Prometheus Intro, Deep Dive

Gouthām Veeramachaneni
Grafana Labs, @putadent

Ganesh Vernekar
Grafana Labs, @_codesome

Prometheus team members

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!

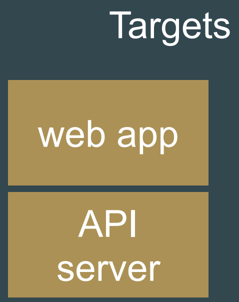
Made for dynamic cloud environments.

History

- Started 2012 at SoundCloud
- Fully publicised in 2015
- Joined CNCF, Prometheus v1.0.0 released in 2016
- Prometheus v2.0.0 released in 2017

Architecture

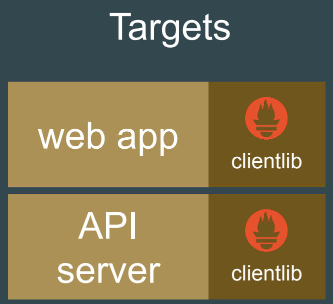
Architecture


Targets

web app

API server

Architecture



Targets

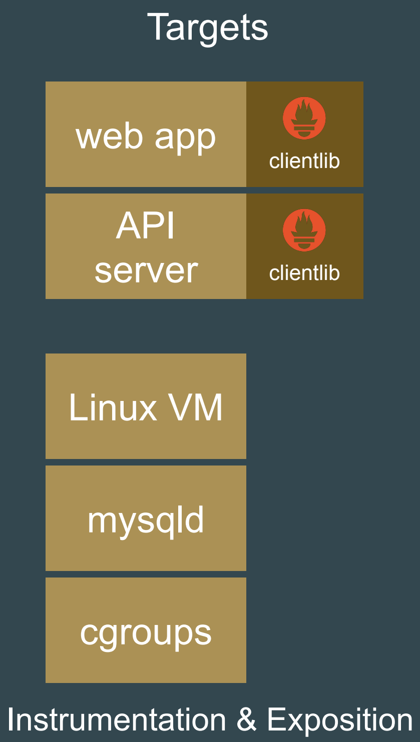
web app

clientlib

API server

clientlib

Architecture



Targets

web app

clientlib

API server

clientlib

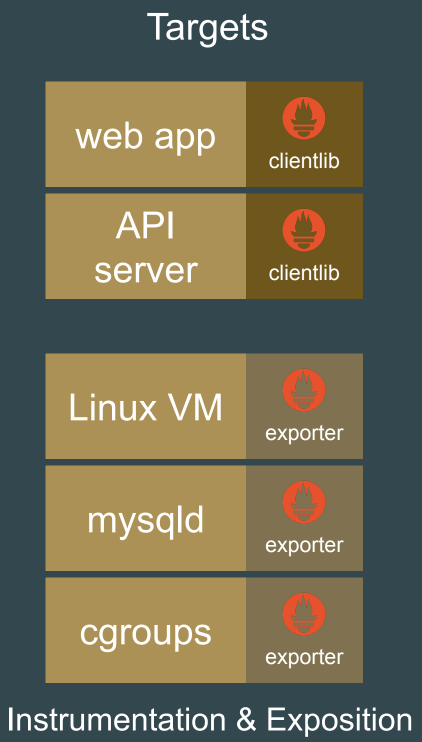
Linux VM

mysqld

cgroups

Instrumentation & Exposition

Architecture



Targets

web app | clientlib

API server | clientlib

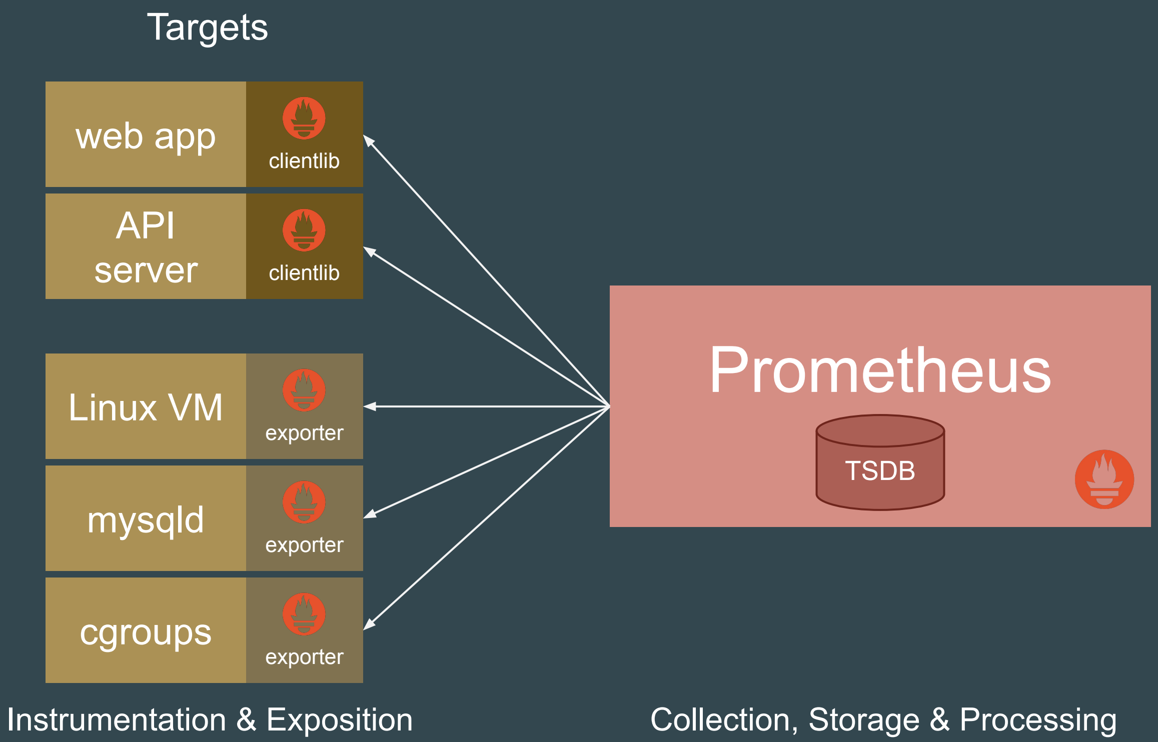
Linux VM | exporter

mysqld | exporter

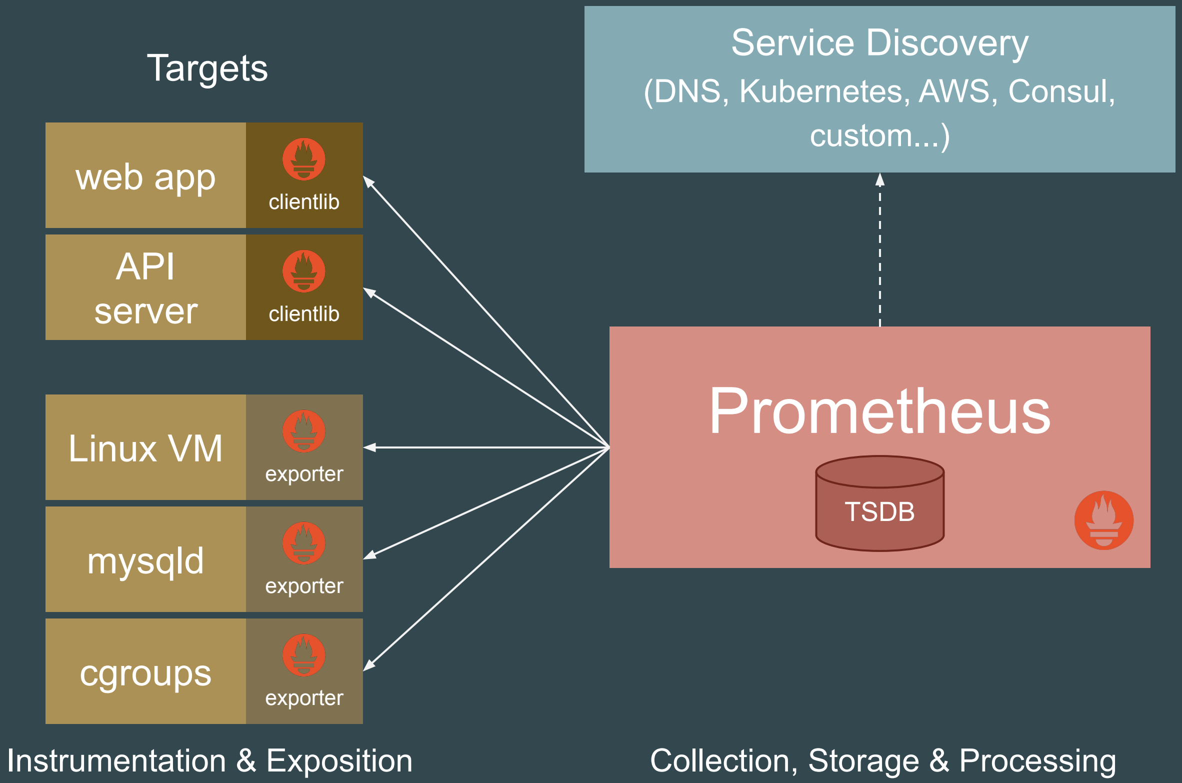
cgroups | exporter

Instrumentation & Exposition

Architecture



Architecture


Targets

web app
clientlib

API server
clientlib

Linux VM
exporter

mysqld
exporter

cgroups
exporter

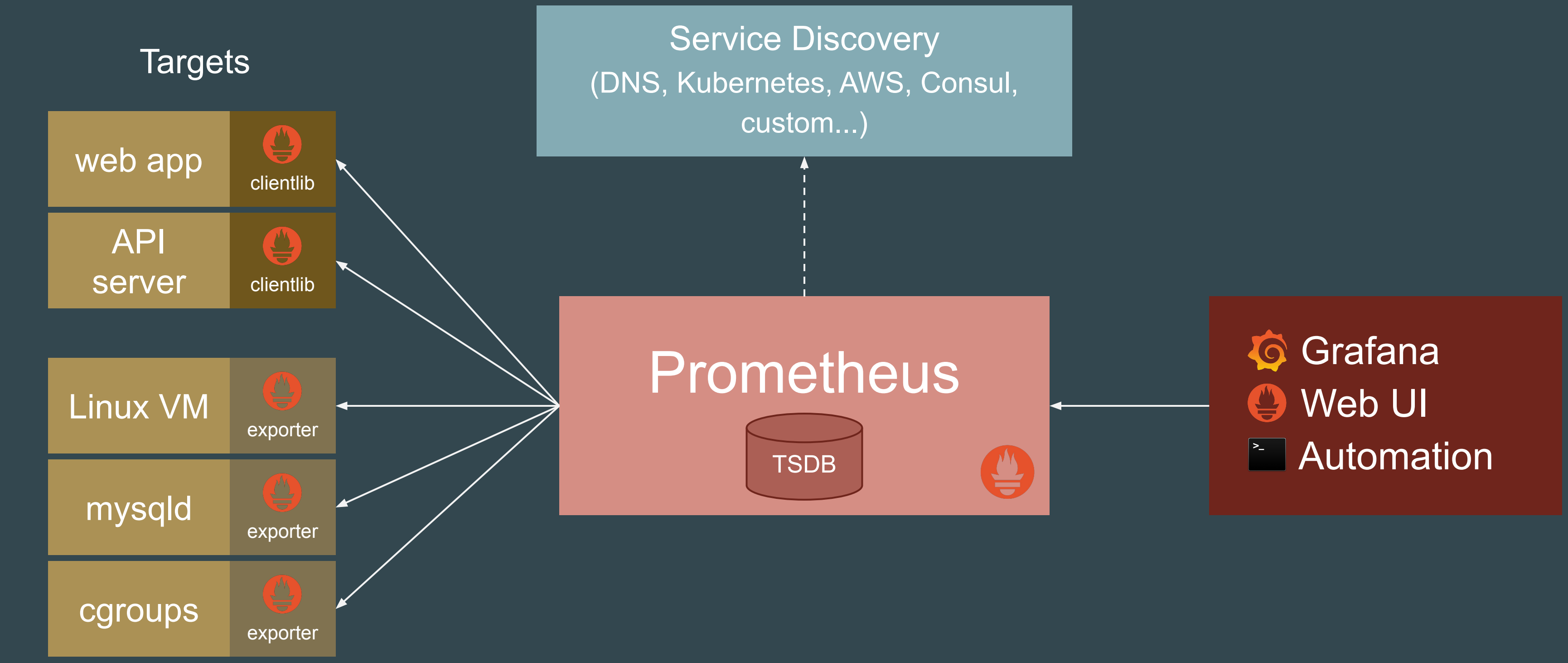
Service Discovery
(DNS, Kubernetes, AWS, Consul, custom...)

Prometheus
TSDB

Instrumentation & Exposition

Collection, Storage & Processing

Architecture

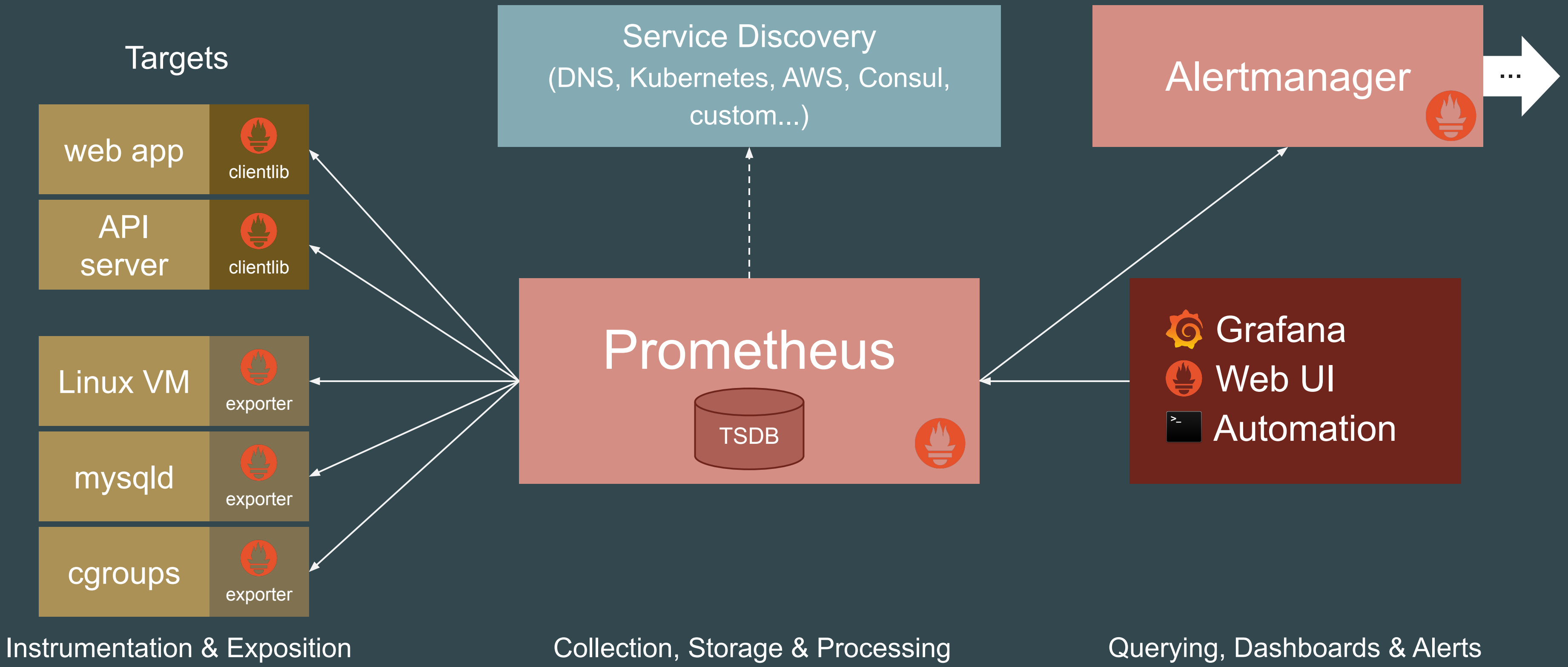


Instrumentation & Exposition

Collection, Storage & Processing

Querying, Dashboards

Architecture

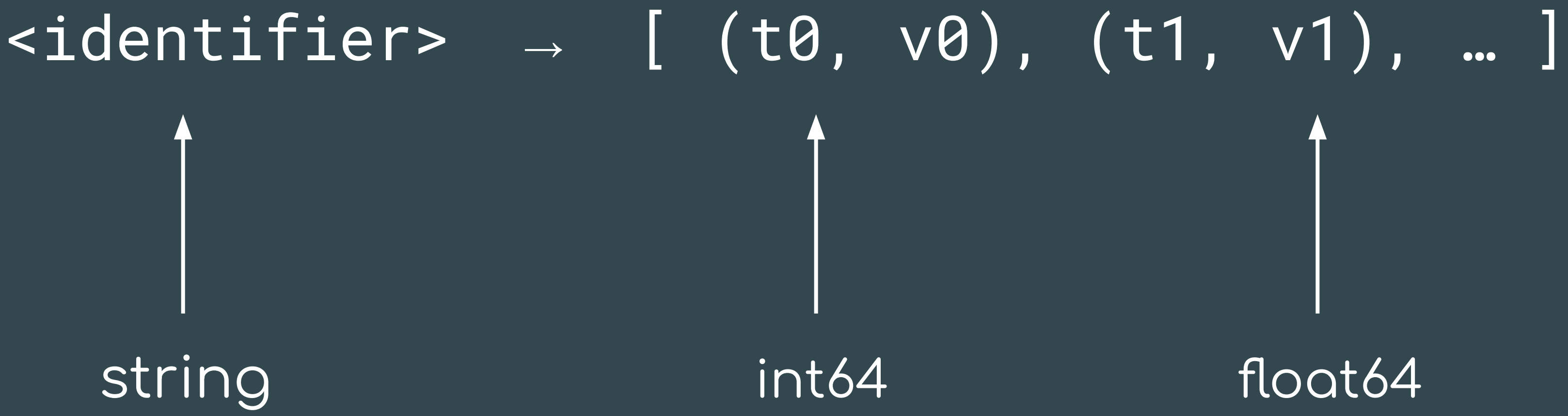


Selling Points

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

Data Model

What is a time series?



`<identifier> → [ (t0, v0), (t1, v1), … ]`

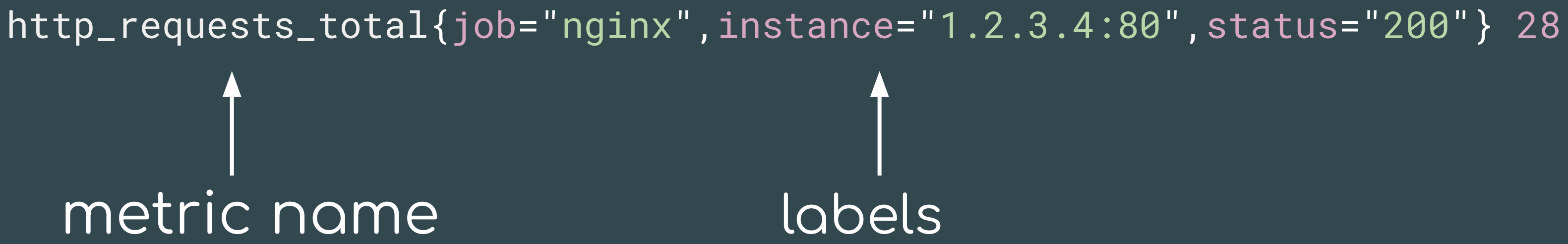
string

int64

float64

Data Model

What identifies a time series?



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

- 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

What’s new

Some of the new features released in the last ~12 months

…only looking at prometheus/prometheus

PromQL

```
temperature_kelvin - avg_over_time(temperature_kelvin[1d] offset -12h)

  rate(http_requests_total[1m])
and
  topk(5, rate(http_requests_total[1h] @ end()))
```

v2.25 (feature flag) v2.33 (stable)

```
sin(rad(wind_turbine_angle_degrees))
x_value atan2 y_value
```

v2.31

Remote Write Receiver

```
$ prometheus --enable-feature=remote-writer-receiver
```

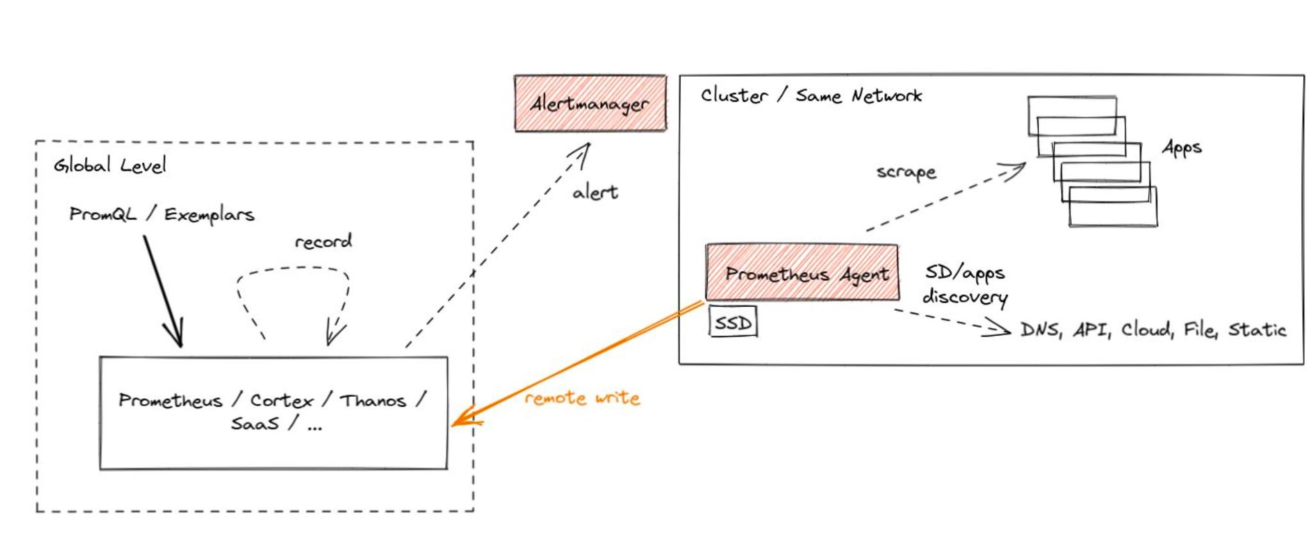
v2.25 (feature flag) v2.33 (stable)

Agent mode

v2.32

```
$ prometheus --enable-feature=agent
```

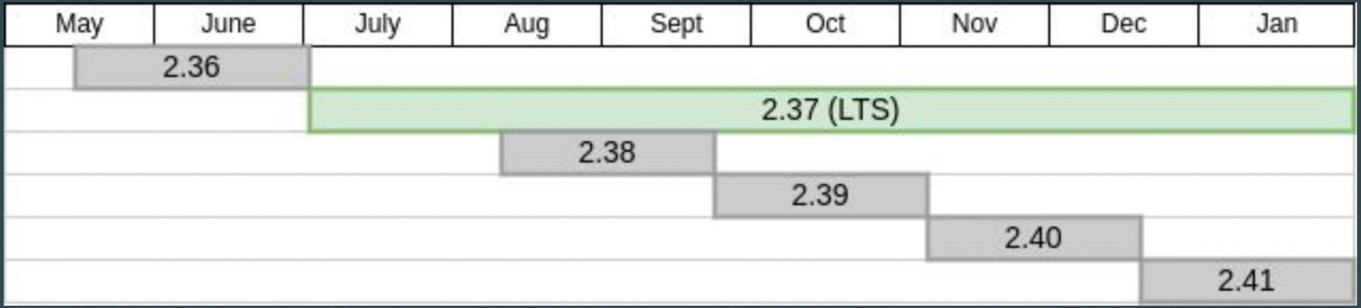
https://prometheus.io/blog/2021/11/16/agent/



LTS releases

LTS for Long Term Support.

A release of Prometheus supported for 6+ months.



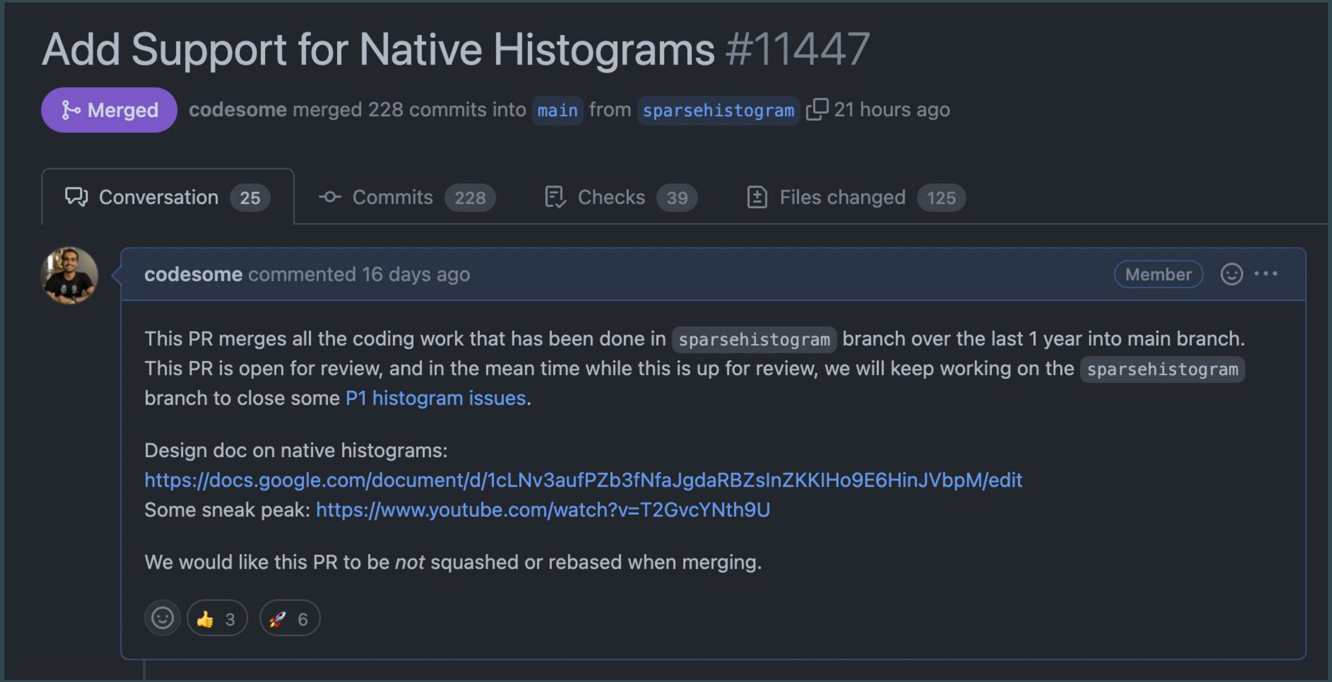
May	June	July	Aug	Sept	Oct	Nov	Dec	Jan
2.36								
		2.37 (LTS)						
			2.38					
				2.39				
						2.40		
							2.41	

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

Out-of-order ingestion v2.39

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

What's Coming



Add Support for Native Histograms #11447

Merged codesome merged 228 commits into `main` from `sparsehistogram` 21 hours ago

codesome commented 16 days ago

This PR merges all the coding work that has been done in `sparsehistogram` branch over the last 1 year into main branch. This PR is open for review, and in the mean time while this is up for review, we will keep working on the `sparsehistogram` branch to close some P1 histogram issues.

Design doc on native histograms:
https://docs.google.com/document/d/1cLNv3aufPZb3fNfaJgdaRBZsInZKKIHo9E6HinJVbpM/edit
Some sneak peak: https://www.youtube.com/watch?v=T2GvcYNth9U

We would like this PR to be *not* squashed or rebased when merging.

Thanks!