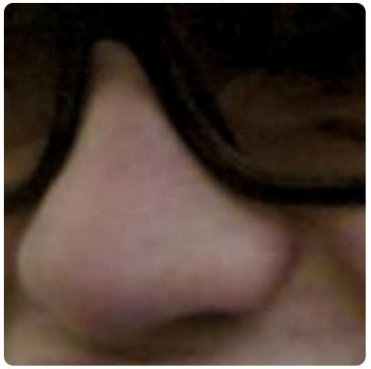
AIOps: Anomaly detection with Prometheus

Spice up your Monitoring with AI

Marcel Hild

Principal Software Engineer @ Red Hat AI CoE / Office of the CTO



Marcel Hild

durandom

old school opensource hacker and daemon zombie slayer at **@b4mad** and Red Hat's **@AICoE** CTO Office

Red Hat

Kiel, Germany

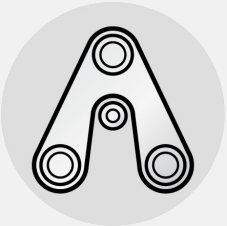
hild@b4mad.net

http://durandom.de

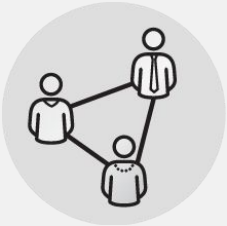
Organizations

HOW RED HAT SEES AI

Represents a workload requirement for our **platforms** across the hybrid cloud.



Applicable to Red Hat's existing core business in order to increase **Open Source** development and production **efficiency**.



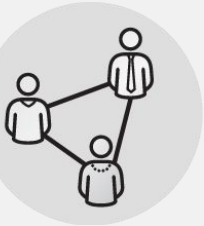
Valuable to our customers as specific services and product capabilities, providing an **Intelligent Platform** experience.

Enable customers to build **Intelligent Apps** using Red Hat products as well as our broader partner ecosystem.

Data as the Foundation

HOW RED HAT SEES AI

Project Thoth and Bots
http://bit.ly/2zYfb6h



Represents a workload requirement for our **platforms** across the hybrid cloud.

Applicable to Red Hat's existing core business in order to increase **Open Source** development and production **efficiency**.

Valuable to our customers as specific services and product capabilities, providing an **Intelligent Platform** experience.

Enable customers to build **Intelligent Apps** using Red Hat products as well as our broader partner ecosystem.

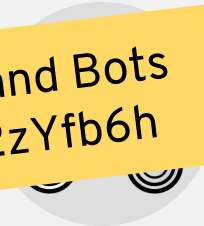
010110
101010

Data as the Foundation

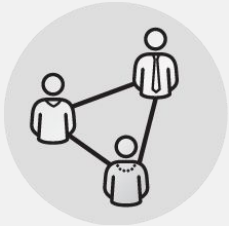
HOW RED HAT SEES AI

Project Thoth and Bots
http://bit.ly/2zYfb6h

Represents a workload requirement for our **platforms** across the hybrid cloud.



Applicable to Red Hat's existing core business in order to increase **Open Source** development and production **efficiency**.



Valuable to our customers as specific services and product capabilities, providing an **Intelligent Platform** experience.

Enable customers to build **Intelligent Apps** using Red Hat products as well as our broader partner ecosystem.

OpenDataHub
http://bit.ly/2y6Nh6m

Data as the Foundation

HOW RED HAT SEES AI

Project Thoth and Bots
http://bit.ly/2zYfb6h

This Talk

Represents a workload requirement for our **platforms** across the hybrid cloud.

Applicable to Red Hat's existing core business in order to increase **Open Source** development and production **efficiency**.

Valuable to our customers as specific services and product capabilities, providing an **Intelligent Platform** experience.

Enable customers to build **Intelligent Apps** using Red Hat products as well as our broader partner ecosystem.

OpenDataHub
http://bit.ly/2y6Nh6m

Data as the Foundation

Overview

Prometheus

Long term storage

Atonomy of an Anømål¥

Integration into monitoring setup

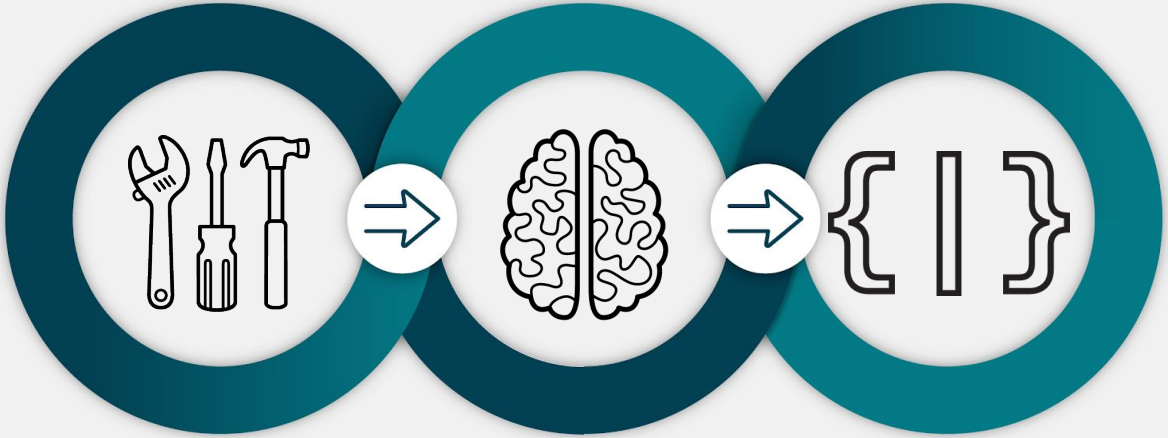
What's **not** in this talk

shiny product and the holy grail of monitoring

ready solution to turn your monitoring setup into spider demon

success story how we turned our messy monitoring into an advance ai monitoring

What **is** in this talk



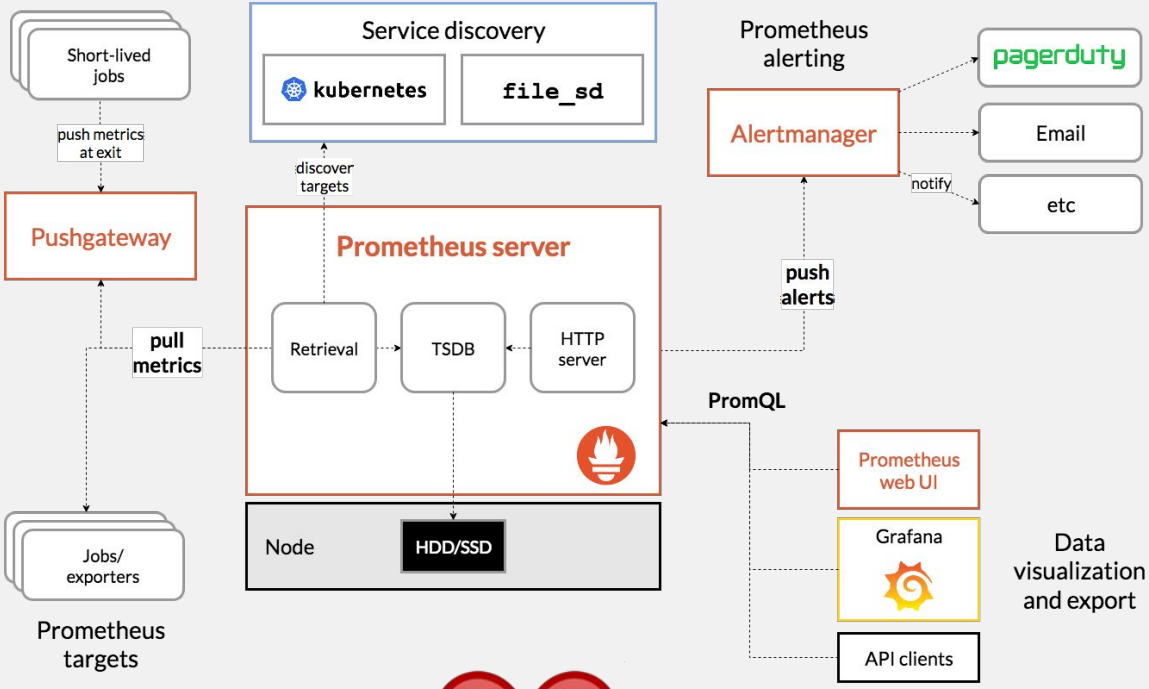
tools and scripts
to get you started

Q&A to problems

all OSS

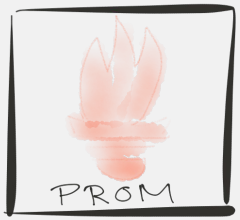
What is prometheus?

Prometheus architecture



Everybody  architecture slides

Prometheus architecture

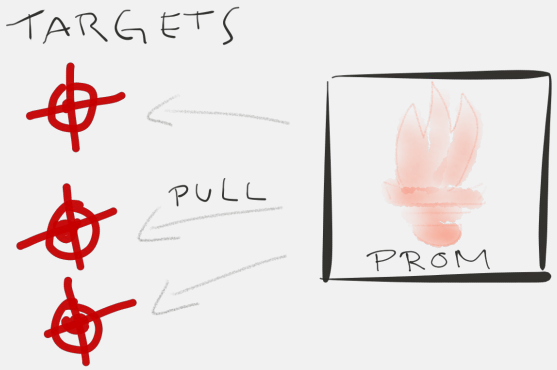


Simplistic world view

Prometheus architecture

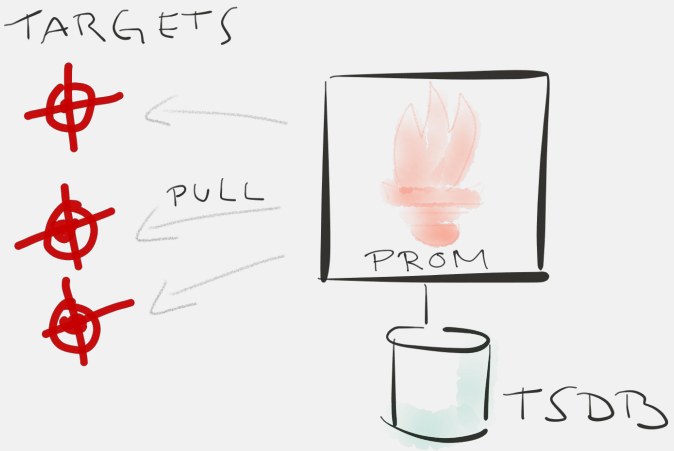
Simplistic world view

Prometheus architecture



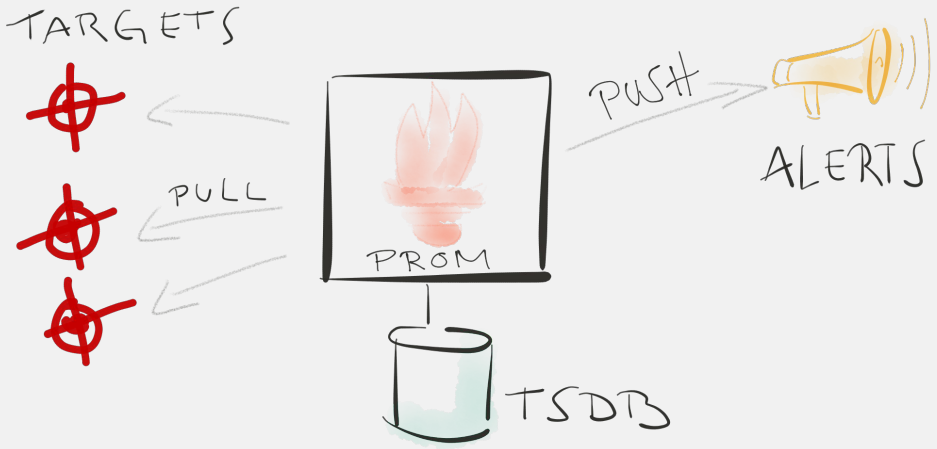
Simplistic world view

Prometheus architecture



Simplistic world view

Prometheus architecture



Simplistic world view

Prometheus is made for

What do we need for machine learning?
---> DATA DATA DATA

Long term storage of Prometheus data

Too good to be true...

- Prometheus at scale
- Global query view
- Reliable historical data storage
- Unlimited retention
- Downsampling

thanos is in the making,
but until then?

Works great, but...

gh/AICoE/p-influx
http://bit.ly/2y6CvwX

- easily hooked into prometheus with write and read endpoint
- Reliable historical data storage
- Great for data science
  - Pandas integration

Eats RAM for breakfast

Let’s just store it...

prometheus scraper

- container can be configured to scrape any prometheus server
- can scrape all or a subset of the metrics
- stores data in ceph or S3 compliant storage
- can be queried with spark sql
- Future Proof: path to Thanos

gh/AICoE/p-lts
http://bit.ly/2Qw9pho

Harness the power of spark to

- Query stored JSON files
- Distribute the workload
- Use spark library

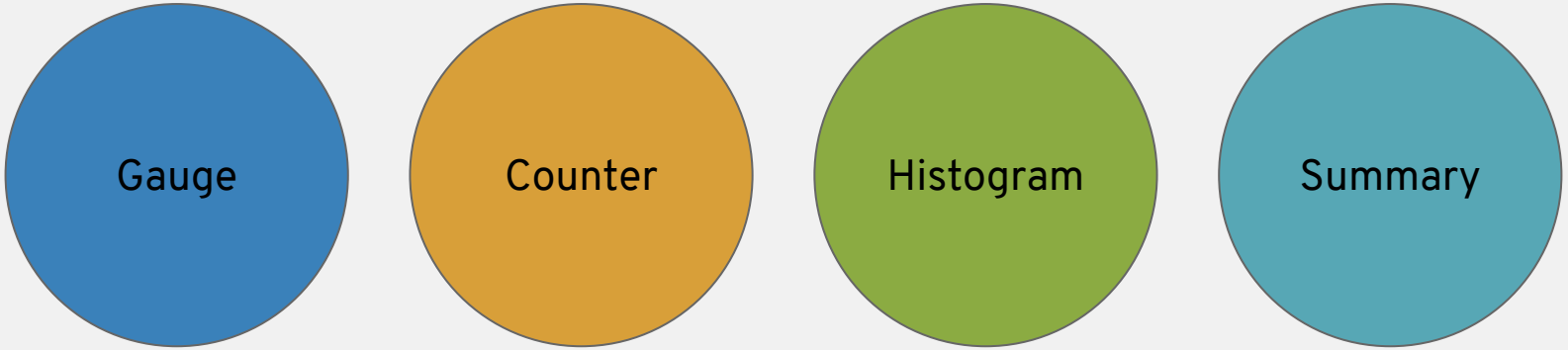
notebook
http://bit.ly/2PIZZVG

```
def get_stats(df):
    # calculate mean
    mean = df.agg(F.avg(F.col("values"))).head()[0]

    # calculate variance
    var = df.agg(F.variance(F.col("values"))).head()[0]

    # calculate standard deviation
    stddev = df.agg(F.stddev(F.col("values"))).head()[0]

    # calculate median
    median = float(df.approxQuantile("values", [0.5], 0.25)[0])

    return mean, var, stddev, median

mean, var, stddev, median = get_stats(data)

print("\tMean(values): ", mean)
print("\tVariance(values): ", var)
print("\tStddev(values): ", stddev)
print("\tMedian(values): ", median)
```

```
        Mean(values):  67087.9063346175
        Variance(values):  56691431555.4375
        Stddev(values):  238099.62527361838
        Median(values):  628.0
```

What do we need for machine learning?
---> CONSISTENT DATA

Prometheus Metric Types



A Time Series

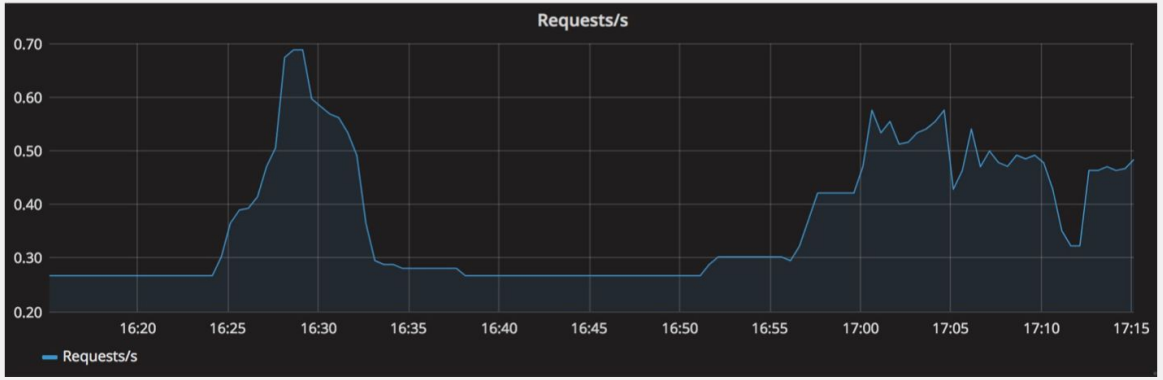
Monotonically Increasing

Cumulative Histogram of Values

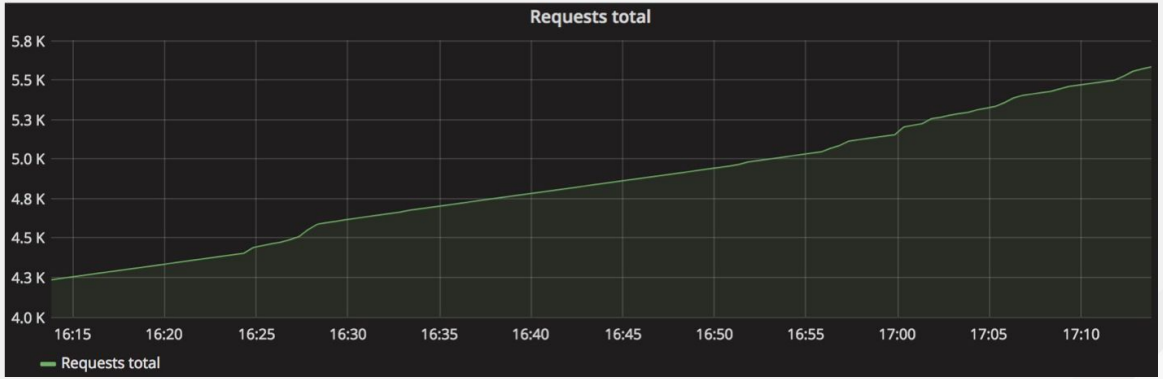
Snapshot of Values in a Time Window

Prometheus Metric Types

Gauge



Counter



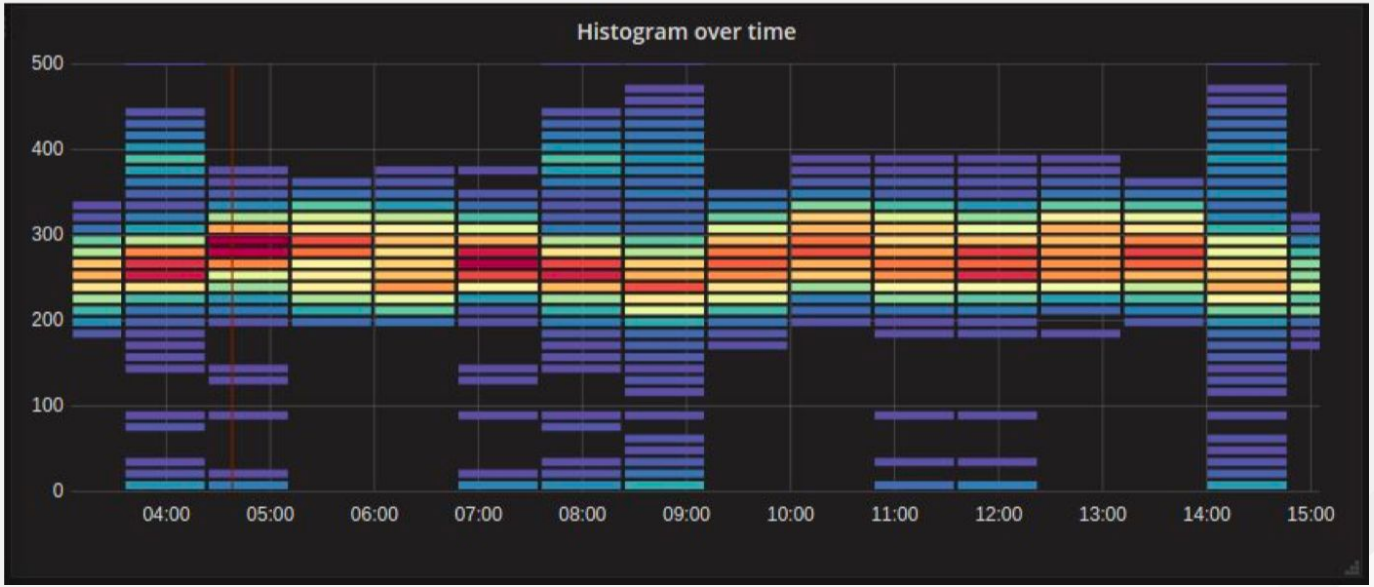
Prometheus Metric Types

Histogram

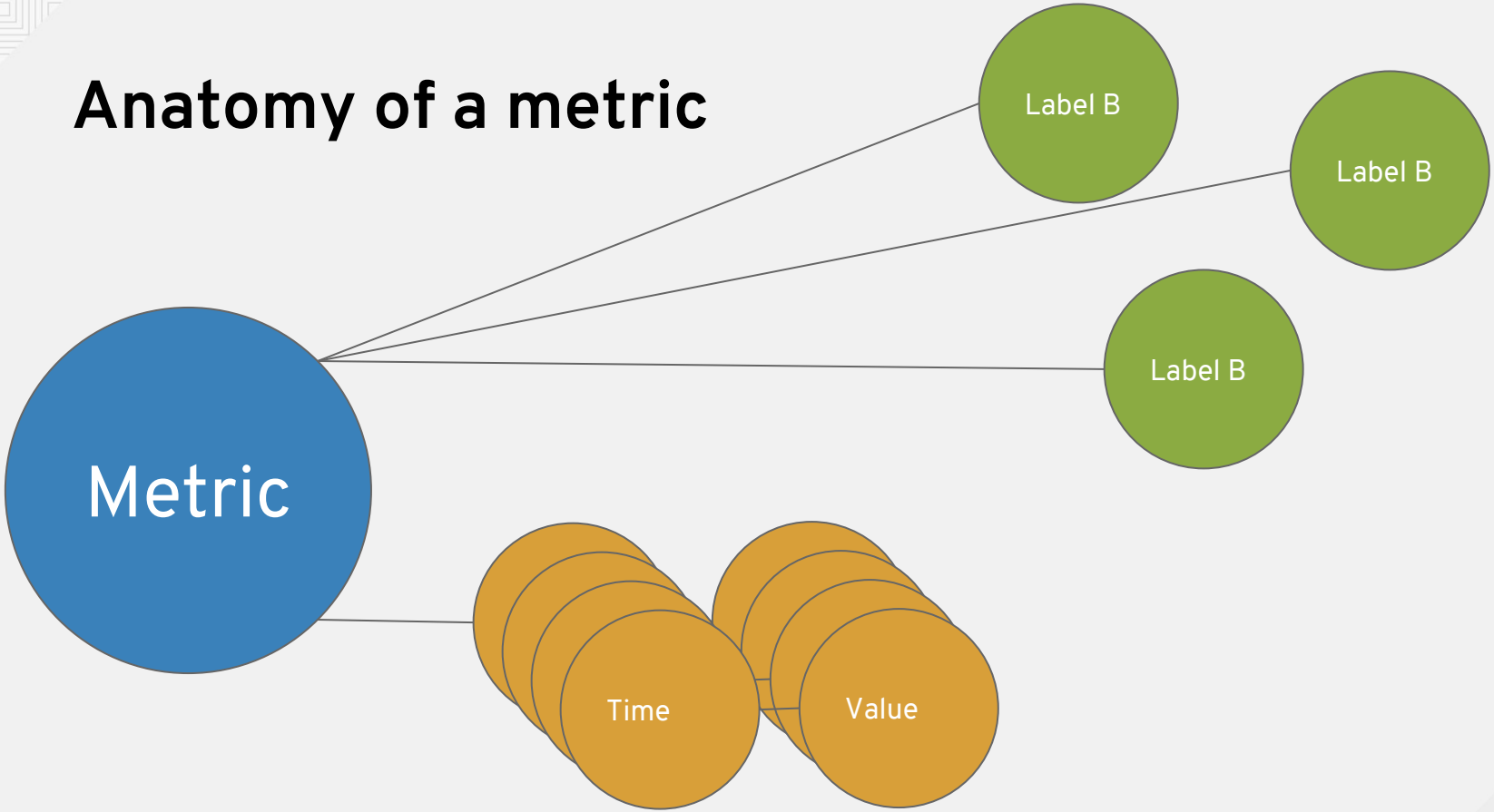
Cumulative

Summary

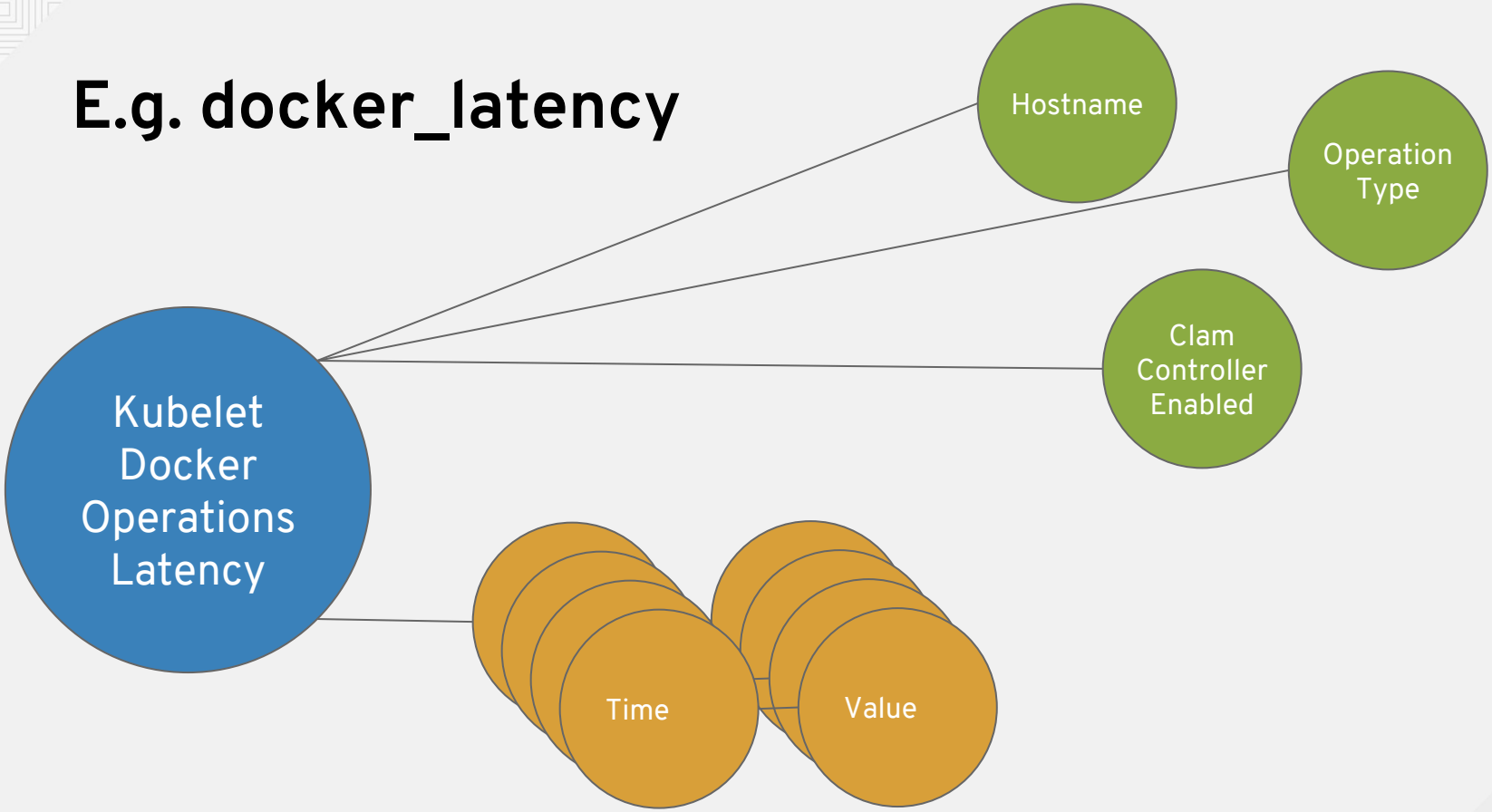
Time Window



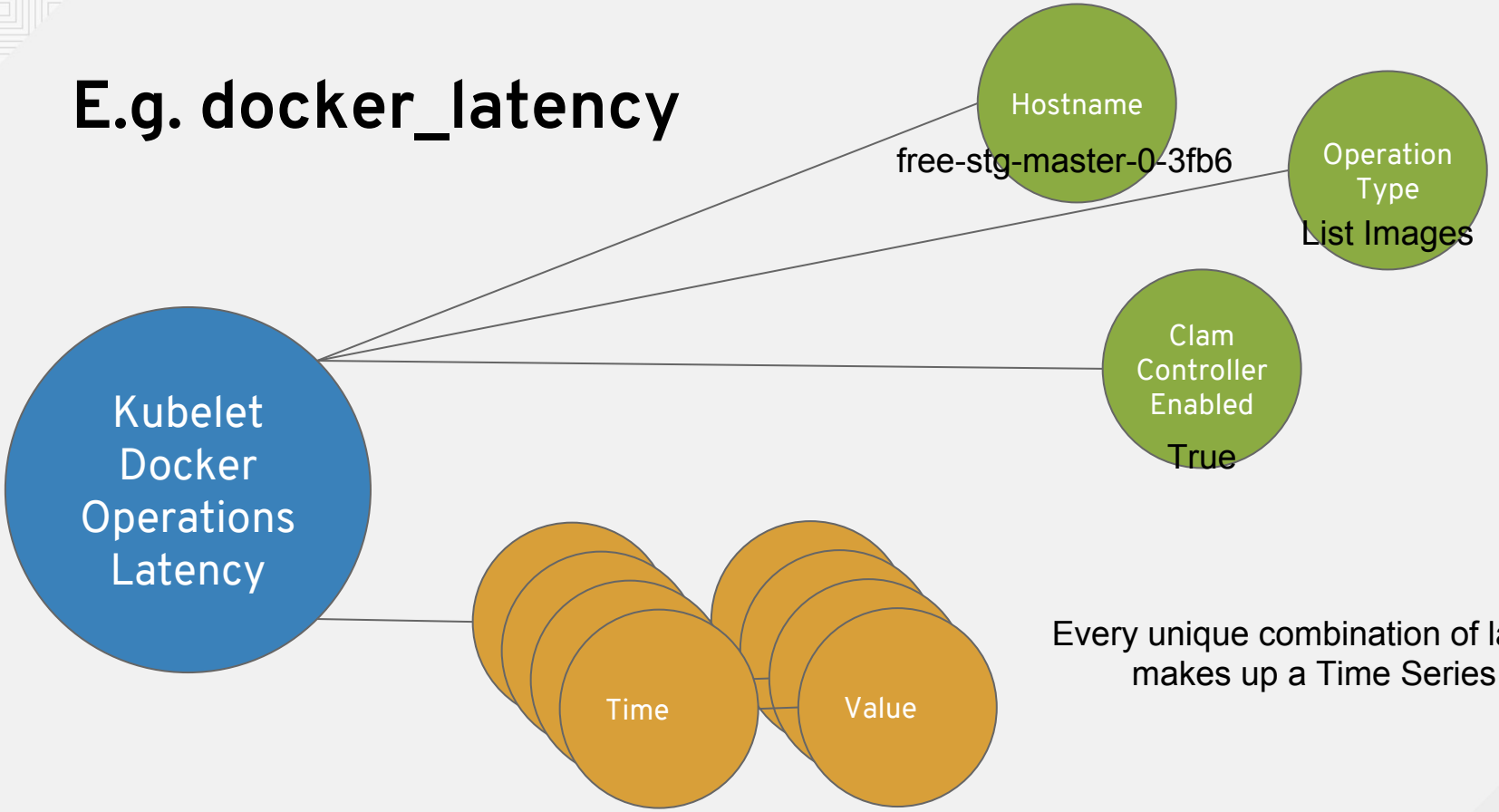
Anatomy of a metric



E.g. docker_latency



E.g. docker_latency



Every unique combination of labels makes up a Time Series

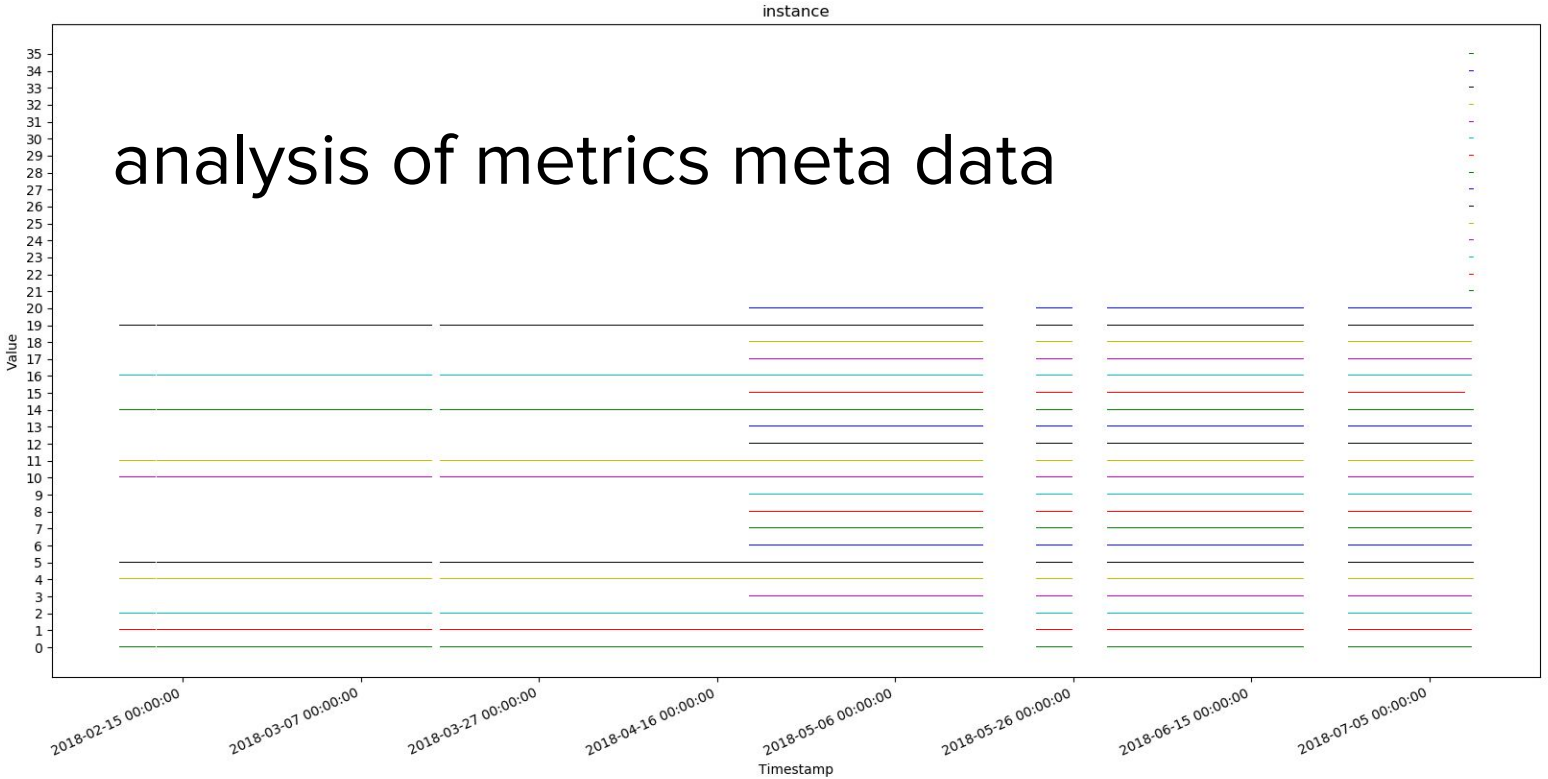
Monitoring is hard

GET /metrics

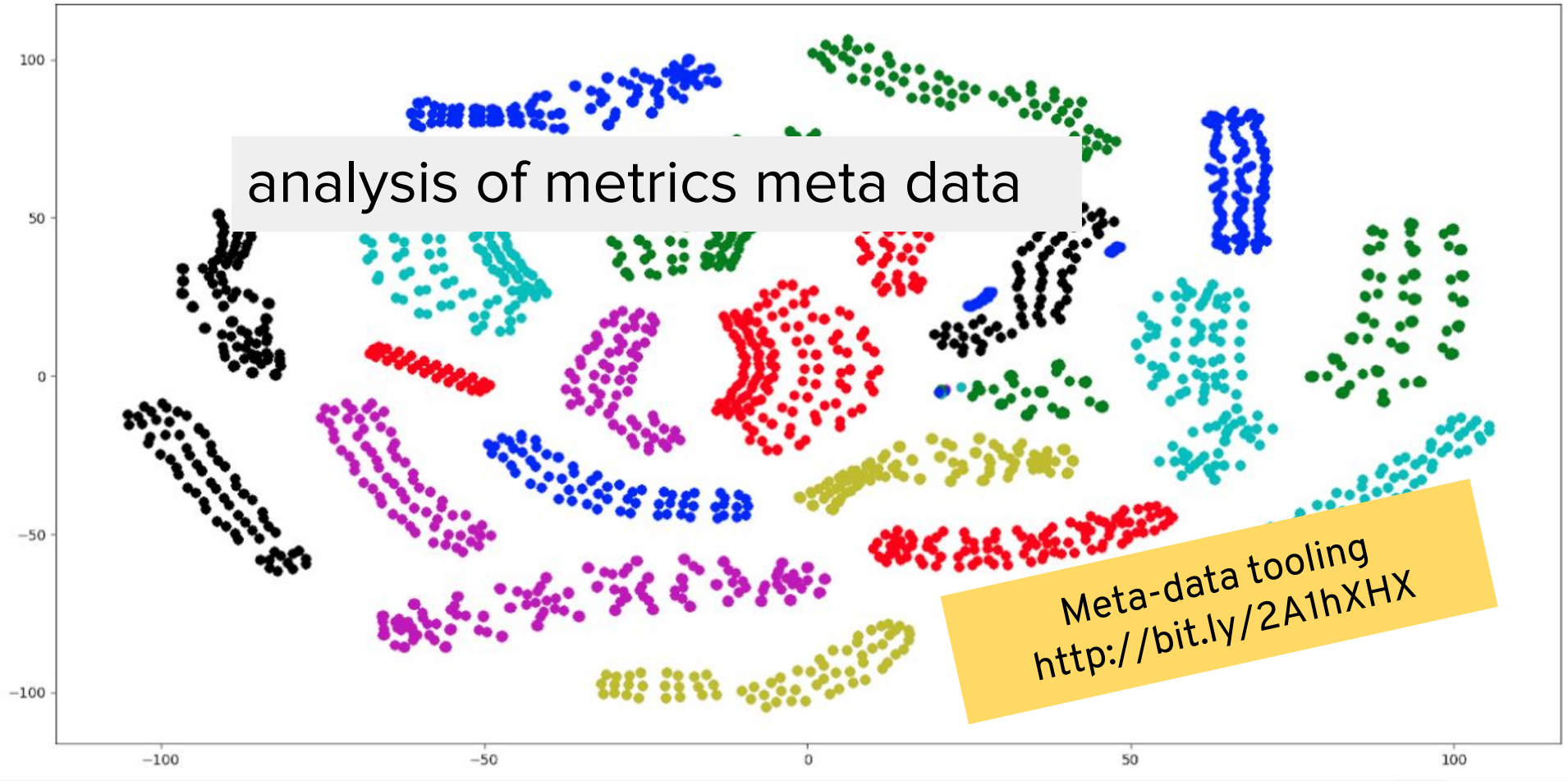
```
# HELP go_gc_duration_seconds A summary of t
# TYPE go_gc_duration_seconds summary
go_gc_duration_seconds{quantile="0"} 9.7014e
go_gc_duration_seconds{quantile="0.25"} 0.00
go_gc_duration_seconds{quantile="0.5"} 0.000
go_gc_duration_seconds{quantile="0.75"} 0.00
go_gc_duration_seconds{quantile="1"} 0.10290
go_gc_duration_seconds_sum 0.239829369
go_gc_duration_seconds_count 196
# HELP go_goroutines Number of goroutines th
# TYPE go_goroutines gauge
go_goroutines 144
# HELP go_memstats_alloc_bytes Number of byt
# TYPE go_memstats_alloc_bytes gauge
go_memstats_alloc_bytes 4.5694928e+07
# HELP go_memstats_alloc_bytes_total Total n
# TYPE go_memstats_alloc_bytes_total counter
go_memstats_alloc_bytes_total 4.19435624e+09
```

- prometheus doesn't enforce a schema
  - /metrics can expose anything it wants
  - no control over what is being exposed by endpoints or targets
  - it can change if your endpoints change versions
- # of metrics to choose from
  - 1000+ for OpenShift
- State of the Art is Dashboards and Alerting
  - Dashboards and Alerting need domain knowledge
- No tools to explore meta-information in metrics

kubelet_docker_operations_latency_microseconds



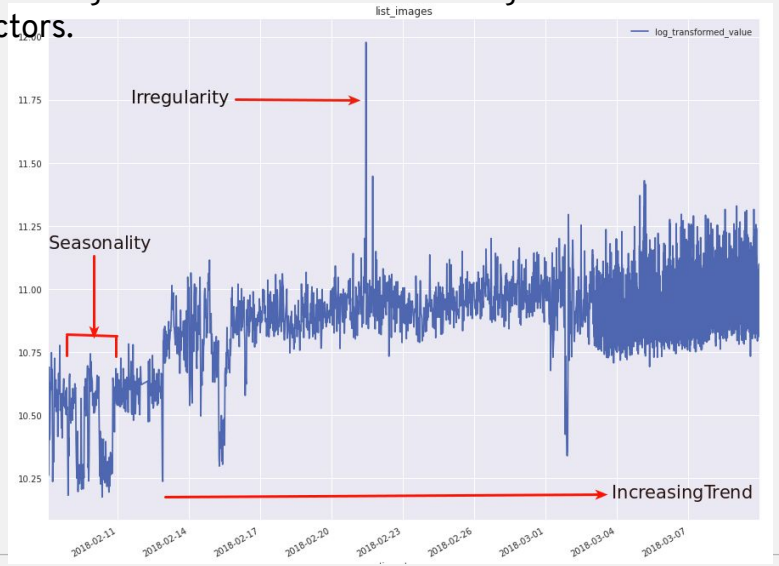
analysis of metrics meta data


analysis of metrics meta data
Meta-data tooling
http://bit.ly/2A1hXHX
100
50
0
−50
−100
−100
−50
0
50
100

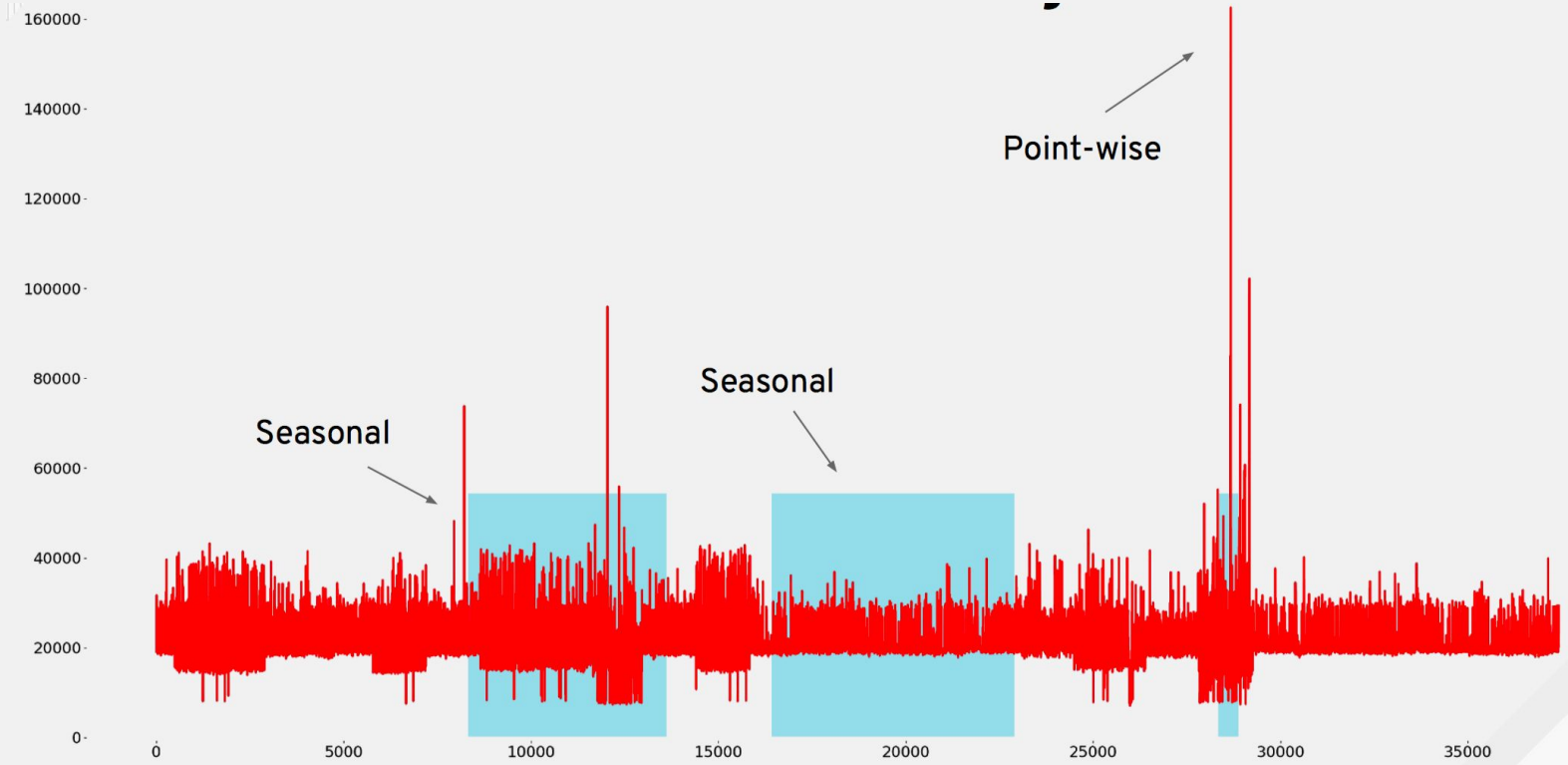
Anomaly Types

Components of Time Series

Trend	Seasonality	Cyclicity	Irregularity
Increase or decrease in the series over a period of time.	Regular pattern of up and down fluctuations. It is a short-term variation occurring due to seasonal factors.	It is a medium-term variation caused by circumstances, which repeat in irregular intervals.	It refers to variations which occur due to unpredictable factors and also do not repeat in particular patterns.

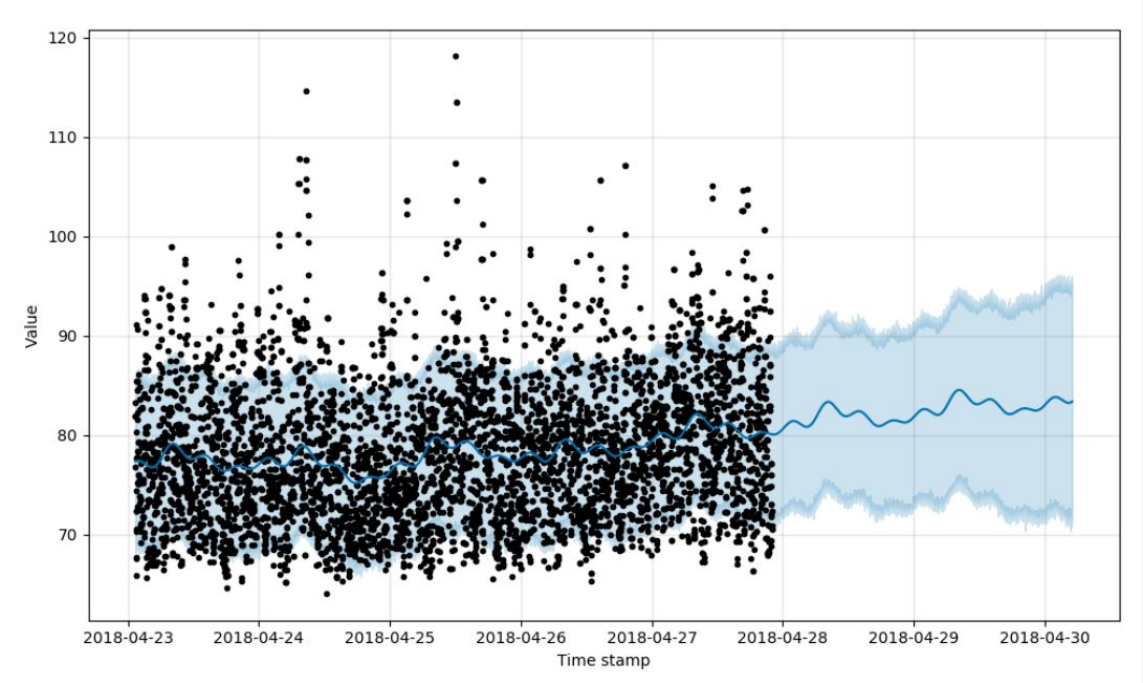


Anomaly Types



Anomaly Detection with Prophet

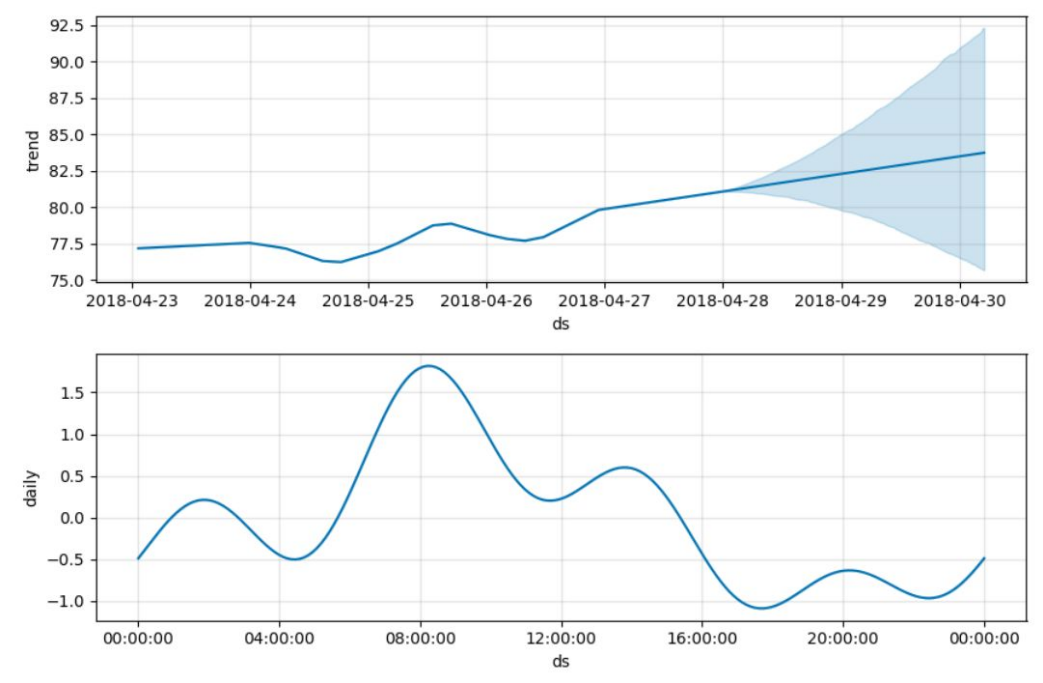
Predicting future data and dynamic thresholds



- `list_images` operation
- on OpenShift
- monitored by prometheus
- detecting outliers
- upper and lower bands

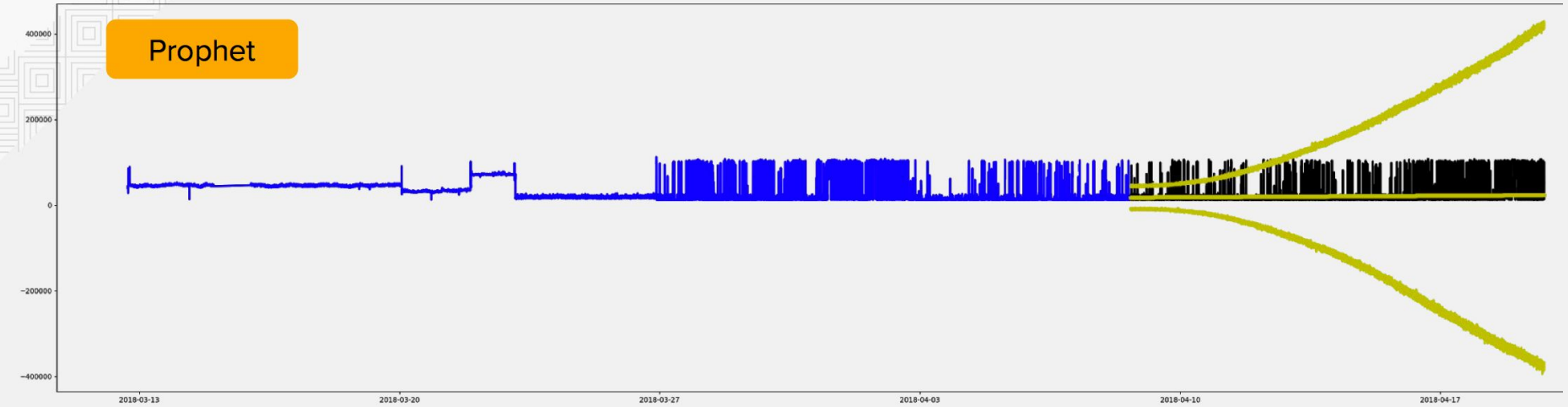
Anomaly Detection with Prophet

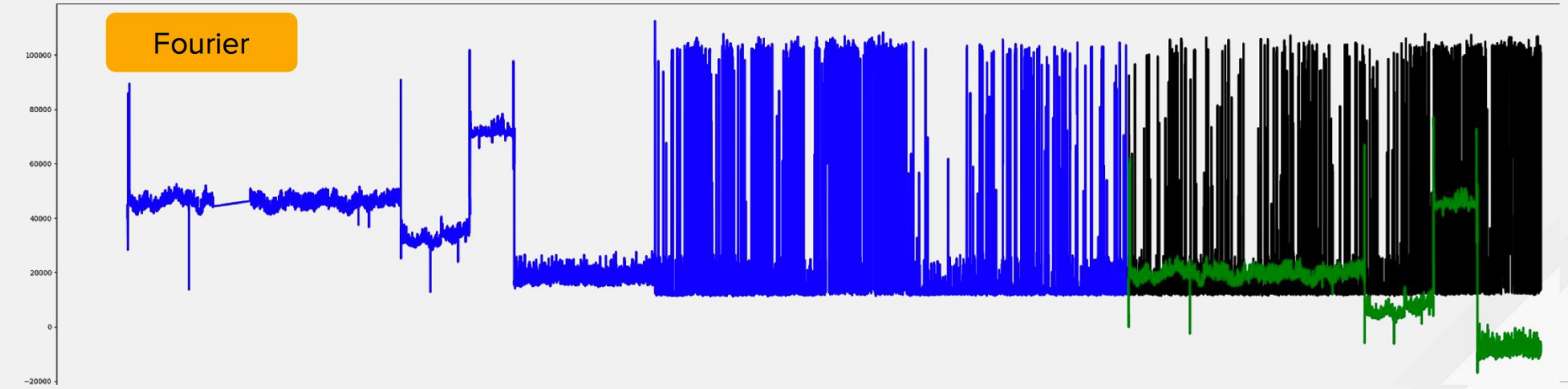
Extracting trends and seasonality



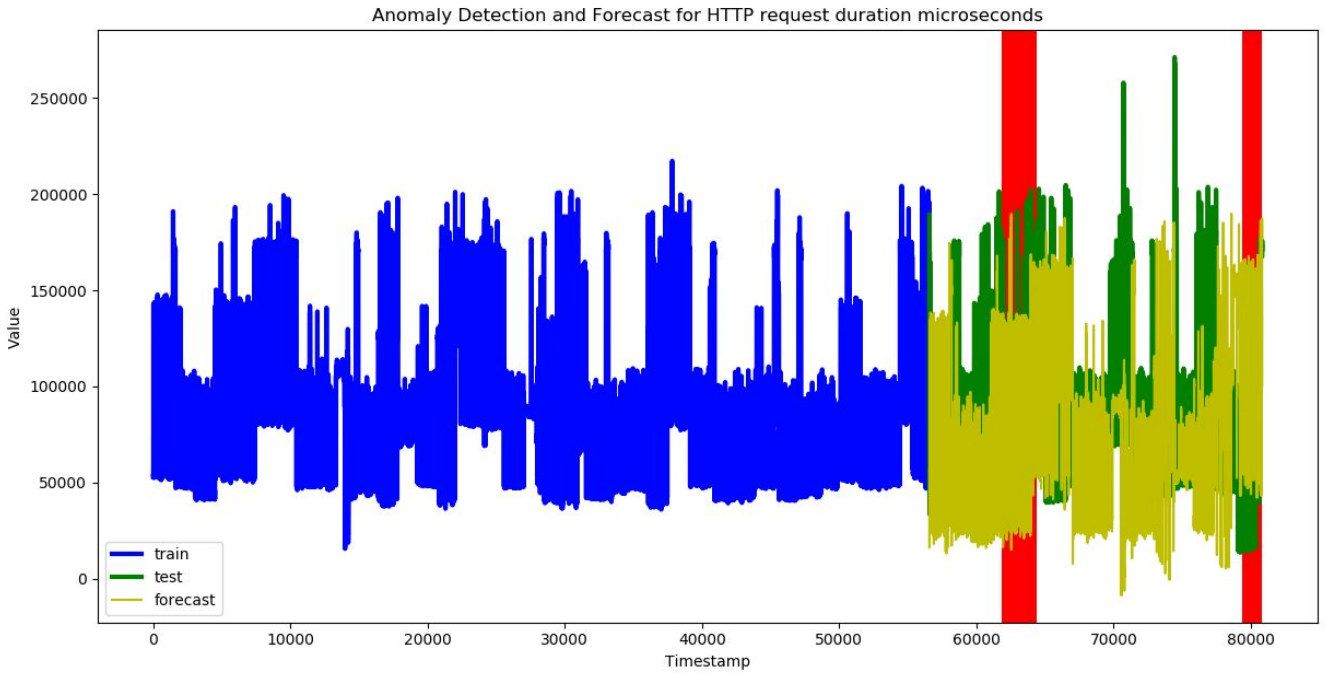
- `list_images` operation
- on OpenShift
- monitored by prometheus
- upward trends
- intraday seasonality

CoE/prophet
http://bit.ly/2pLzGNj

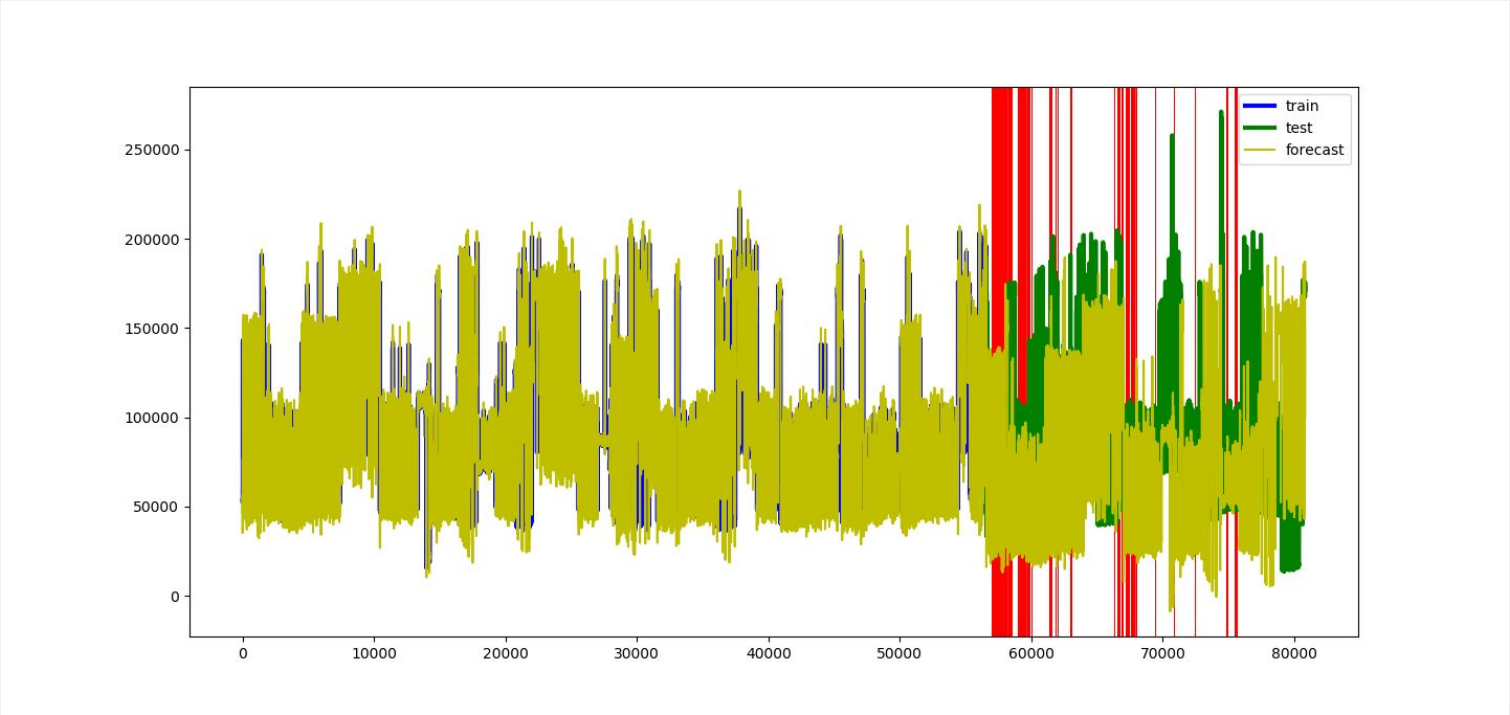

Prophet
400000
200000
0
−200000
−400000
2018-03-13
2018-03-20
2018-03-27
2018-04-03
2018-04-10
2018-04-17


Fourier
100000
80000
60000
40000
20000
0
−20000

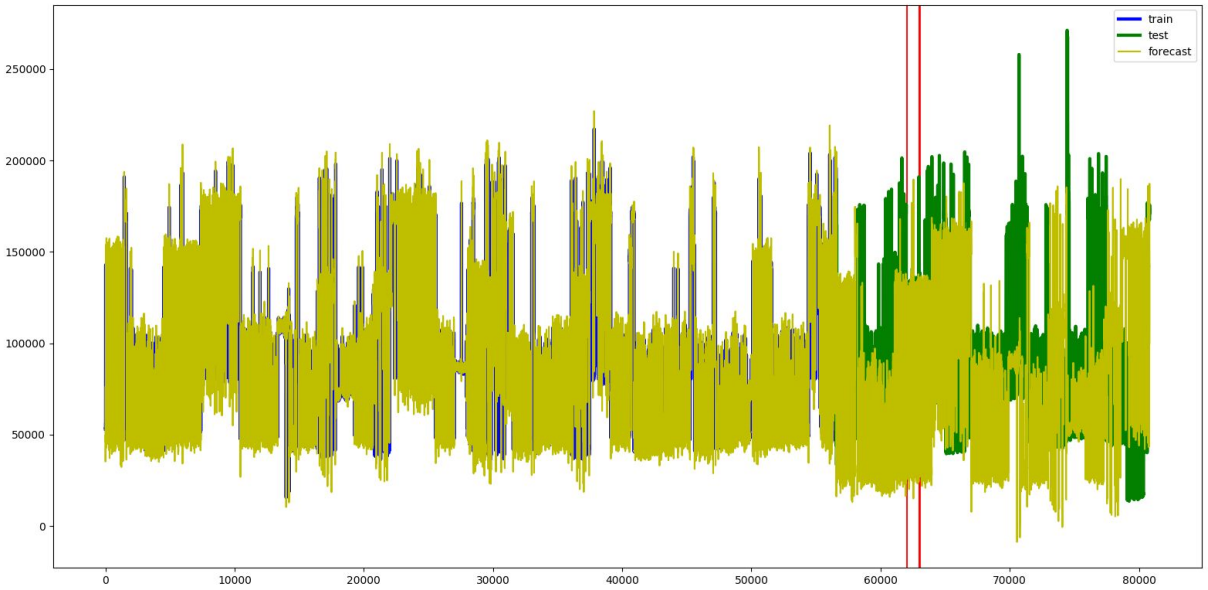
The Accumulator



The Tail Probability



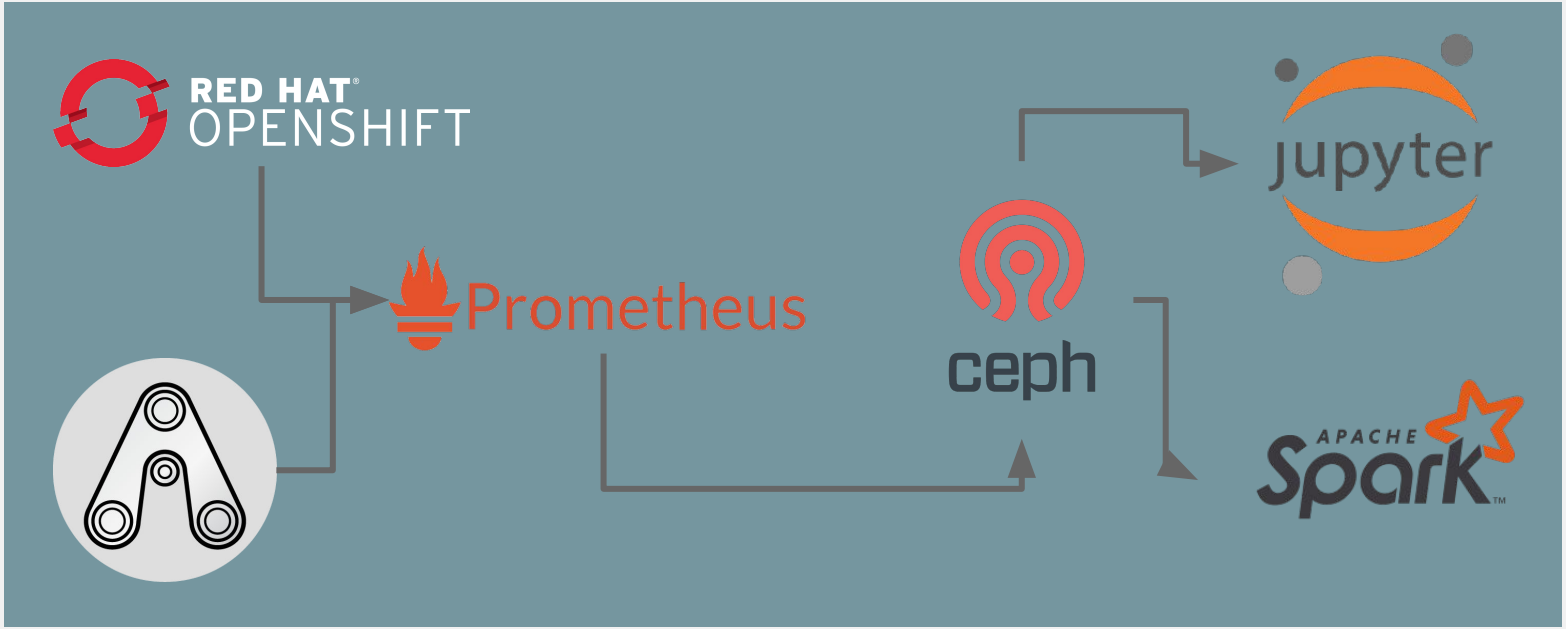
Combined



architecture setup so far

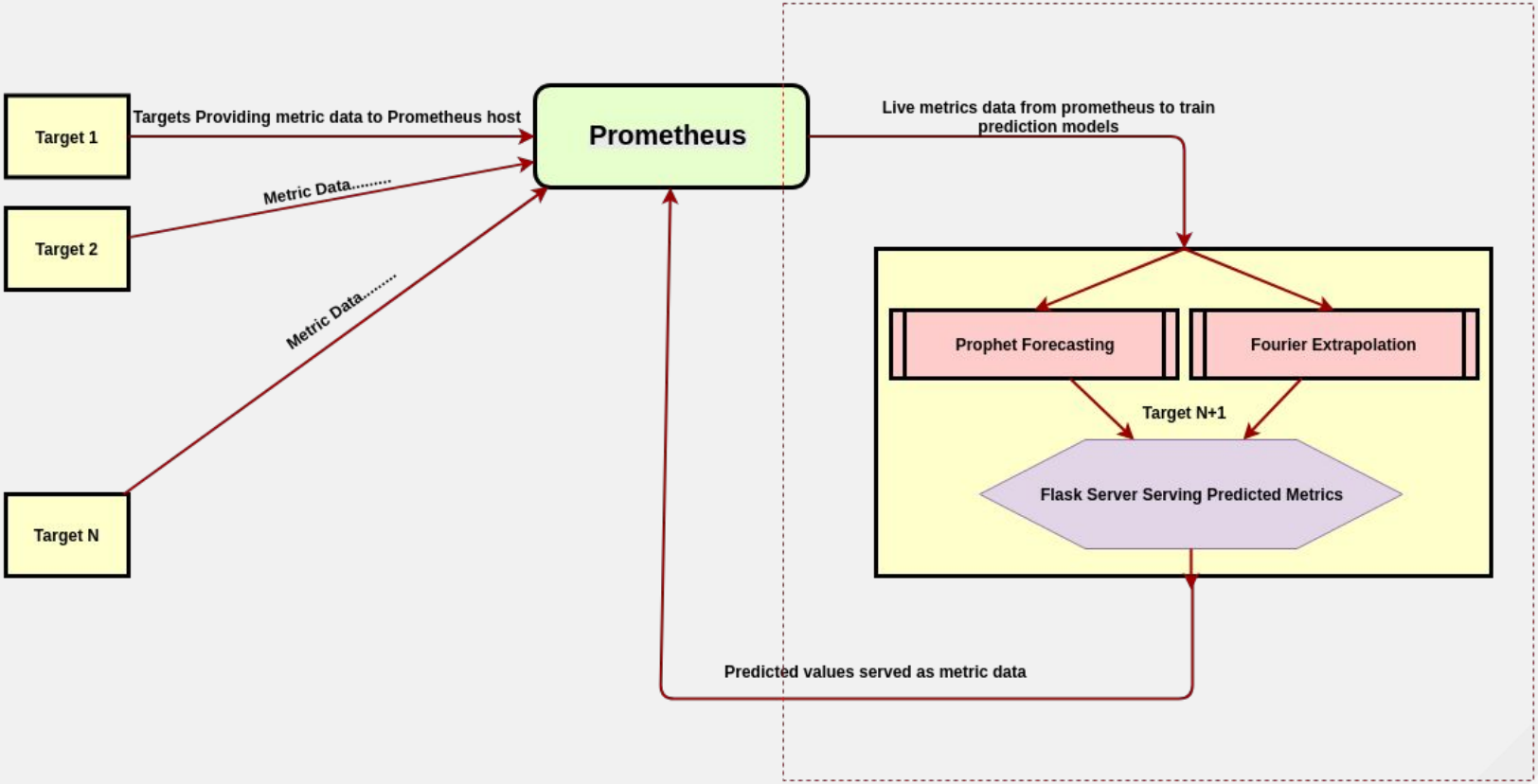
Research Setup

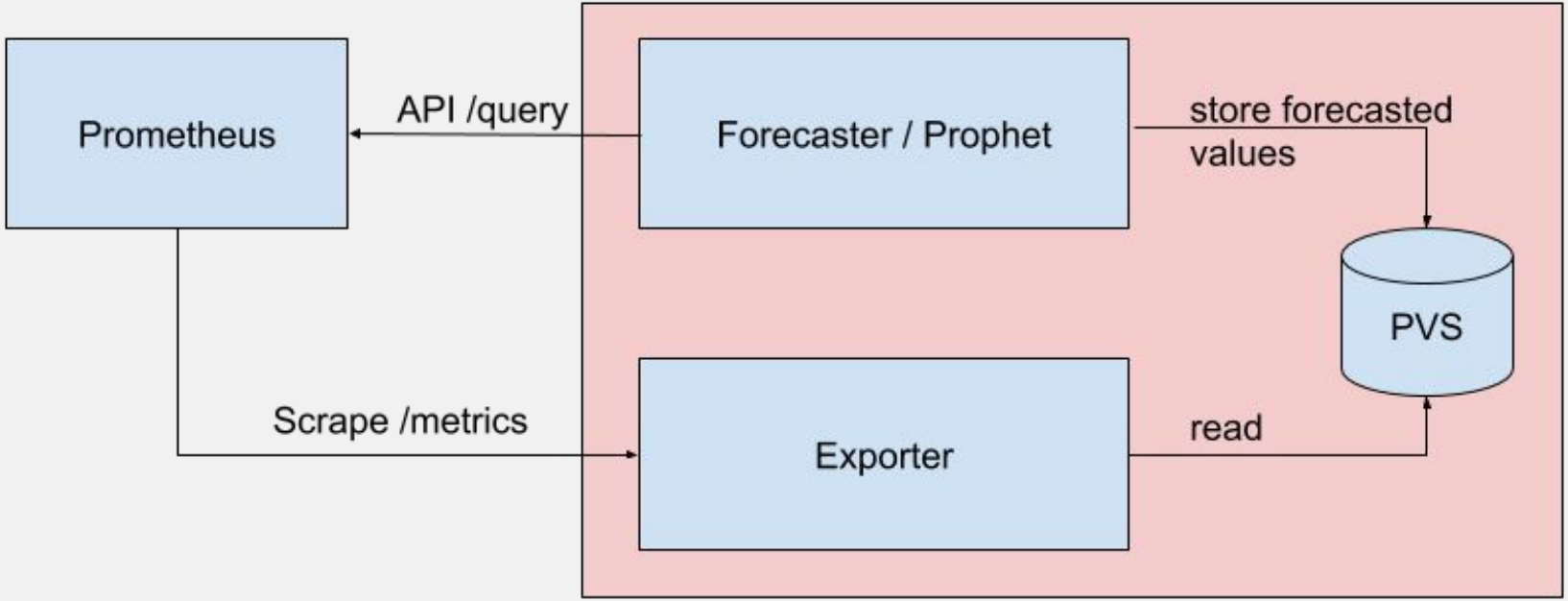
100% OpenSource Tooling

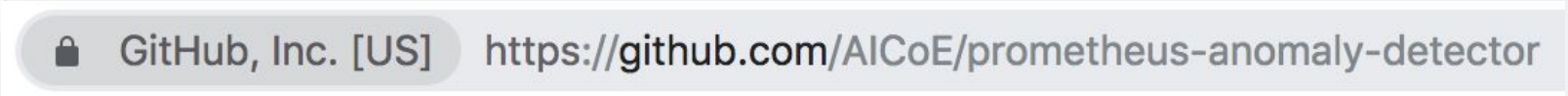


Now what? I want to <insert installer img>

Prometheus Training Pipeline




Prometheus
API /query
Forecaster / Prophet
store forecasted values
PVS
Scrape /metrics
Exporter
read



Dockerfile	Update Dockerfil
Makefile	Add Makefile for ease of
README.md	Update README.md
app.py	Add more comments for
ceph.py	Add functionality to retai
model.py	Make the live data query
prometheus.py	Make the live data query
requirements.txt	Update requirements.txt
train-prophet-deployment-templa...	Add deployment templat

- Ready to use container
  - Local deployment
  - Kubernetes
  - OpenShift build config

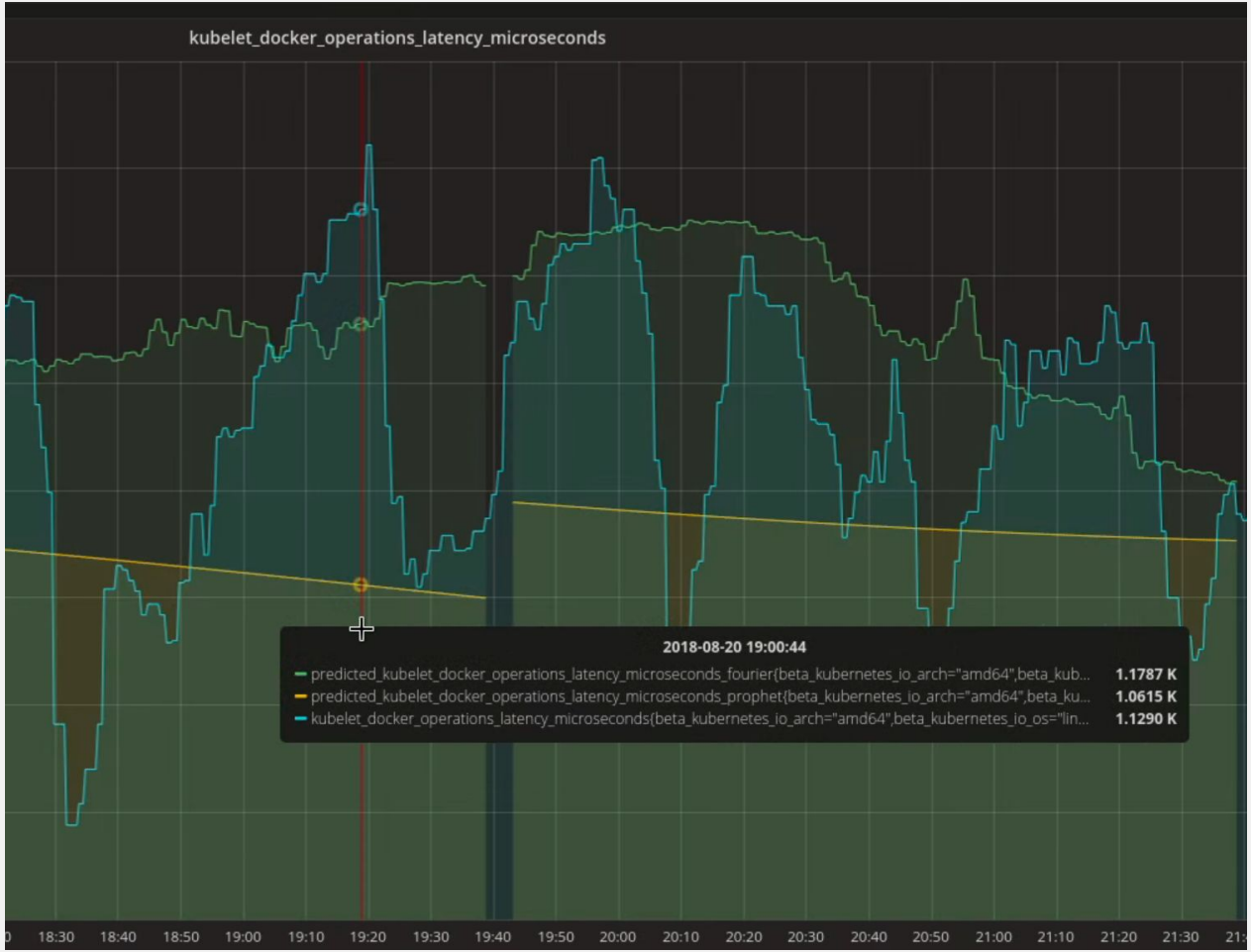
CoE/prom-ad
http://bit.ly/2yulCfh

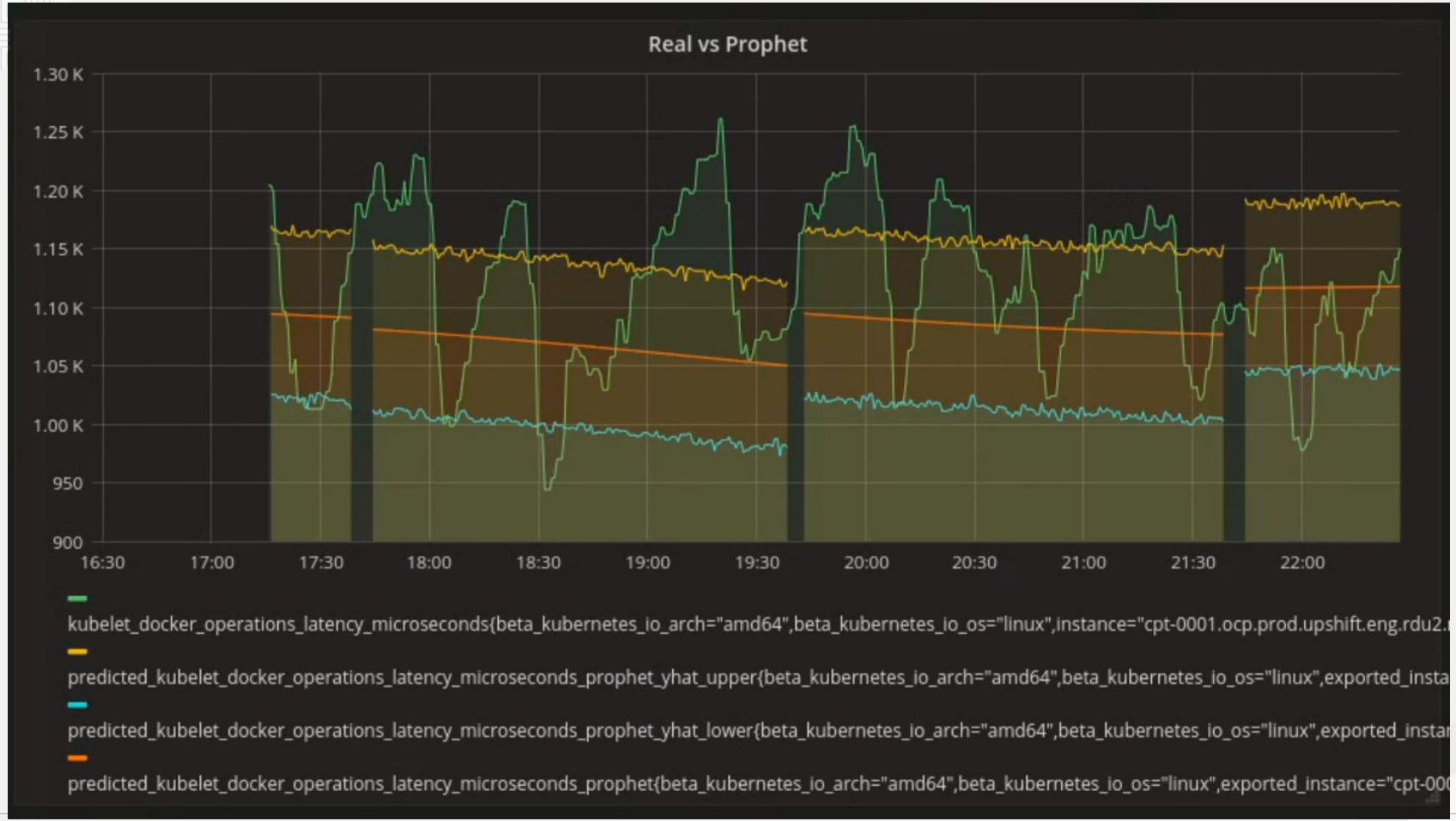
Runtime configuration

```
29  # Specific metric to run the model on
30  metric_name = os.getenv('METRIC_NAME','kubelet_docker_operations_latency_microseconds')
```

Expose predictions via **/metrics** endpoint

```
# HELP predicted_kubelet_docker_operations_latency_microseconds_prophet_anomaly Detected Anomaly using the Prophet model
# TYPE predicted_kubelet_docker_operations_latency_microseconds_prophet_anomaly gauge
predicted_kubelet_docker_operations_latency_microseconds_prophet_anomaly{beta_kubernetes_io_arch="amd64",beta_kubernetes_io_os="linux"
0001.ocp.prod.upshift.eng.rdu2.redhat.com",operation_type="version",provider="rhos",quantile="0.5",region="compute",size="small"} 0.0
# HELP predicted_kubelet_docker_operations_latency_microseconds_fourier_anomaly Detected Anomaly using the Fourier model
# TYPE predicted_kubelet_docker_operations_latency_microseconds_fourier_anomaly gauge
predicted_kubelet_docker_operations_latency_microseconds_fourier_anomaly{beta_kubernetes_io_arch="amd64",beta_kubernetes_io_os="linux"
0001.ocp.prod.upshift.eng.rdu2.redhat.com",operation_type="version",provider="rhos",quantile="0.5",region="compute",size="small"} 0.0
```


kubelet_docker_operations_latency_microseconds
2018-08-20 19:00:44
predicted_kubelet_docker_operations_latency_microseconds_fourier{beta_kubernetes_io_arch="amd64",beta_kub... 1.1787 K
predicted_kubelet_docker_operations_latency_microseconds_prophet{beta_kubernetes_io_arch="amd64",beta_ku... 1.0615 K
kubelet_docker_operations_latency_microseconds{beta_kubernetes_io_arch="amd64",beta_kubernetes_io_os="lin... 1.1290 K
18:30
18:40
18:50
19:00
19:10
19:20
19:30
19:40
19:50
20:00
20:10
20:20
20:30
20:40
20:50
21:00
21:10
21:20
21:30


Real vs Prophet
1.30 K
1.25 K
1.20 K
1.15 K
1.10 K
1.05 K
1.00 K
950
900
16:30
17:00
17:30
18:00
18:30
19:00
19:30
20:00
20:30
21:00
21:30
22:00
kubelet_docker_operations_latency_microseconds{beta_kubernetes_io_arch="amd64",beta_kubernetes_io_os="linux",instance="cpt-0001.ocp.prod.upshift.eng.rdu2.
predicted_kubelet_docker_operations_latency_microseconds_prophet_yhat_upper{beta_kubernetes_io_arch="amd64",beta_kubernetes_io_os="linux",exported_insta
predicted_kubelet_docker_operations_latency_microseconds_prophet_yhat_lower{beta_kubernetes_io_arch="amd64",beta_kubernetes_io_os="linux",exported_instan
predicted_kubelet_docker_operations_latency_microseconds_prophet{beta_kubernetes_io_arch="amd64",beta_kubernetes_io_os="linux",exported_instance="cpt-000

Alerting Rules

```
groups:
- name: Testing alert
  rules:

  - alert: MetricOutofProphetBounds
    expr: kubelet_docker <  ignoring(job, instance) predicted_values_prophet_yhat_lower or kubelet_docker  > ignoring(job, instance) predicted_values_prophet_yhat_upper
    #for: 5m
    annotations:
      summary: "Metric out of bounds"
      description: "Metric is out of range of the predicted Prophet values"

  - alert: MetricOutofFourierBounds
    expr: kubelet_docker < ignoring(job, instance) predicted_values_fourier_yhat_lower or kubelet_docker > ignoring(job, instance) predicted_values_fourier_yhat_upper
    annotations:
      summary: "Metric out of bounds"
      description: "Metric is out of range of the predicted Fourier values"

~
~
~
```

notebooks
http://bit.ly/2PIZZVG

gh/AICoE/p-influx
http://bit.ly/2y6CvwX

CoE/prophet
http://bit.ly/2pLzGNj

Project Thoth and Bots
http://bit.ly/2zYfb6h

Meta-data tooling
http://bit.ly/2A1hXHX

QUESTIONS?

CoE/prom-ad
http://bit.ly/2yulCfh

No more g+

dhatinc

linkedin.com/company/red-hat

Hat

OpenDataHub
http://bit.ly/2y6Nh6m

user/RedHatVideos

gh/AICoE/p-lts
http://bit.ly/2Qw9pho