Open Source Monitoring and Observability , where are we now ?

Kris Buytaert @krisbuytaert

NLUUG , May 2024

O11y

An opinionated talk about the Open Source Monitoring and Observability tooling landscape

Kris Buytaert

- I used to be a developer
- Then I became an Ops person
- Chief Yak Shaver @ o11y.eu
- Organiser of #devopsdays, #cfgmgmtcamp, #loadays, …
- Cofounder of all of the above
- Everything is a Freaking DNS Problem
- DNS : devops needs sushi
- @krisbuytaert on github/mastodon/x/bluesky

- Inuits.eu Spinoff
- Open Source Observability
- Currently supporting the Prometheus Ecosystem
- Professional Services & Support (now)
- LTS Release (Long Term stable release) (now)
- Prometheus “Distribution” (soon)

This Talk:

- What is Monitoring / Observability ?
- History
- What's out there ?

What is monitoring?

- High level overview of the state of a service/component
- Availability
- Technical components
- Performance ?

Monitoring answers the question : What is going on?

What's observability?

- Understand how your services behave
- Like you are at their place
- Without incident specific code

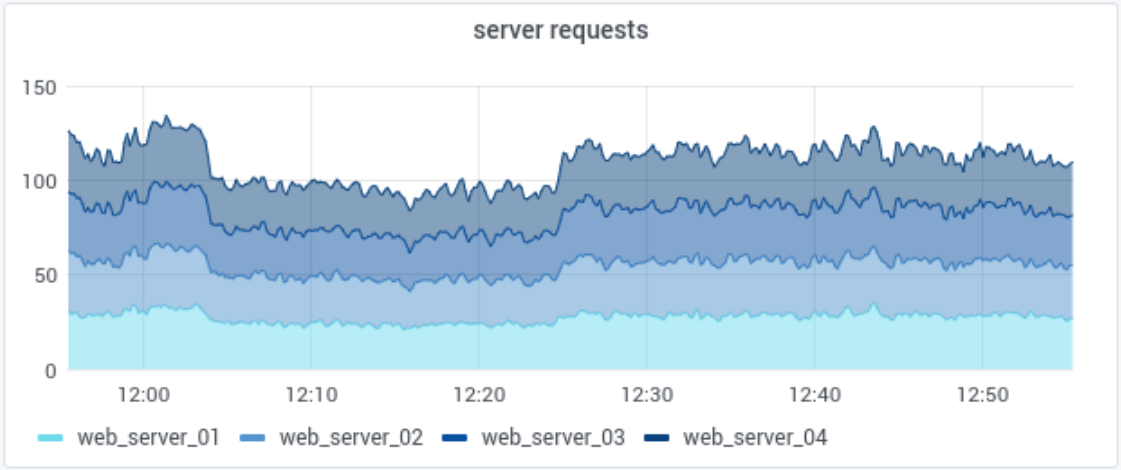
Observability answers : Why is this going on?

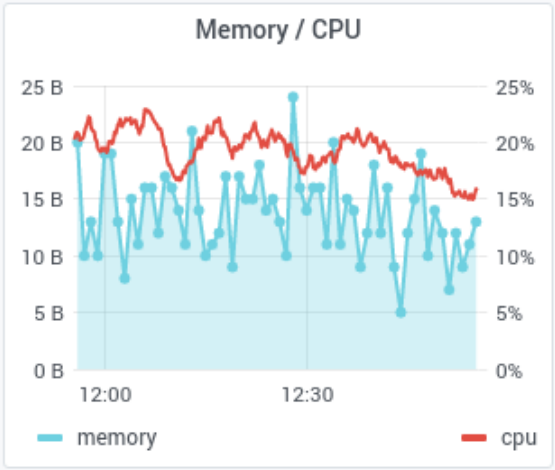
How do monitoring and observability connect?

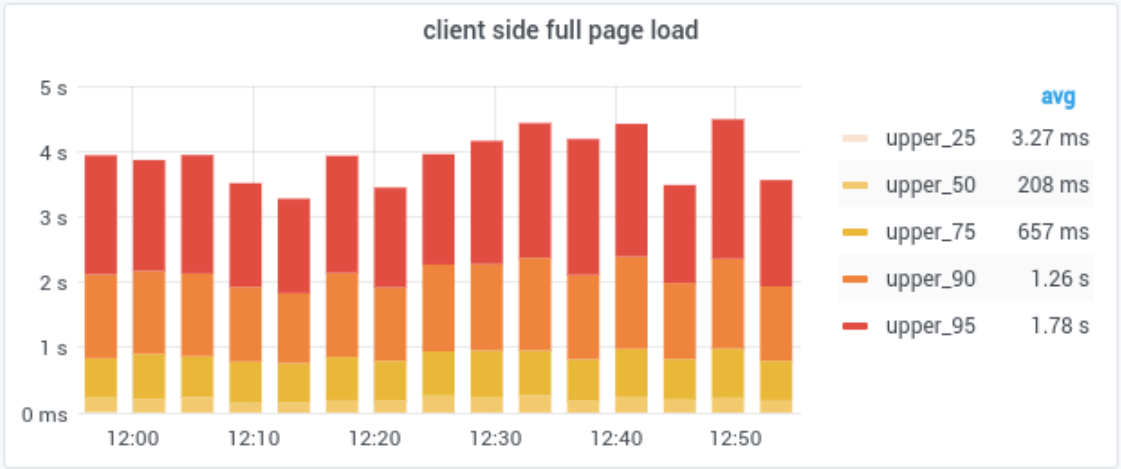
- Monitoring is required
- If lucky, monitoring is enough
- Observability is removing luck <- @roidelapluie

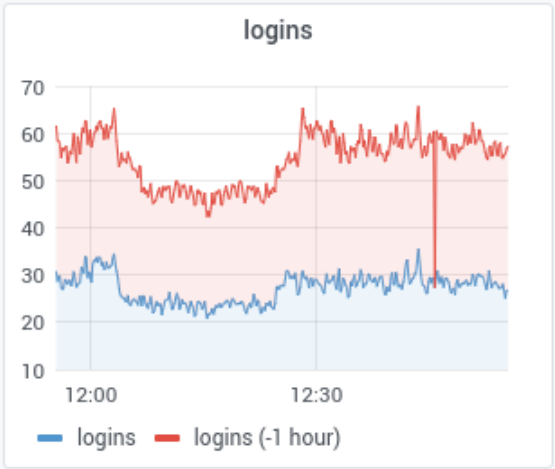
The 3 Pillars

Metrics



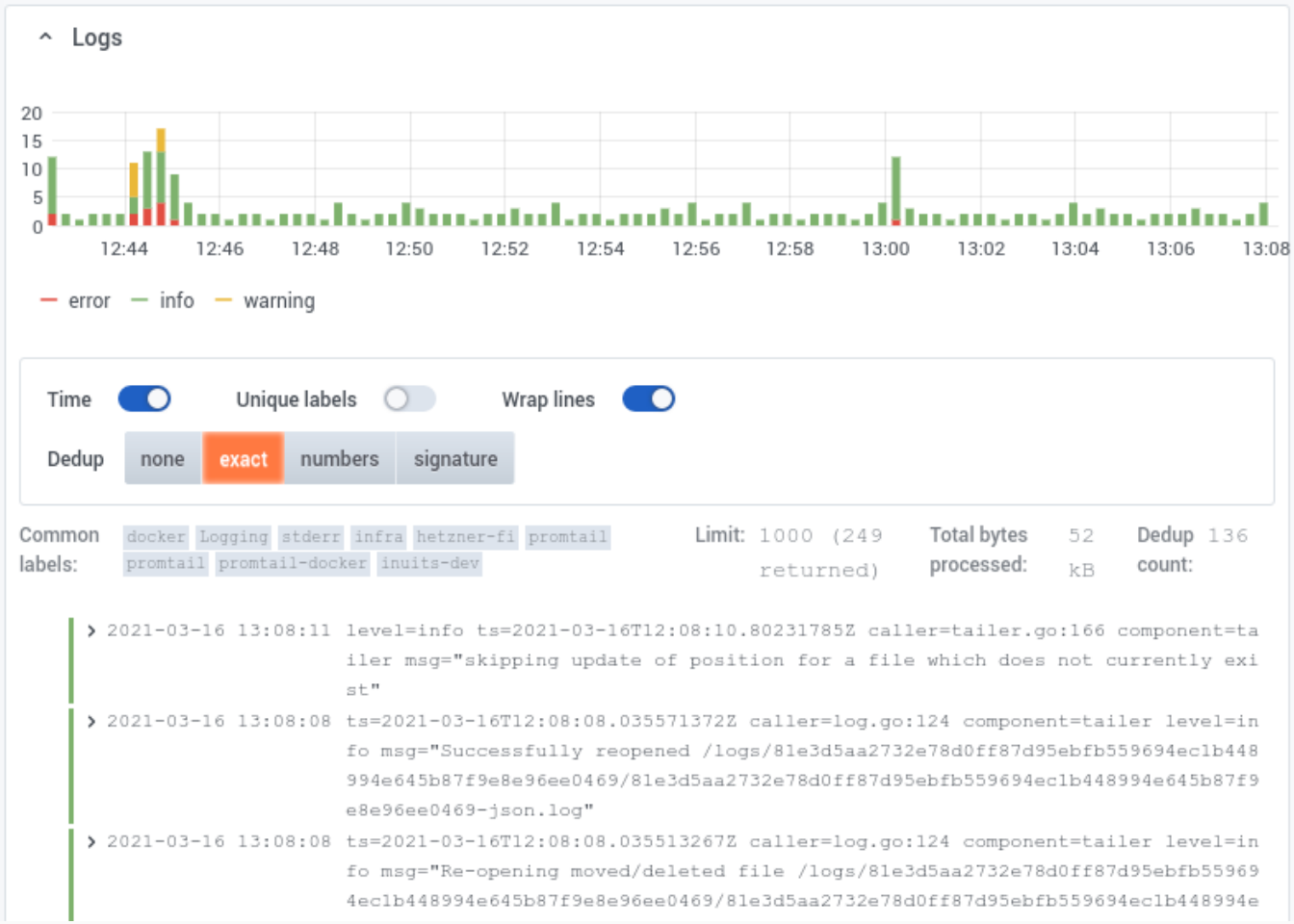






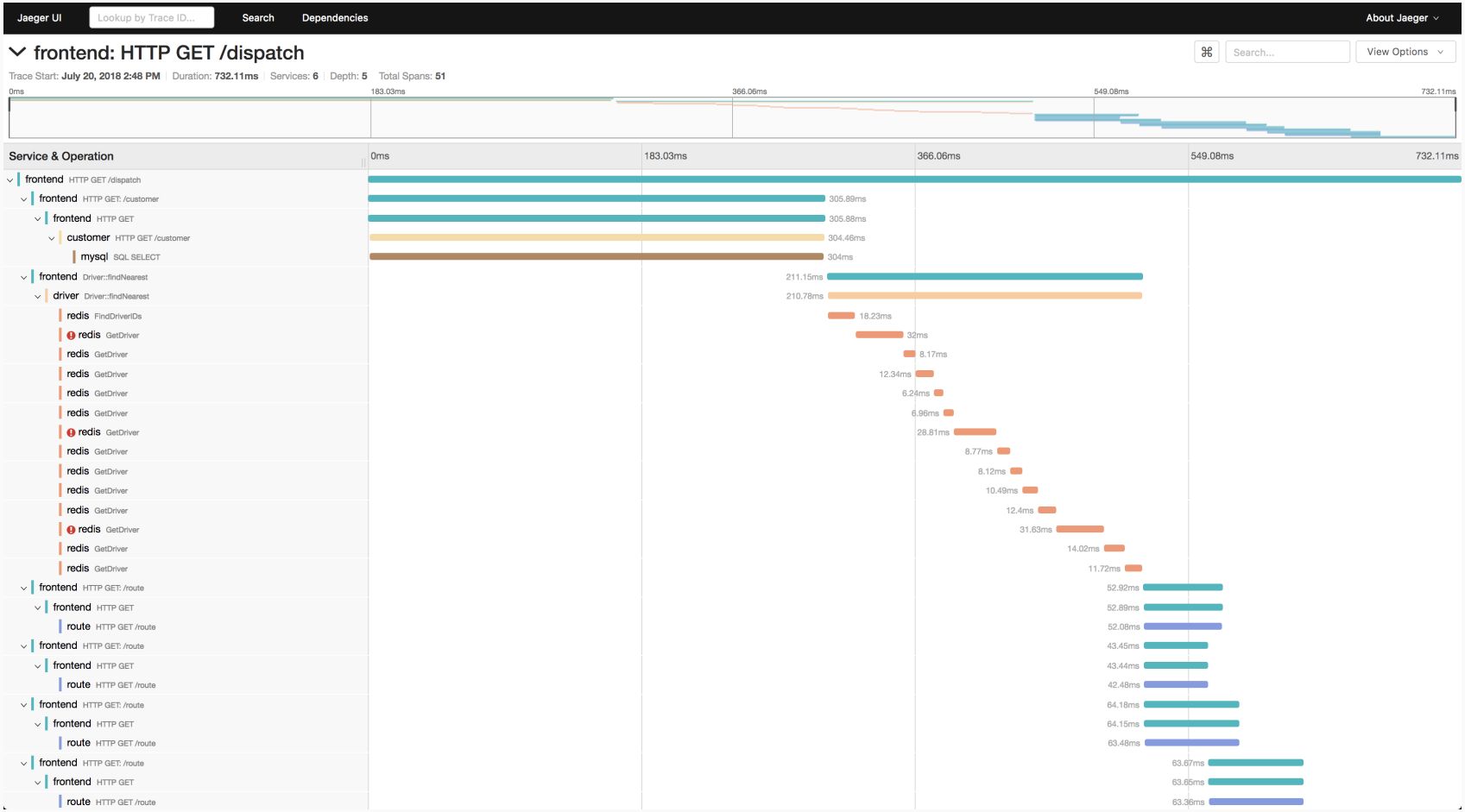
https://play.grafana.org/

Logs



https://play.grafana.org/

Traces



https://www.jaegertracing.io/

The Fourth Pillar

Figure 1: 4th pillar

Introduction, a brief history of Open Source Monitoring

July 2008 Ottawa Linux Symposium Paper

- Bloated Java Tools
- Dysfunctional Open Core Software
- DBA Required
- Nagios was king in the Open Source world

June 2011 #monitoringsucks

- John Vincent (@lusis) , june 2011
- A #devops sub-movement

> Heck no! Monitoring is AWESOME. Metrics are AWESOME. I love it. Here's what I don't love: - Having my hands tied with the model of host and service bindings. - Having to set up "fake" hosts just to group arbitrary metrics together - Having to either collect metrics twice - once for alerting and another for trending - Only being able to see my metrics in 5 minute intervals - Having to chose between shitty interface but great monitoring or shitty monitoring but great interface - Dealing with a monitoring system that thinks **IT** is the system of truth for my environment - Perl (I kid...sort of) - Not actually having any real choices

Why #monitoringsuck(ed) (continued.)

- Manual configuration (GUI)
- Not in sync with reality
- Hosts only
- Services sometimes
- Application never
- Chaos or out of sync with reality
- Alert Fatigue

October 2011 #monitoringlove

- #devopsdays Rome 2011
- Ulf Mansson
- A new found love for monitoring
- Triggered by { New Open Source Tools * Automation }

Automation of #monitoring brought back the #love

What we want : Modularity

Small , well suited components to

- Collect
- Transport / Mangle
- Store
- Analyse
- Act / Alert
- Visualize

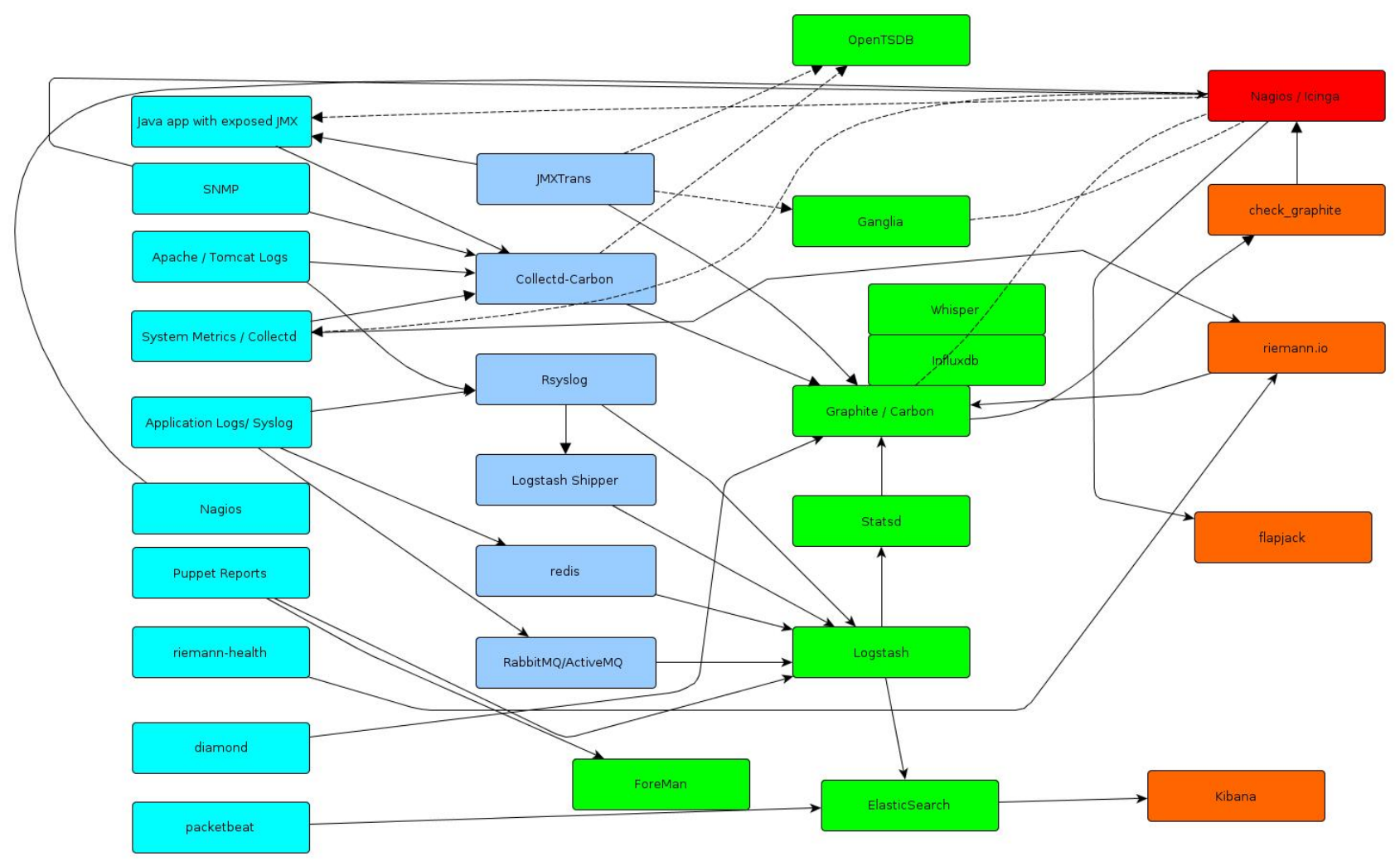


Figure 2: What We Wanted

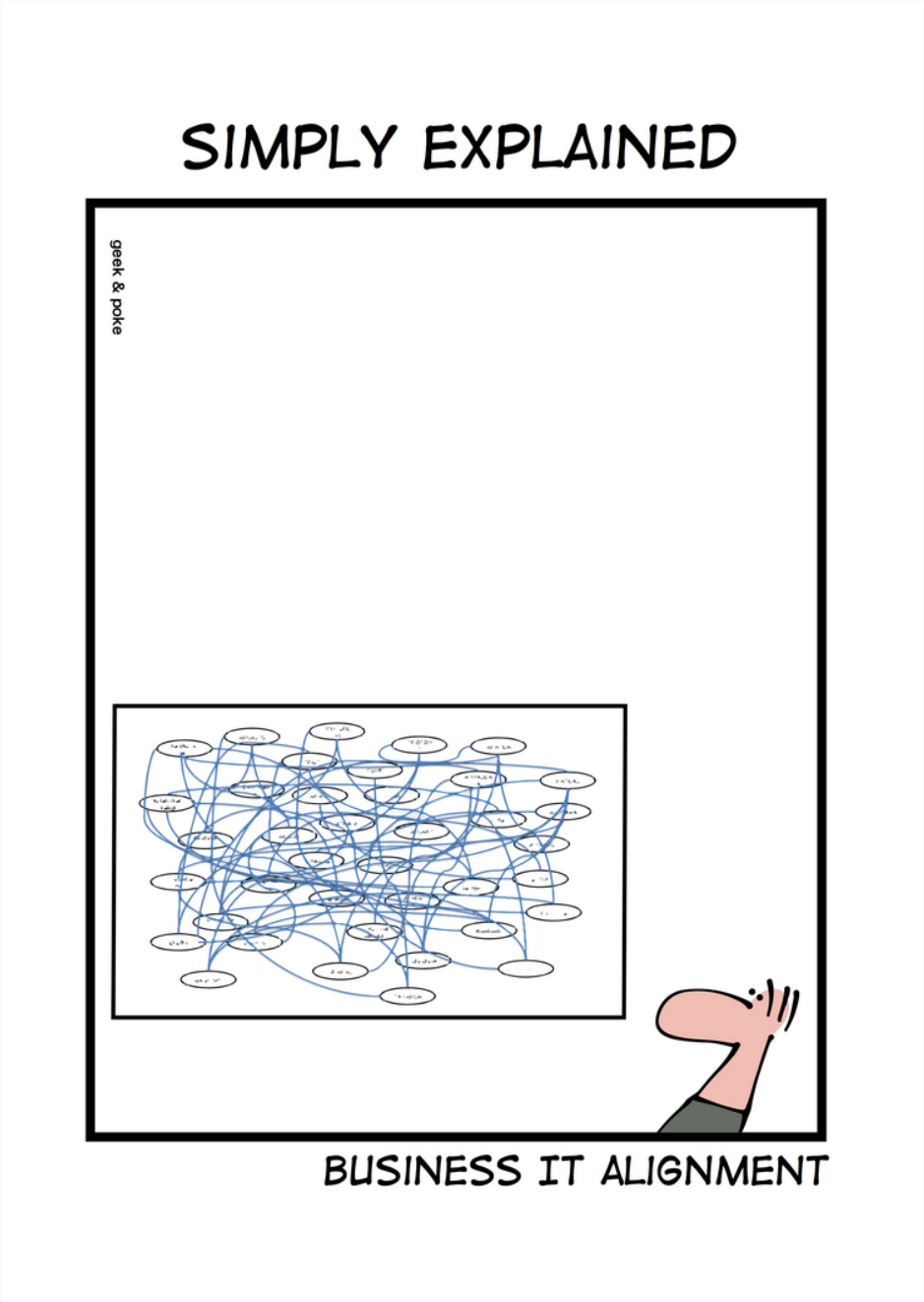


Figure 3: Reality

Data Collection & Shipping Metrics

- collectd
- Diamond
- Icinga
- Nagios
- Zabbix
- Beats
- node_exporter
- a zillion service specific prometheus exporters
- …

Data Collection & Shipping Logs

- journald
- fluentbit
- fluentd
- promtail
- beats
- rsyslog
- syslogng

Transform the data

You can build pipelines in tools such as :

- Logstash
- Fluentd
- Otel (collector)

Store the timeseries data

- Prometheus
- Thanos
- Cortex
- Mimir
- m3
- VictoriaMetrics
- TimescaleDB
- Clickhouse

(Not all of these are prometheus compatible)

Store the Logs & Traces data

- OpenSearch
- Loki
- Tempo
- Jaeger
- Clickhouse

Store the data

- Long term vs Short Term
- data, vs derived data
- default prometheus configuration

Visualize the data

- Grafana
- Kibana / OpenSearch Dashboard
- Perses (CNCF)
- Icinga

Alert

- Alertmanager
- Signalllo
- Icinga
- Alerta
- (Saas Services ..).

Wake You up ?

Schedule and Trigger

- sms
- mobile app alerts
- messages
- phone calls

Trough

- Custom scripts
- Saas offerings
- Little Open Source alternatives
  - OpenDuty
  - Iris
  - Oncall

Analyze / Predict etc

aka “AI Ops”

Previous efforts :

- Skyline
- Occulus
- LogIslands ?

No one size fits all..

- Tamland
- Facebook lib
- RedHat R&D

Lots of custom work… Lots of data needed… Lots of power needed…

Q: What am I missing ?

Existing Stacks

- olk (opensearch, logstash, kibana (formerly elk)
- lgtm (loki, grafana, tempo, mimir) (agpl)
- tick (telegraf,influxdeb,chronograf,kapacitor) (open core)
- …

How do you monitor ?

- Is this thing still on ?

Active Checks

e.g Icinga / Zabbix etc..

- Metric Based e.g Prometheus

Prometheus

- “Traditional monitoring systems can’t deal with the constant change”
- New way of looking at “is this thing still on”
- Look at the Service vs look at the service
- “only for containers”

Prometheus

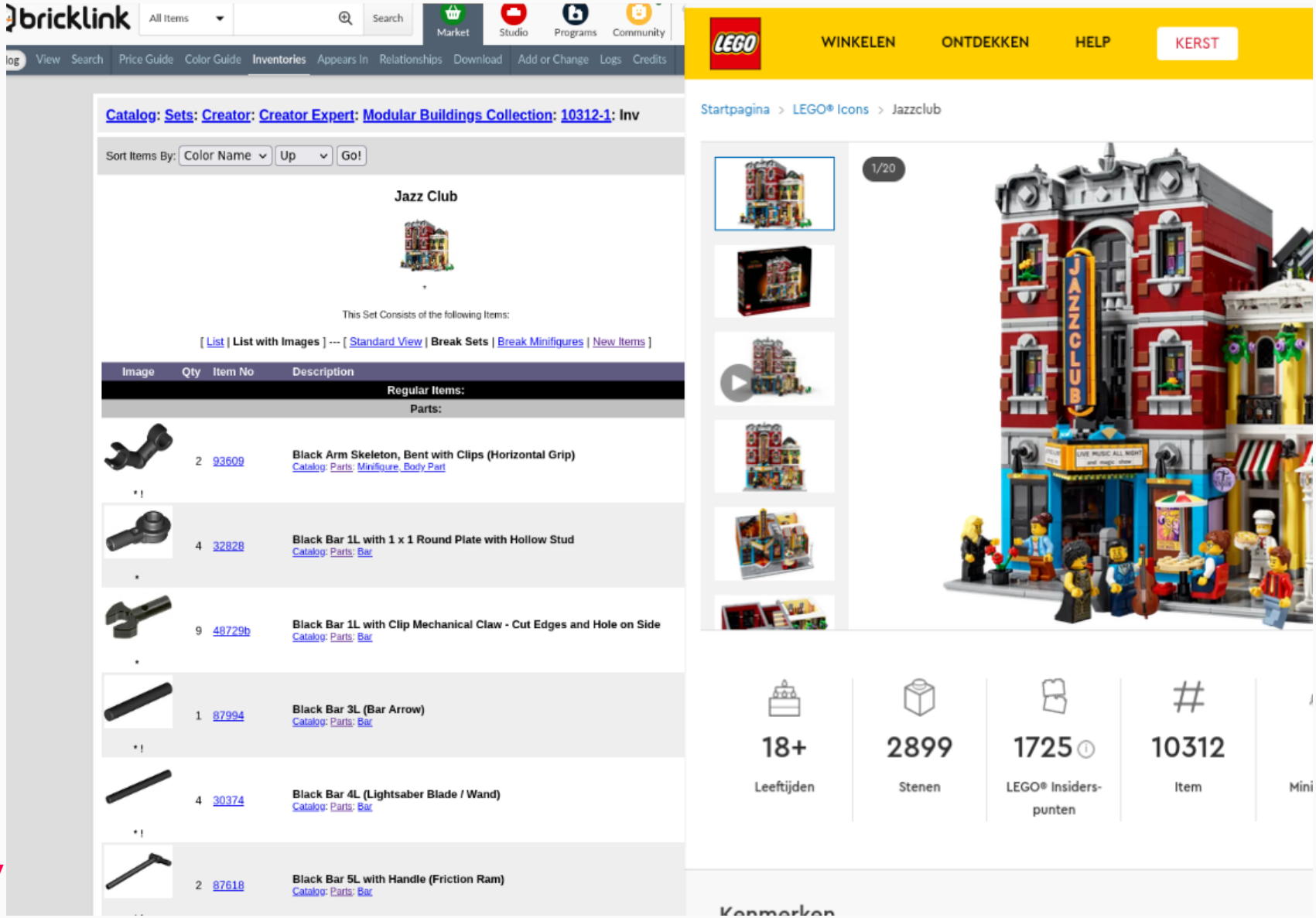
Prometheus

- Prometheus is an Open Source CNCF Project
- Collects and stores metrics - Pull-based
- Service discovery (eg. prom, puppetdb, ….)
- Alerting
- Exporters for every piece of the infra
- Tight integration with k8s (prometheus operator)
- Maintained by multiple companies
- Long-Term Support release (powered by o11y.eu)

“Troubles” with Prometheus

- Mostly short term storage
- ssl fun .. (accessing the data)
- #blackbox exporter
- basic checks are missing (is service running ?)
- default alerts,
- default dashboards
- mixins are broken
- strong k8s focus

Pick a brick vs Icons



How to get from monitoring to observability

Whats your goal in observability

- We expect performance problems
- We really have performance problems
- We have chaos , better insights in what we run
- We need more Hipster Credits
- We just want prometheus, olk and jaeger

Step 1

- Fix your monitoring
  - Single Source of Truth
  - Automated
  - Create clear and Actionable Alerts
- Keep it GREEN

Step 2

- Really,
- Fix your monitoring
  - Single Source of Truth
  - Automated
  - Create clear and Actionable Alerts
- No manual configurations !
- No long running Acknowledgements
- Keep it GREEN

Then

- Fix your metrics
- I bet you have regression on scraping your Metrics
- I bet you have some broken dashboards
- Do you still have your metrics from last year ?
- Fix your logs
- I bet you logshipping is partially broken

Ask

- Who wants Observability ?
- Devs / Management / Ops ?
- What do they really want ?
- Get them in one room
- Ask them what is really hurting them ?
- Where they need help …
- Listen,
- This sounds trivial .. yet 10+ years of devops and still

Then

- Choosing Observability Stack
- Who owns your O11y stack ?
- Who owns your Data ?
- Beware of the Fauxpen Source
- Build your automated Observability Infrastructure
- Monitor it
- Pick a Project to start investigating.
- Build dashboards together with your peers.
- From multiple angles

What is still missing ?

- Probably nothing
- This might be sufficient for your use cases.
- Except if it isn’t.
- You might need traces

Tracing Tools & Protocols

- Zipkin (Twitter)
- Jaeger (Uber/CNCF)
- Tempo (Grafanalabs)
- OpenTracing (RIP) + OpenCensus (RIP) = OpenTelemetry

OTEL

- “Vendor Driven Standardisation”
- Protocol, not implementation
  - Metrics
  - Tracing
  - Logging
- Claims AutoInstrumentation,
- Different components have different maturity levels
- Still needs to do some work ..

There is no magic

Paul Johnston - current focus: moving electrons
@PaulDJohnston

This.

Cannot believe how many people think that generalised instrumentation is somehow the "best" solution.

"What do I need to know? It would be good to instrument exactly that wouldn't it!"

Getting data and querying is hard.

Producing the data you need in the first place? 👍

Luc van Donkersgoed @donkersgood · Sep 10

Replying to @dfrasca80 and @honeycombio

Not a silly question at all! OpenTelemetry supports auto instrumentation, but I'm a strong proponent for manual instrumentation. This allows me to add business context which makes the output WAY more valuable.

Otel Collector

- Man in the Middle
- Standarised endpoint
- mangle /ship to
- Multiple backends
- Define your own pipeline
- Other vendors have collectors too ..

Pipelines

```
service:
  pipelines:
    traces:
      receivers: [otlp,jaeger,zipkin]
      processors: [batch]
      exporters: [otlp]
    metrics:
      receivers: [otlp,prometheus]
      processors: [batch]
      exporters: [prometheus]

    traces/lab:
      receivers: ["otlp", "opencensus", "jaeger", "zipkin"]
      processors: ["batch"]
      exporters: ["otlp/lab", "otlphttp/coroot"]
```

eBPF ?

- Linux Kernel level
- extended Berkeley Packet filter
- Framework to attach to low level functions
- Framework used to collect statistics
- Framework to build a monitoring tool
- “A marriage of strace tcpdump on steriods”
- Maybe there is magic after all ?
- easy on k8s,
- hard outside

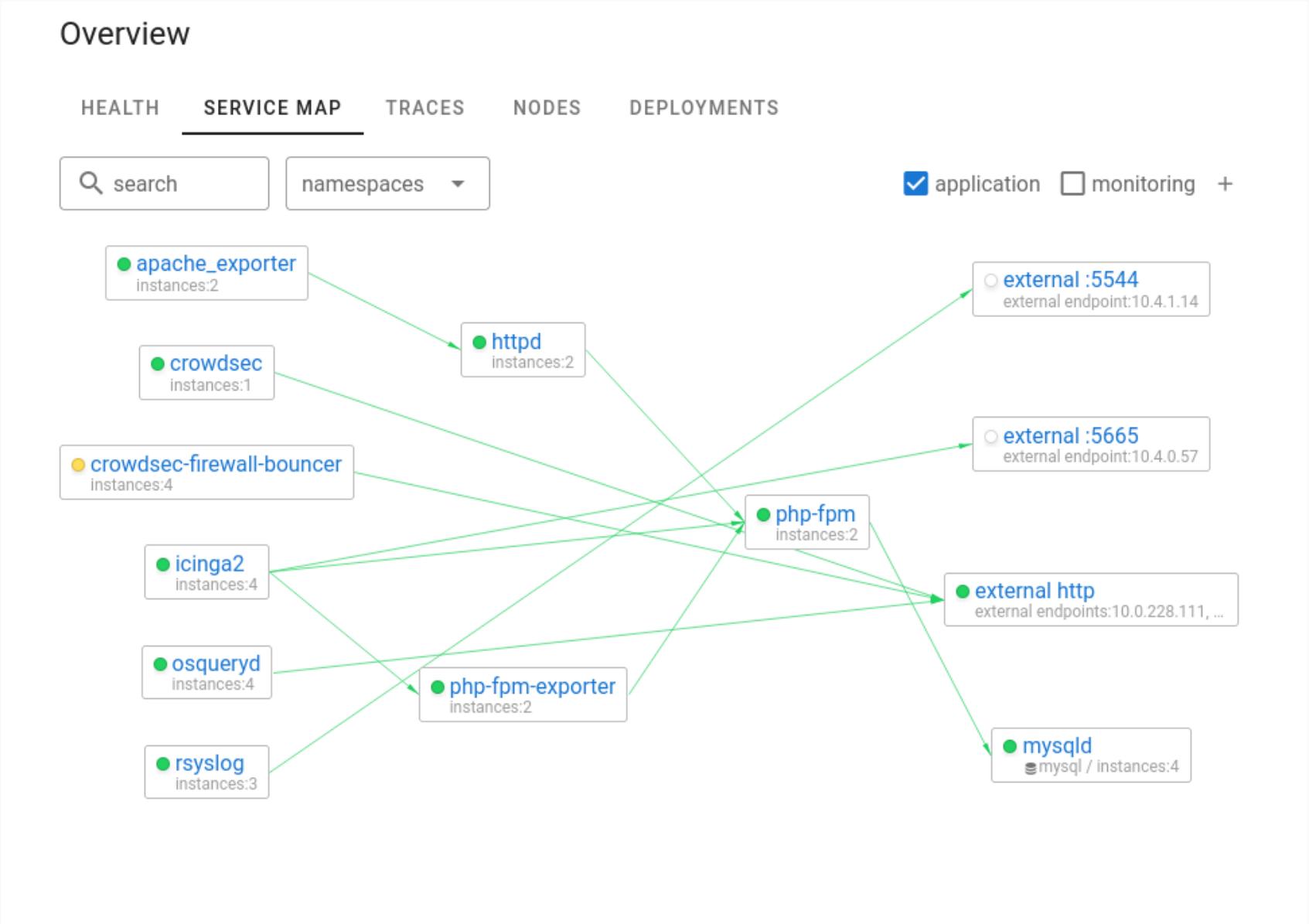
Tools

- ebpf_exporter
- opentelemetry-network
- Falco (security)
- Coroot
- Pixie (k8s)
- Odigos (k8s)
- Tracee (docker/k8s)
- Caretta (maps your dependencies) (k8s)
- Keppler (power usage / k8s)

Coroot

- 1.0 released
- Cloud-agnostic : bare metal, k8s …
- agent + server
- ebpf based exporter
- Ships : metrics, logs and traces
- zero instrumentation
- Stores per project data in Prometheus / Clickhouse

Service Map


Overview
HEALTH
SERVICE MAP
TRACES
NODES
DEPLOYMENTS
search
namespaces
application
monitoring
apache_exporter
instances:2
crowdsec
instances:1
crowdsec-firewall-bouncer
instances:4
icinga2
instances:4
osqueryd
instances:4
rsyslog
instances:3
httpd
instances:2
php-fpm-exporter
instances:2
php-fpm
instances:2
external :5544
external endpoint:10.4.1.14
external :5665
external endpoint:10.4.0.57
external http
external endpoints:10.0.228.111, ...
mysqld
mysql / instances:4

Health

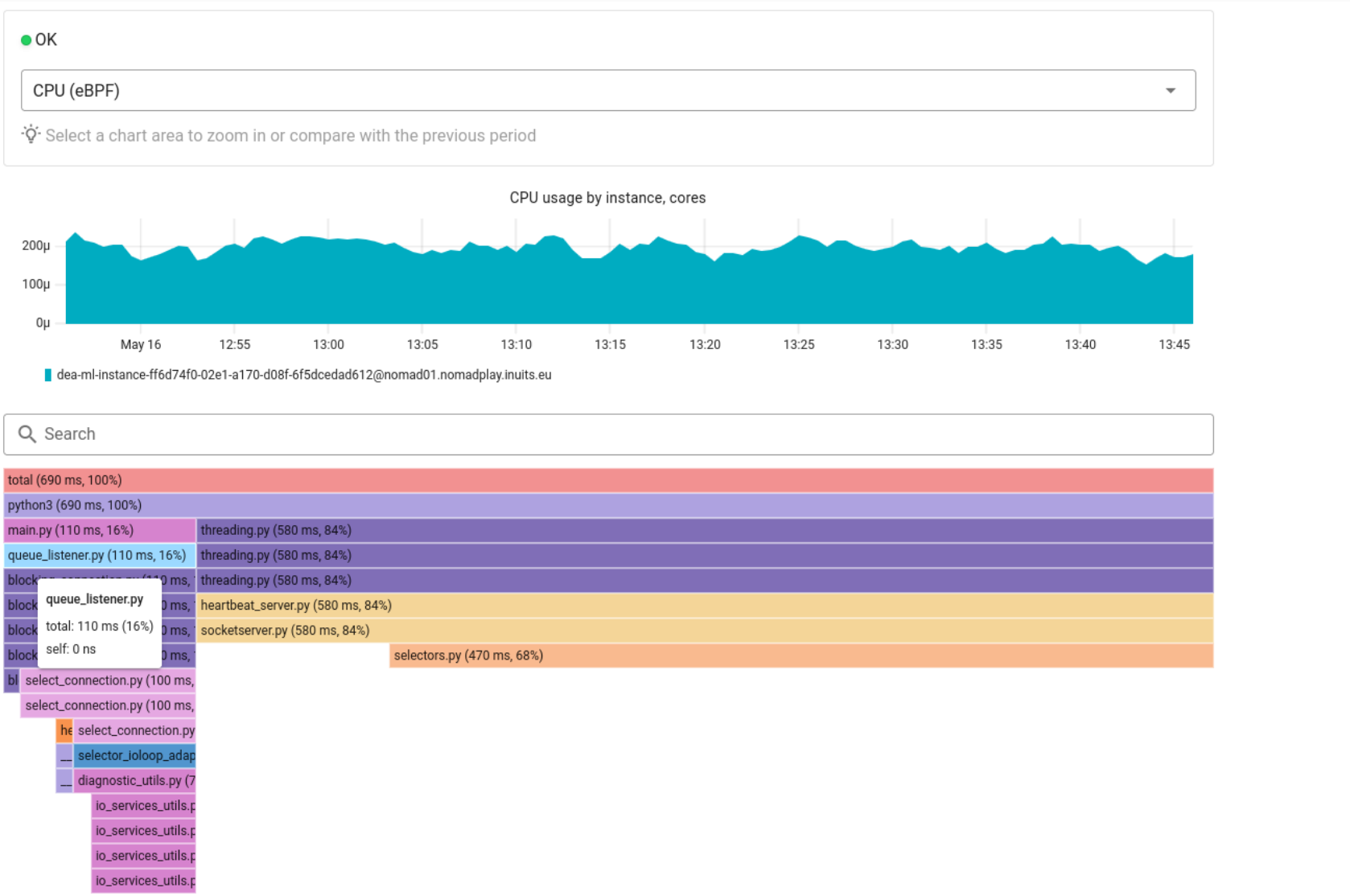
HEALTH | SERVICE MAP | TRACES | NODES | DEPLOYMENTS

search | namespaces | application | monitoring | +

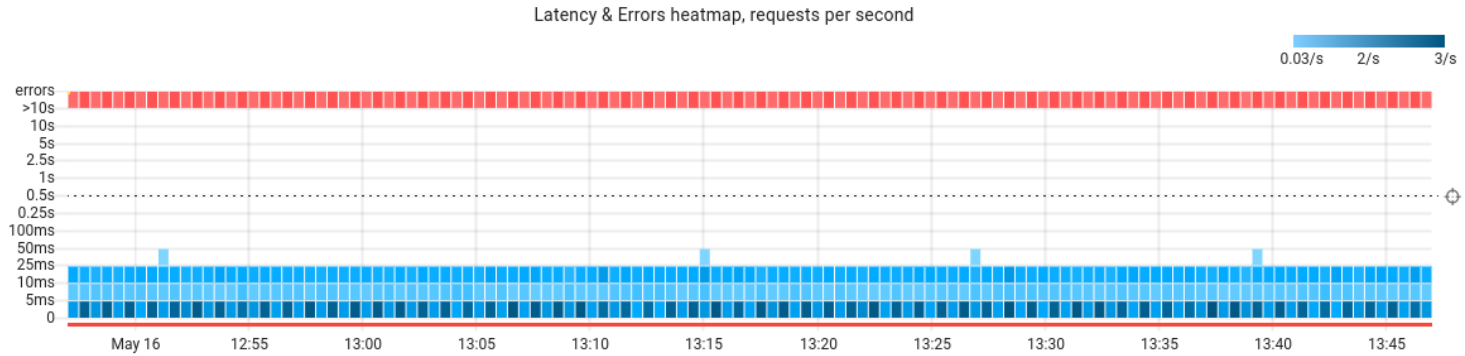
5 SLO violation | 53 Warning | 5 Errors in logs | 44 OK

Application	Errors	Latency	Upstreams	Instances	Restarts	CPU	Mem	I/O	Disk	Net	L
consul	–	10s	55/55	6/6	–	–	–	2%	90%	25ms	
mysql	12%	25ms	–	3/3	–	–	–	<1%	82%	10ms	
mysql-06ebec06-c4ff-902f-80e6-36a...	13%	192ms	0/1	1/1	–	–	–	<1%	3%	failed conns	
mysql-45f329ab-2ef3-29d1-5538-cc...	9%	74ms	0/2	1/1	–	–	–	<1%	3%	25ms	
mysql-4fcfd407-cf7e-8810-a205-05f...	8%	92ms	0/1	1/1	–	–	–	<1%	2%	failed conns	
auditbeat	–	–	–	7/7	–	–	–	2%	93%	26ms	1 uniq
backup-nomad-jobs-5b60a0b5-3b73...	–	–	1/1	0/1	–	–	–	–	–	–	
compactor-160cd7ab-9630-3164-ef...	–	5ms	0/1	1/1	–	–	–	–	–	failed conns	
compactor-6491125d-0e18-7c9e-70...	–	5ms	–	1/1	–	–	–	–	–	failed conns	1 uniq
crowdsec-firewall-bouncer	–	–	1/1	10/10	2158	–	–	<1%	82%	44ms	
distributor-048a9392-dbcd-de17-db...	–	49ms	3/4	1/1	–	–	–	–	–	failed conns	
distributor-62235234-e70b-f793-a09...	–	49ms	1/2	1/1	–	–	–	–	–	failed conns	
distributor-772c1fa9-10e2-1d37-b22...	<1%	49ms	3/4	1/1	–	–	–	–	–	failed conns	
distributor-7a93fd91-f783-d893-b60...	<1%	50ms	3/4	1/1	–	–	–	–	–	failed conns	
distributor-daf5993e-f588-a13a-580...	–	49ms	3/4	1/1	–	–	–	–	–	failed conns	
distributor-e26a2c2d-bf29-6c41-95b...	<1%	49ms	3/4	1/1	–	–	–	–	–	failed conns	
dnf-makecache	–	–	1/1	0/11	–	–	–	–	–	–	
icinga2	–	–	3/4	17/17	–	–	–	2%	93%	failed conns	
ingester-559935ea-58f0-4786-f529-...	–	5ms	0/1	1/1	–	–	–	<1%	2%	failed conns	
ingester-ac391338-00fa-f51b-a3bc-...	–	50ms	0/1	1/1	–	–	–	<1%	3%	failed conns	

Profiling


OK
CPU (eBPF)
Select a chart area to zoom in or compare with the previous period
CPU usage by instance, cores
200µ
100µ
0µ
May 16
12:55
13:00
13:05
13:10
13:15
13:20
13:25
13:30
13:35
13:40
13:45
dea-ml-instance-ff6d74f0-02e1-a170-d08f-6f5dcedad612@nomad01.nomadplay.inuits.eu
Search
total (690 ms, 100%)
python3 (690 ms, 100%)
main.py (110 ms, 16%)
threading.py (580 ms, 84%)
queue_listener.py (110 ms, 16%)
threading.py (580 ms, 84%)
threading.py (580 ms, 84%)
heartbeat_server.py (580 ms, 84%)
socketserver.py (580 ms, 84%)
selectors.py (470 ms, 68%)
queue_listener.py
total: 110 ms (16%)
self: 0 ns
select_connection.py (100 ms,
select_connection.py (100 ms,
select_connection.py
selector_ioloop_adap
diagnostic_utils.py (7
io_services_utils.p
io_services_utils.p
io_services_utils.p
io_services_utils.p

Tracing



Trace ID	Client	Status	Duration	Name	Details
176f1871	proxysql	OK	11.96ms	query	SELECT (SELECT VARIABLE_VALUE FROM performance_schema.global_status WHERE VAR
6f82bbbd	proxysql	OK	3.03ms	query	SELECT (SELECT VARIABLE_VALUE FROM performance_schema.global_status WHERE VAR
a457ea0a	proxysql	OK	13.74ms	query	SELECT (SELECT VARIABLE_VALUE FROM performance_schema.global_status WHERE VAR
40a7d695	proxysql	OK	5.02ms	query	SELECT 1 AS one FROM `authie_sessions` WHERE `authie_sessions`.`browser_id` =
1dc496d8	proxysql	OK	1.62ms	query	SELECT (SELECT VARIABLE_VALUE FROM performance_schema.global_status WHERE VAR
40c97512	proxysql	OK	13.76ms	query	SELECT (SELECT VARIABLE_VALUE FROM performance_schema.global_status WHERE VAR
15447ac7	proxysql	OK	21.07ms	query	SELECT (SELECT VARIABLE_VALUE FROM performance_schema.global_status WHERE VAR
fae343f5	proxysql	OK	15.46ms	query	SELECT (SELECT VARIABLE_VALUE FROM performance_schema.global_status WHERE VAR
1a1e1727	proxysql	OK	11.76ms	query	SELECT (SELECT VARIABLE_VALUE FROM performance_schema.global_status WHERE VAR
5a2d7145	proxysql	OK	7.09ms	query	SELECT (SELECT VARIABLE_VALUE FROM performance_schema.global_status WHERE VAR
555bdd98	proxysql	OK	5.13ms	query	SELECT `queued_messages`.* FROM `queued_messages` WHERE `queued_messages`.`lo
e2c7bd7d	proxysql	OK	4.58ms	query	SELECT 1 AS one FROM `authie_sessions` WHERE `authie_sessions`.`browser_id` =
4ee932b9	mysqld_exporter	OK	0.28ms	query	SHOW SLAVE STATUS
a82bf16c	mysqld_exporter	ERROR	0.33ms	query	SHOW ALL SLAVES STATUS

Coroot

- Easy to deploy
- Works both with k8s and other ecosystems
- Fast insight in problems
- application AND ebpf based tracing
- New way of “alerting”
- SLO Driven

What is missing ?

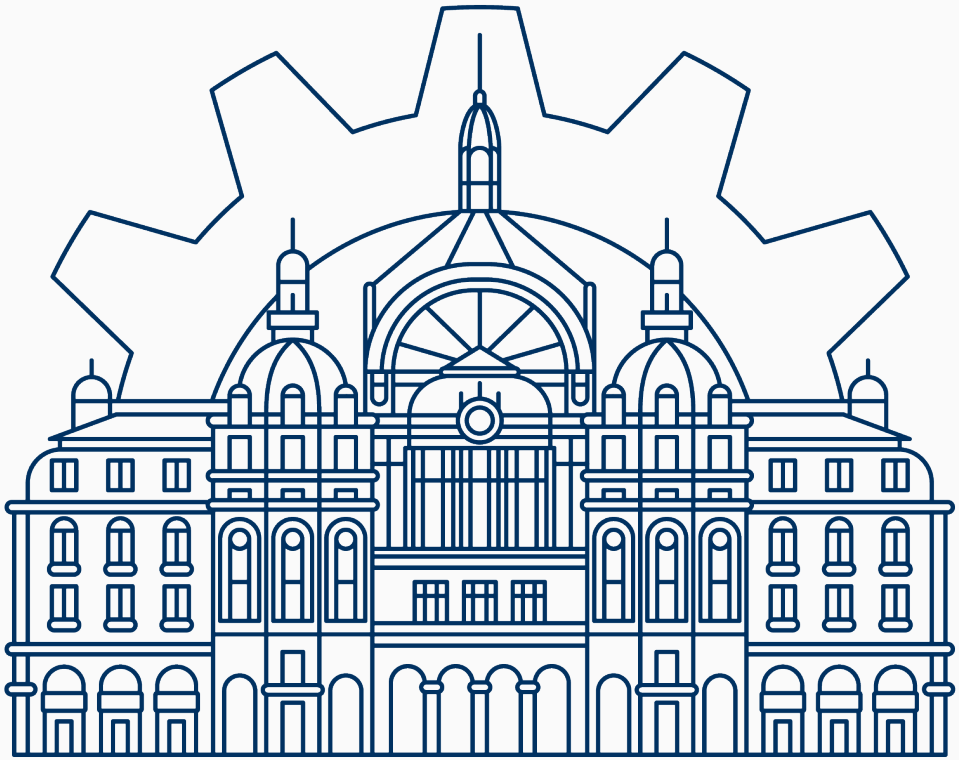
- Still no full standardisation (alerts , metric format, …)
- Open Source Alerting / Scheduling / Escalation
- (true) Open Source Backend variety
- SSOT's

Conclusion

Conclusion

- You might not need Observability (yet)
- But you DO need to fix your monitoring
- And THEN you can think about o11y
- But just adopting o11y, will not fix your broken monitoring or culture.

One more thing …



ANTWERP

DEVOPSDAYS2024

15 year anniversary

Be there ..

15 years of devopsdays

4-5 september 2024

Antwerpen

CFP still open

Ticketsales open NOW

Contact

O11y

kris@o11y.eu

https://o11y.eu

info@o11y.eu