Project Analysis

Prometheus

Introduction

Prometheus is an open-source systems monitoring and alerting toolkit originally built at SoundCloud. Since its inception in 2012, many companies and organizations have adopted Prometheus, and the project has a very active developer and user community.

Prometheus was developed at SoundCloud starting in 2012, when the company discovered that its existing metrics and monitoring solutions (using StatsD and Graphite) were not sufficient for their needs. Specifically, they identified needs that Prometheus was built to meet including: a multi-dimensional data model, operational simplicity, scalable data collection, and a powerful query language, all in a single tool. The project was open-source from the beginning and began to be used by Boxever and Docker users as well, despite not being explicitly announced.[8][9] Prometheus was inspired by the monitoring tool Borgmon used at Google. By 2013, Prometheus was introduced for production monitoring at SoundCloud. The official public announcement was made in January 2015

Project Summary

Website	https://prometheus.io/
Organization/Foundation Name	Soundcloud
License	Apache License 2.0
Open/Proprietary	Open
Source Path(if open source)	https://github.com/prometheus
Brief Description	Prometheus 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.

Project Details

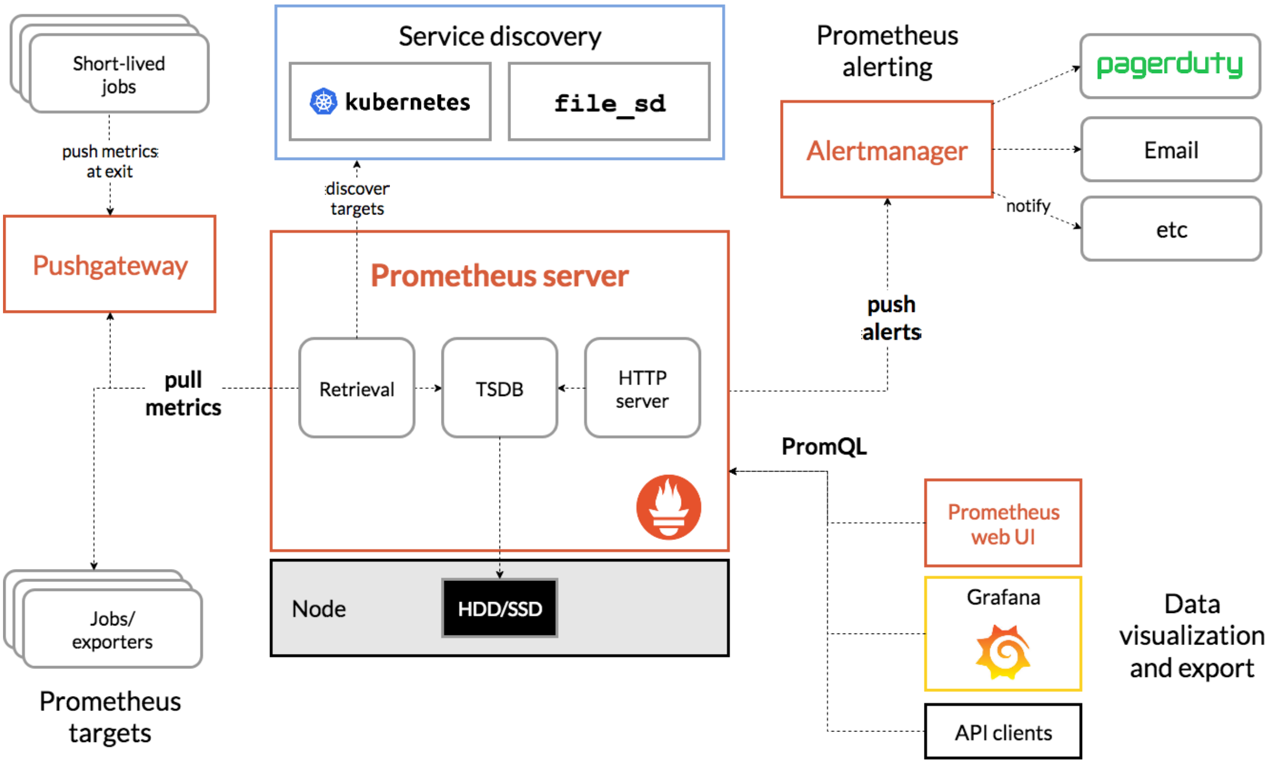
Key Features

Prometheus's main features are:

- 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

Architecture



Prometheus scrapes metrics from instrumented jobs, either directly or via an intermediary push gateway for short-lived jobs. It stores all scraped samples locally and runs rules over this data to either aggregate and record new time series from existing data or generate alerts. Grafana or other API consumers can be used to visualize the collected data.

Current Usage

Prometheus was first used in-house at SoundCloud, where it was developed, for monitoring their systems. The Cloud Native Computing Foundation has a number of case studies of other companies using Prometheus. GitLab provides a Prometheus integration guide to export GitLab metrics to Prometheus and it is activated by default since version 9.0. Some of the organizations using this project are

Amadeus

Cesanta

Core OS

DigitalOcean

Giant Swarm

Grafana Labs

Weaveworks

Technical Details

Prometheus data is stored in the form of metrics, with each metric having a name that is used for referencing and querying it. Each metric can be drilled down by an arbitrary number of key=value pairs (labels). Labels can include information on the data source (which server the data is coming from) and other application-specific breakdown information such as the HTTP status code (for metrics related to HTTP responses), query method (GET versus POST), endpoint, etc. The ability to specify an arbitrary list of labels and to query based on these in real time is why Prometheus' data model is called multi-dimensional.

Prometheus collects data in the form of time series. The time series are built through a pull model: the Prometheus server queries a list of data sources (sometimes called exporters) at a specific polling frequency. Each of the data sources serves the current values of the metrics for that data source at the endpoint queried by Prometheus. The Prometheus server then aggregates data across the data sources. Prometheus has a number of mechanisms to automatically discover resources that should be used as data sources.

Configuration for alerts can be specified in Prometheus which specifies a condition that needs to be maintained for a specific duration in order for an alert to trigger. When alerts trigger, they are forwarded to the Alertmanager service. Alertmanager can include logic to silence alerts and also to forward them to email, Slack, or notification services such as PagerDuty. Some other messaging systems like Microsoft Teams could be configured using the Alertmanager Webhook Receiver as a mechanism for external integrations. Also Prometheus Alerts can be used to receive alerts directly on android devices even without the requirement of any targets configuration in Alert Manager.

Project comparison

Prometheus vs Grafana

Grafana is an open-source visualization software, which helps the users to understand the complex data with the help of data metrics.

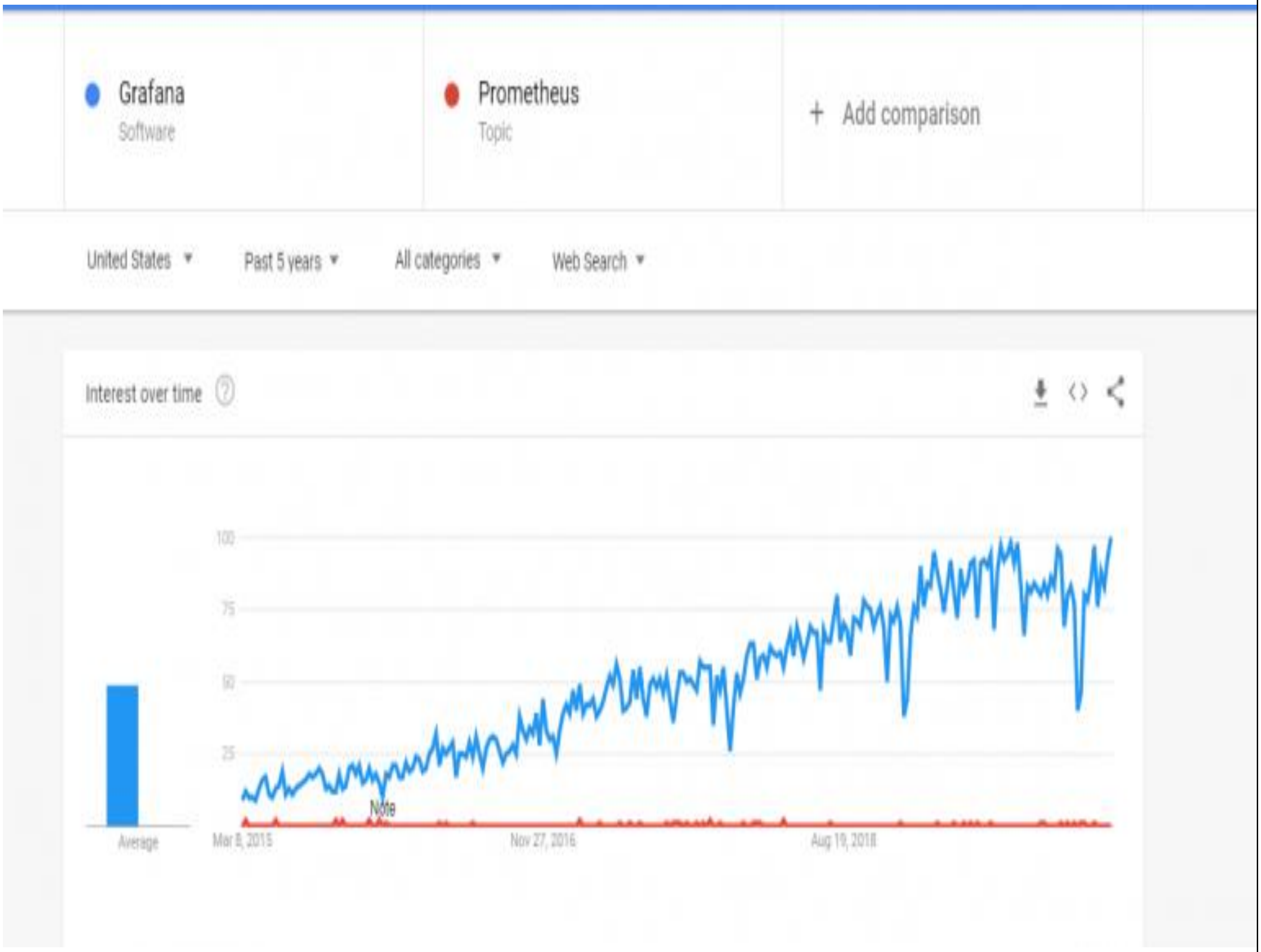
It equips the users with features like alert systems, sharing of the dashboard.

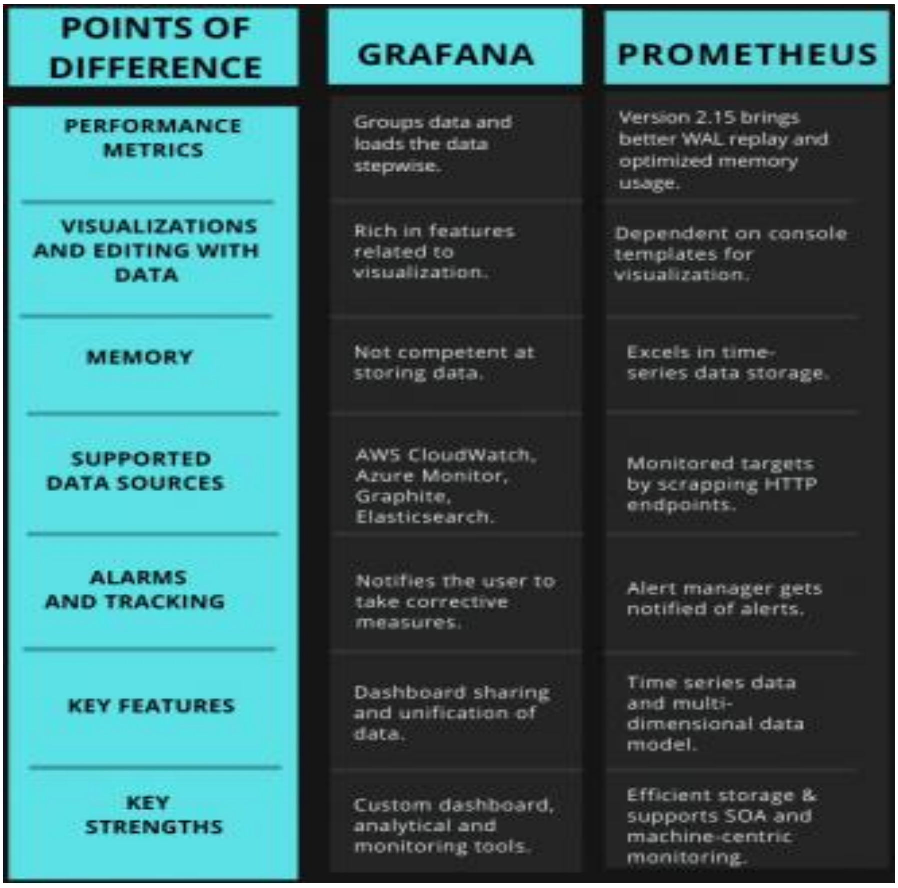
It also helps users in studying time-series analytics.

Prometheus is an open-source event monitoring and alerting tool.

It stores the majority of the data locally after scrapping metrics, and after that, to generate alerts, it runs rules over data.

It came into existence because of the need for monitoring multiple microservices running within the systems.It was initially built at SoundCloud.





POINTS OF DIFFERENCE	GRAFANA	PROMETHEUS
PERFORMANCE METRICS	Groups data and loads the data stepwise.	Version 2.15 brings better WAL replay and optimized memory usage.
VISUALIZATIONS AND EDITING WITH DATA	Rich in features related to visualization.	Dependent on console templates for visualization.
MEMORY	Not competent at storing data.	Excels in time-series data storage.
SUPPORTED DATA SOURCES	AWS CloudWatch, Azure Monitor, Graphite, Elasticsearch.	Monitored targets by scrapping HTTP endpoints.
ALARMS AND TRACKING	Notifies the user to take corrective measures.	Alert manager gets notified of alerts.
KEY FEATURES	Dashboard sharing and unification of data.	Time series data and multi-dimensional data model.
KEY STRENGTHS	Custom dashboard, analytical and monitoring tools.	Efficient storage & supports SOA and machine-centric monitoring.

Prometheus and Grafana are both great observability solutions. However, they were created with different priorities. Prometheus focuses on data acquisition, allowing users to select and aggregate time series data in real time. Grafana, on the other hand, focuses on data visualization.

Reference / Acknowledgements

<https://prometheus.io/->:official website

< https://github.com/prometheus/prometheus> : Github Repository

https://betterstack.com/community/comparisons/prometheus-vs-grafana/ - Blog on prometheus