ecosystem
- Metadata Improvements
  - Metric type
  - Created time
  - Persisted in TSDB
- Exemplar Improvements
  - Retained by recording rules
  - Persisted in TSBD
- Remote Write v2
  - Transactional remote write
  - Reduce bandwidth

Add warnings (and annotations) to PromQL query results #12152

Open **zenador** wants to merge 16 commits into `prometheus:main` from `zenador:bubblebobble`

Conversation 21 | Commits 16 | Checks 22 | Files changed 26

zenador commented on Mar 19 • edited

First-time contributor | Tip | ...

Addresses some of #10839 and #11216 as a hackathon project.

Most of the code changes are just changing from `storage.Warnings` to `notes.Warnings` to move it to a more neutral location as the new warnings we are adding are not related to storage.

Annotations are created and handled separately from warnings here, but they are merged together with the warnings in the PromQL engine because the HTTP API and the UI don't support it yet and we want to avoid making external changes for now. By creating annotations separately though, it would make it easier in future when we want to display them differently.

Due to the way warnings are created, we end up with many duplicate warnings, or perhaps we find the warning is not needed later on, e.g. for the `RangeTooSmallWarning`, we may add this warning at each section of the range, or we may only add it at the beginning or end but still return a non-empty result for the query. For now we do the deduplication and the removal of the unnecessary warning at the final response writing step in the API.

New warnings (`util/notes/warnings.go`):

- RangeTooSmallWarning = errors.New("Need at least 2 points to compute, perhaps time range is too small")

Add support for exemplars during recording rule evaluation #12056

Open **vjsamuel** wants to merge 7 commits into `prometheus:main` from `vjsamuel:rules_exemplar`

Conversation 21 | Commits 7 | Checks 22 | Files changed 5

vjsamuel commented on Mar 4 — Contributor — Tip

This PR adds support for recording rules to emit exemplars. Would like to get some early feedback before I add things like unit test cases, feature flags (if necessary).

I wanted to add a sampling ratio to only append a subset of exemplars along with the recorded metric but would love to get feedback before hand.

 **vjsamuel** requested a review from **codesome** as a code owner last month

We need you!

Friday, April 21 • 16:00 - 17:30 Back To Schedule

ContribFest: Prometheus - Hands-on Development and Contribution Workshop (Limited Availability; First-Come, First-Served)

Sign up or log in to save this to your schedule, view media, leave feedback and see who's attending!

https://sched.co/1Hzc Tweet Share

Download the code ahead of time. **DCO Required.**

Wished you know how to write exporter in Go for Prometheus? How to use Prometheus APIs programmatically? Need to quickly instrument you Go code with Prometheus metrics? Join us to learn how to contribute, develop and test Prometheus integrations useful in day to day use. Unblock yourself and others! It's easier than you think!

We will go through useful resources and ways to interact with the project and community, to create meaningful applications that use Prometheus effectively!

e.g.

Upgrade to github.com/Azure/azure-sdk-for-go/sdk #11860

Open **mmorel-35** wants to merge 3 commits into `prometheus:main` from `mmorel-35:azure-sdk-for-go`

Conversation 9 | Commits 3 | Checks 22 | Files changed 4

mmorel-35 commented on Jan 18 • edited by kakkoyun — Contributor — Tip

I had troubles with my commits signature so I recreated this branch with everything merged.
Replaces #11849
Fixes #10942

xref: https://github.com/Azure/azure-sdk-for-go/blob/main/documentation/MIGRATION_GUIDE.md

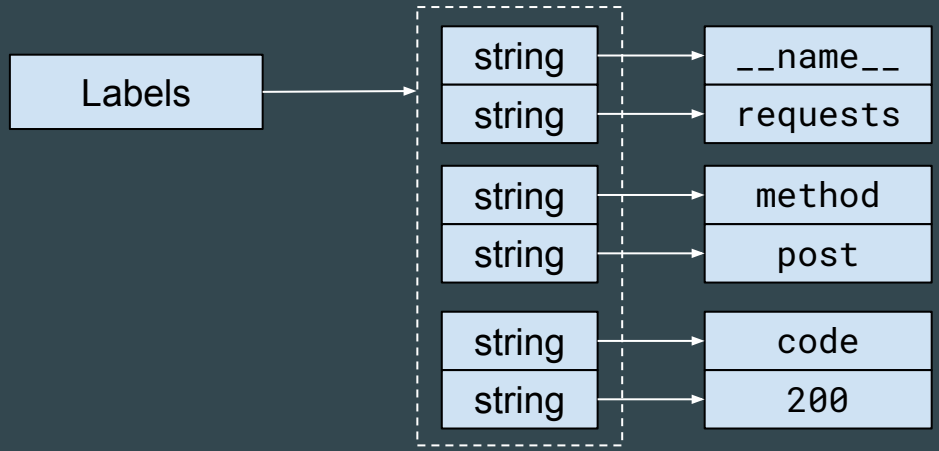
Signed-off-by: Matthieu MOREL matthieu.morel35@gmail.com

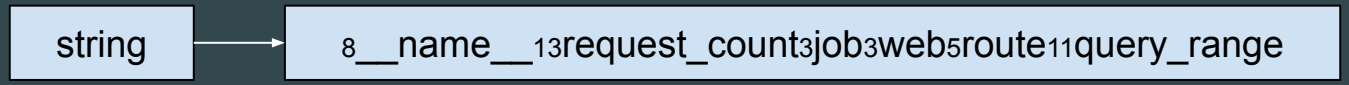
1

Q&A

Thanks!

Labels data structure





Out-of-order ingestion

v2.39

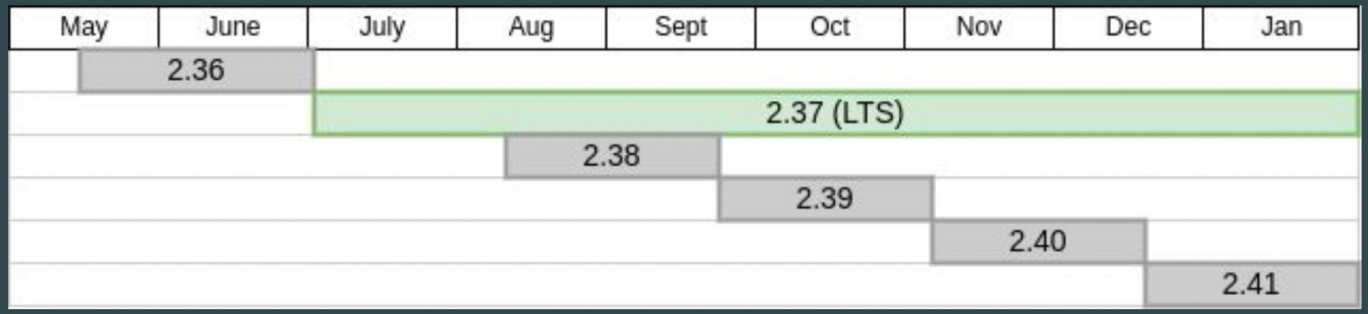
```
$ cat config.yml
...
storage:
  tsdb:
    out_of_order_time_window: 2h
...
```

https://promcon.io/2022-munich/talks/out-of-order-support-in-prom
ethe

LTS releases

LTS for Long Term Support.

A release of Prometheus supported for 6+ months.



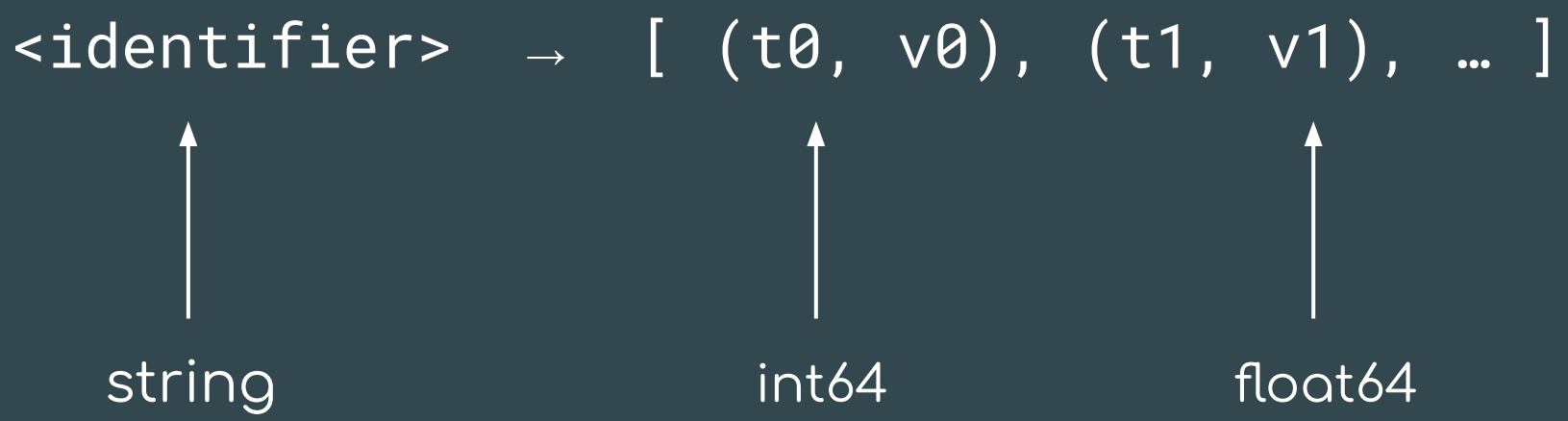
https://prometheus.io/docs/introduction/release-cycle/

Selling Points

- Dimensional data model
- Powerful query language (PromQL)
- Simple & efficient server
- Service discovery integration

Data Model

What is a time series?



Data Model

What identifies a time series?

```
http_requests_total{job="nginx",instance="1.2.3.4:80",status="200"} 28
```

metric name

labels

- Flexible
- No hierarchy
- Explicit dimensions

Querying

Querying

PromQL

- Functional query language
- Great for time series computations
- Not SQL-style

Querying

All partitions in my entire infrastructure with more than 100GB capacity that are not mounted on root?

```
node_filesystem_bytes_total{mountpoint!="/"} / 1e9 > 100
```

```
{device="sda1", mountpoint="/home", instance="10.0.0.1"}        118.8
{device="sda1", mountpoint="/home", instance="10.0.0.2"}        118.8
{device="sdb1", mountpoint="/data", instance="10.0.0.2"}        451.2
{device="xdvc", mountpoint="/mnt", instance="10.0.0.3"}         320.0
```

Querying

What's the ratio of request errors across all service instances?

```
  sum(rate(http_requests_total{status="500"}[5m]))
/ sum(rate(http_requests_total[5m]))
```

```
{}                                  0.029
```

Querying

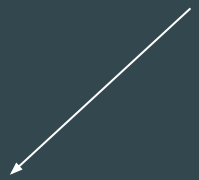
What's the ratio of request errors across all service instances?

```
    sum by(path) (rate(http_requests_total{status="500"}[5m]))
  / sum by(path) (rate(http_requests_total[5m]))
```

{path="/status"}	0.0039
{path="/"}	0.0011
{path="/api/v1/topics/:topic"}	0.087
{path="/api/v1/topics}	0.0342

Alerting

generate an alert for each
path with an error rate of >5%



```
alert: Many500Errors
expr: |
  (
      sum by(path) (rate(http_requests_total{status="500"}[5m]))
    /
      sum by(path) (rate(http_requests_total[5m]))
  ) * 100 > 5
for: 5m
labels:
  severity: "critical"
annotations:
  summary: "Many 500 errors for path {{$labels.path}} ({{$value}}%)"
```

Efficiency

Local storage is scalable enough for many orgs:

- 1 million+ samples/s
- Millions of series
- 1-2 bytes per sample

Good for keeping a few weeks or months of data. Some people keep years, with careful backups.

Bridging the gap

Not everything speaks Prometheus – exporters help

- Translate from other metric systems (statsd, CloudWatch, ...)
- Transform system-specific metrics (Linux, MySQL, HAProxy, ...)
- Do it yourself (JSON exporter, Python, Go, ...)

Conclusion

Prometheus helps you make sense of complex dynamic environments via its:

- Dimensional data model
- Powerful query language
- Simplicity + efficiency
- Service discovery integration