

Network Monitoring and Management

Prometheus



INNOG 6

March 22 - 25, 2023

What is Prometheus?

- Prometheus is an open-source monitoring and alerting toolkit
- Built on Go, by Soundcloud
- It collects and stores its metrics as time series data, i.e. metrics information is stored with the timestamp at which it was recorded, alongside optional key-value pairs called labels.

Features of Prometheus?

- a multi-dimensional data model with time series data identified by metric name and key/value pairs
- PromQL, a flexible query language to leverage this dimensionality
- no reliance on distributed storage; single server nodes are autonomous
- time series collection happens via a pull model over HTTP
- pushing time series is supported via an intermediary gateway
- targets are discovered via service discovery or static configuration
- multiple modes of graphing and dashboarding support

What are metrics?

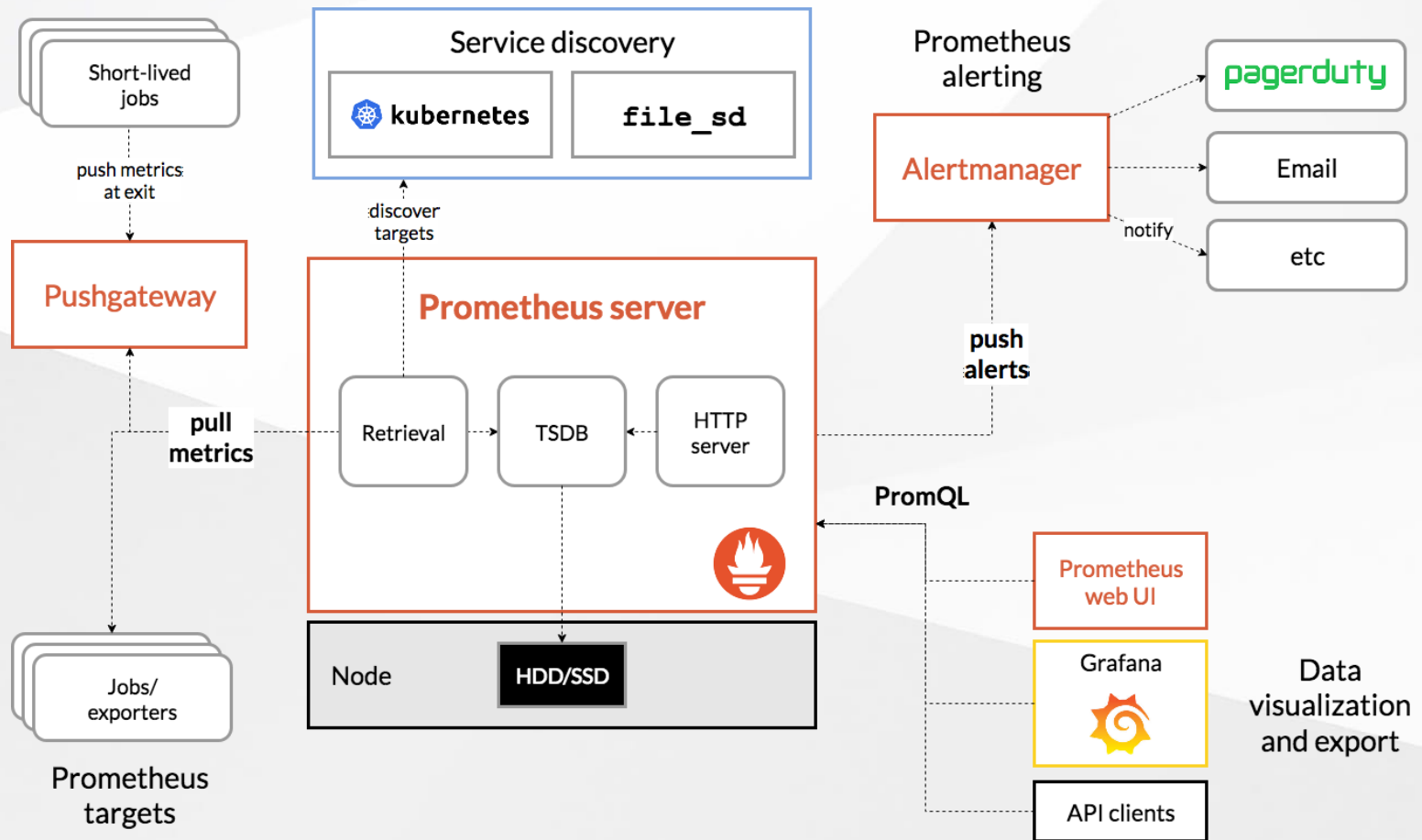
- In layperson terms, metrics are numeric measurements.
- Time series means that changes are recorded over time.
- Metrics play an important role in understanding why your application is working in a certain way.

Components

The Prometheus ecosystem consists of multiple components, many of which are optional:

- the main Prometheus server which scrapes and stores time series data
- client libraries for instrumenting application code
- a push gateway for supporting short-lived jobs
- special-purpose exporters for services like HAProxy, StatsD, Graphite, etc.
- an alertmanager to handle alerts
- various support tools

Architecture








Metric Types

- Counter
 - Int value that only increases or can be reset to 0
- Gauge
 - single numerical value that can arbitrarily go up or down
- Histogram
 - a histogram samples observations and counts them in configurable buckets. It also provides a multiple time series during scrape like sum of all observed valuesm count of events
- Summary
 - a summary samples observation (usually thihngs like request durations and response size) over a sliding time window

Exporters


- Exporters expose metrics with an HTTP API
- Bindings available for many languages
- Exporters do not save data
- Exporters are not "proxies" and don't "cache" anything

Exploring Metrics


 Prometheus

☐ Use local time ☐ Enable query history ☒ Enable autocomplete ☒ Enable highlighting

☒ Enable linter



Expression (press Shift+Enter for newlines)



Execute

Table

Graph

<

Evaluation time

>

No data queried yet

Remove Panel

Add Panel

Exploring Metrics (contd.)

The screenshot shows the Prometheus web interface. At the top is a dark header with the Prometheus logo and name, and settings icons. Below the header are several checkboxes: 'Use local time' (unchecked), 'Enable query history' (unchecked), 'Enable autocomplete' (checked), 'Enable highlighting' (checked), and 'Enable linter' (checked). The main area features a search bar with the text 'mkt'. A dropdown menu is open, displaying a list of metrics starting with 'mkt', each preceded by a cube icon. The metrics listed are: 'mktxp_collection_time_total', 'mktxp_interface_full_duplex', 'mktxp_interface_rate', 'mktxp_interface_rx_byte_total', 'mktxp_interface_rx_drop_total', 'mktxp_interface_rx_error_total', 'mktxp_interface_rx_packet_total', 'mktxp_interface_sfp_temperature', 'mktxp_interface_status', 'mktxp_interface_tx_byte_total', 'mktxp_interface_tx_drop_total', 'mktxp_interface_tx_error_total', 'mktxp_interface_tx_packet_total', 'mktxp_internet_bandwidth', and 'mktxp_internet_latency'. To the left of the dropdown, there are buttons for 'Table', '<', and 'No d'. To the right of the search bar is an 'Execute' button. Below the dropdown, there is an 'Add P' button. On the right side of the interface, there is a 'Remove Panel' link and some partially visible text 'gauge' and 'gauge'.

Prometheus

☐ Use local time ☐ Enable query history ☒ Enable autocomplete ☒ Enable highlighting

☒ Enable linter

Search: mkt

Execute

Table

<

No d

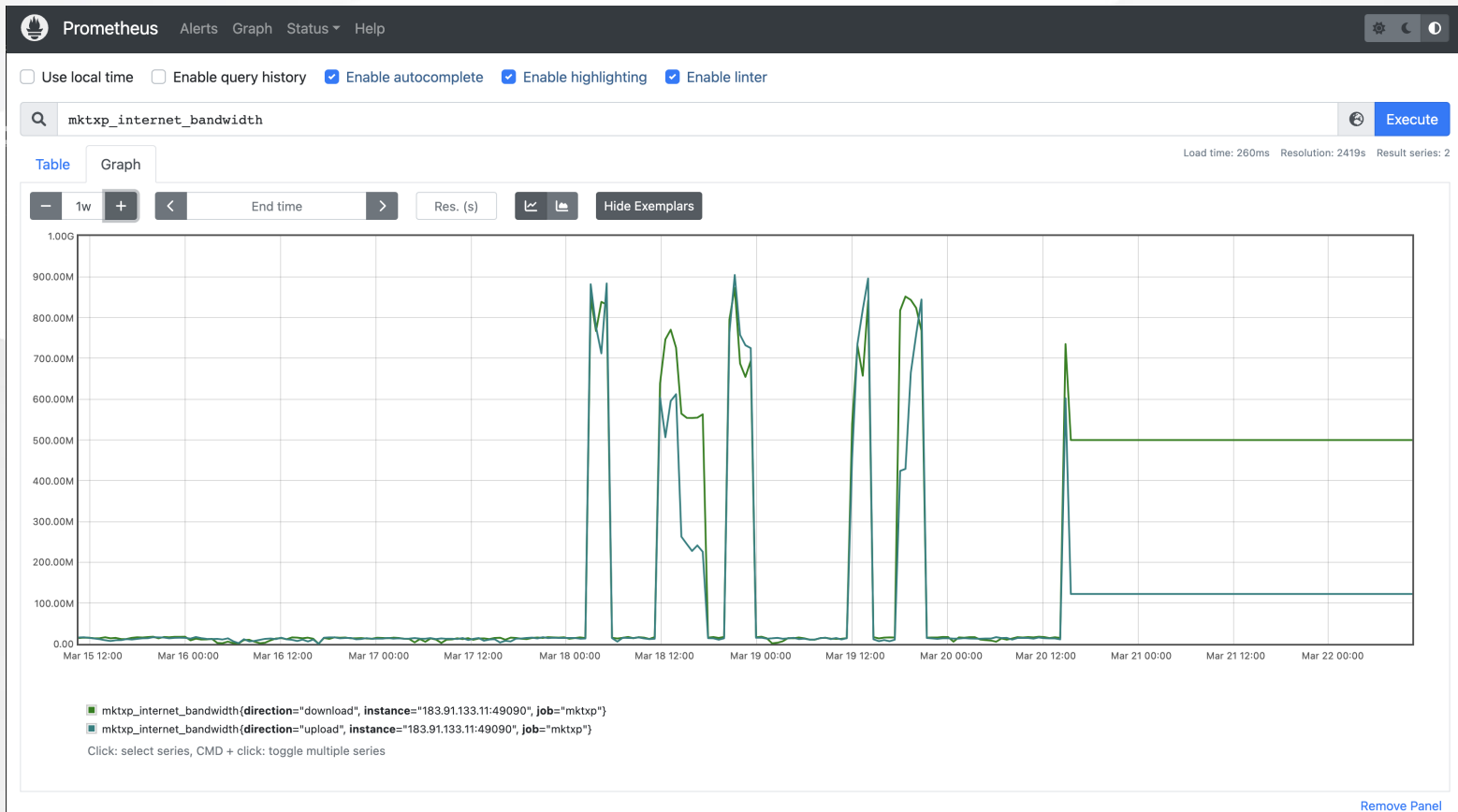
Add P

Remove Panel

gauge

gauge

Exploring Metrics (contd..)



PromQL

Prometheus provides a functional query language called PromQL (Prometheus Query Language)

- PromQL lets the user select and aggregate time series data in real time
- The result of an expression can either be shown as a graph, viewed as tabular data in Prometheus's expression browser, or consumed by external systems via the HTTP API.

Alerting

Alerting with Prometheus is separated into two parts.

- Alerting rules in Prometheus servers send alerts to an Alertmanager.
- The Alertmanager then manages those alerts, including silencing, inhibition, aggregation and sending out notifications via methods such as email, on-call notification systems, and chat platforms.

