Using Loki for Simplifying the Usage of Shasta Logs

Siqi Deng • 08.22.2021

About myself

NERSC Operations

Since January, 2018

Participated in OMNI admin work

- Took ownership of new Elasticsearch projects

Participated in N9

- Built critical components in the monitoring pipeline
  - Telemetry data shipper
  - Victoriametrics
  - Elasticsearch
  - Loki

What Shasta Provides

Monitoring UI

- Sysmgmt-health Grafana
- SMA Grafana
- Istio-system Grafana
- One Kibana

Perlmutter
X Muller
Alvarez

Query Language

- PromQL
- PostgreSQL
- KQL

Notification Engine

- Alertmanager
- Monasca-notification
- Elastalert

What if

One Monitoring UI

- Grafana

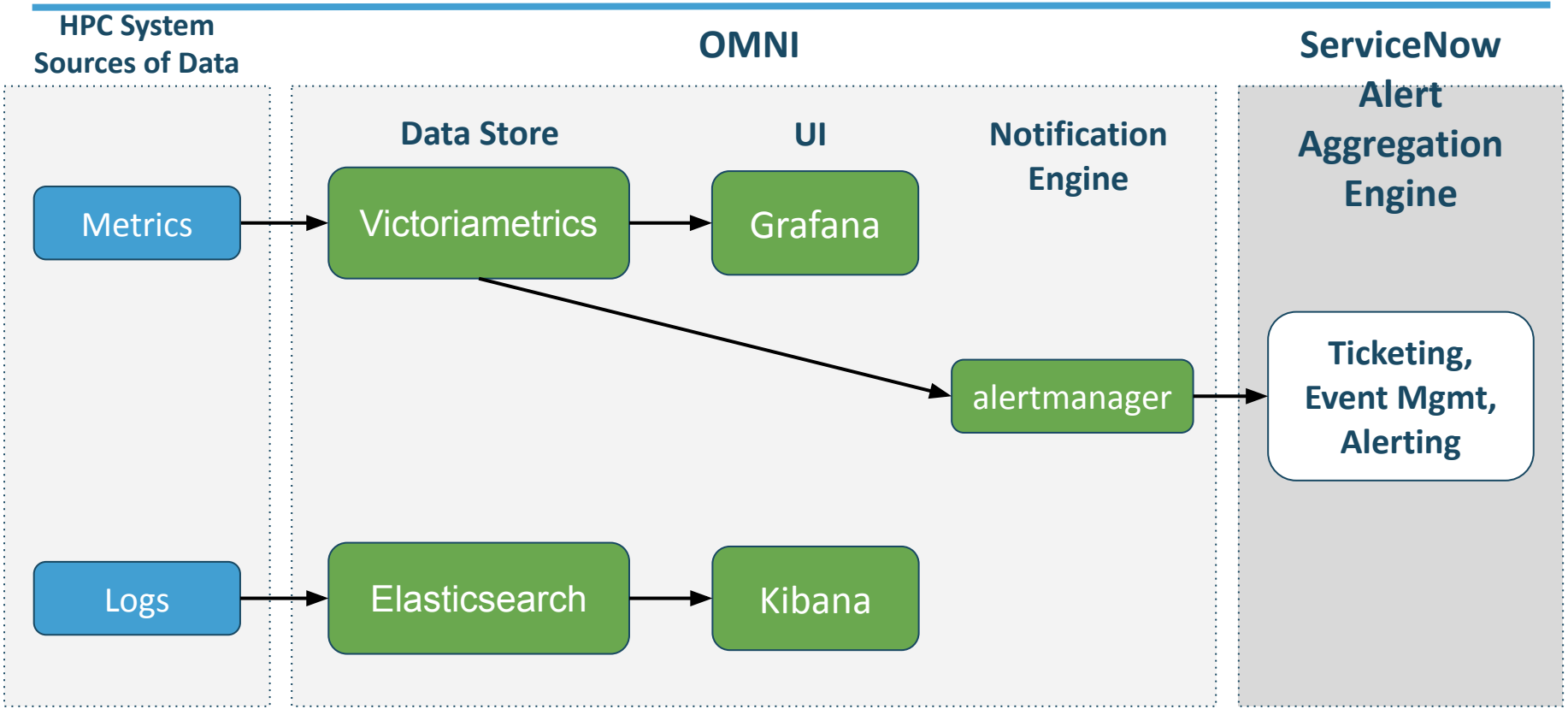
One Query Language

- PromQL/LogQL

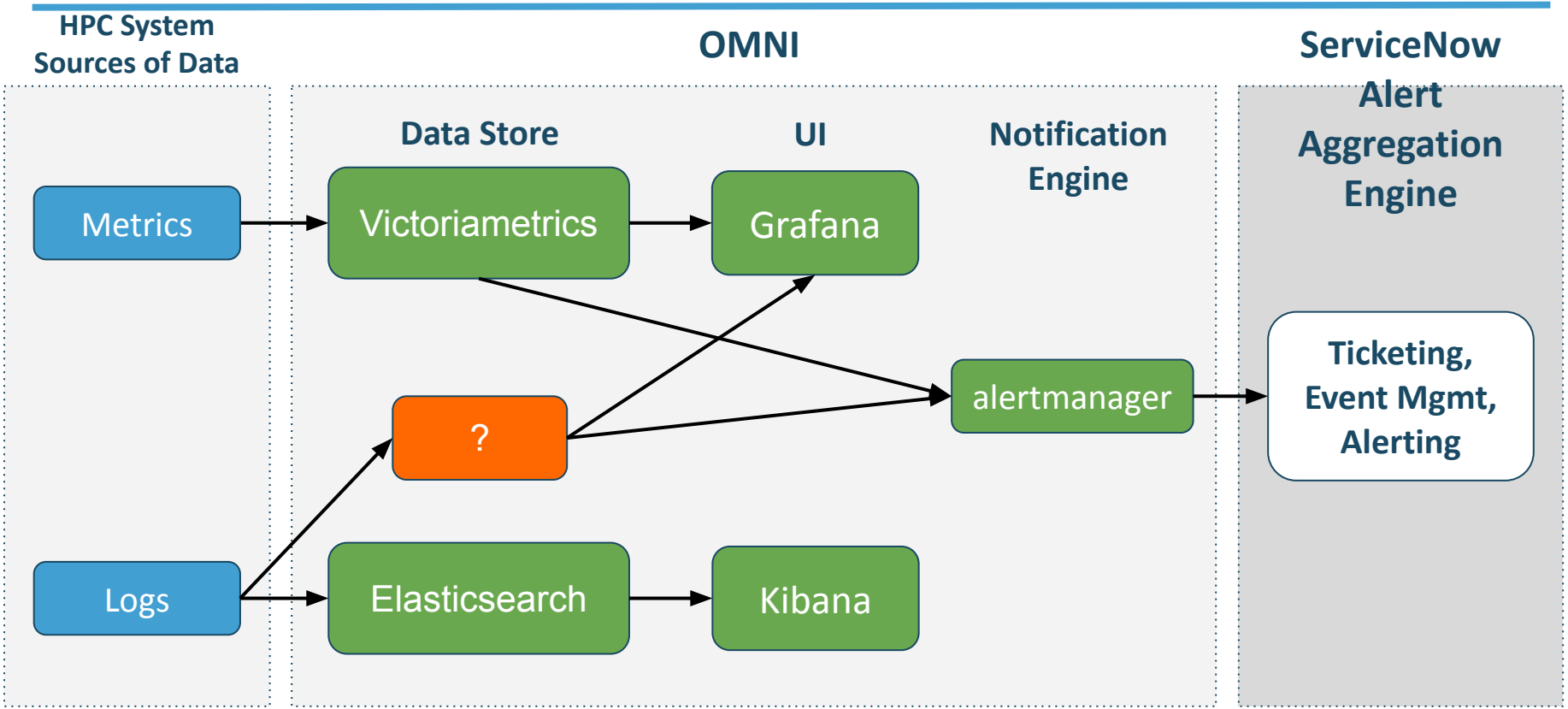
One Notification Engine

- Alertmanager

Data pipeline



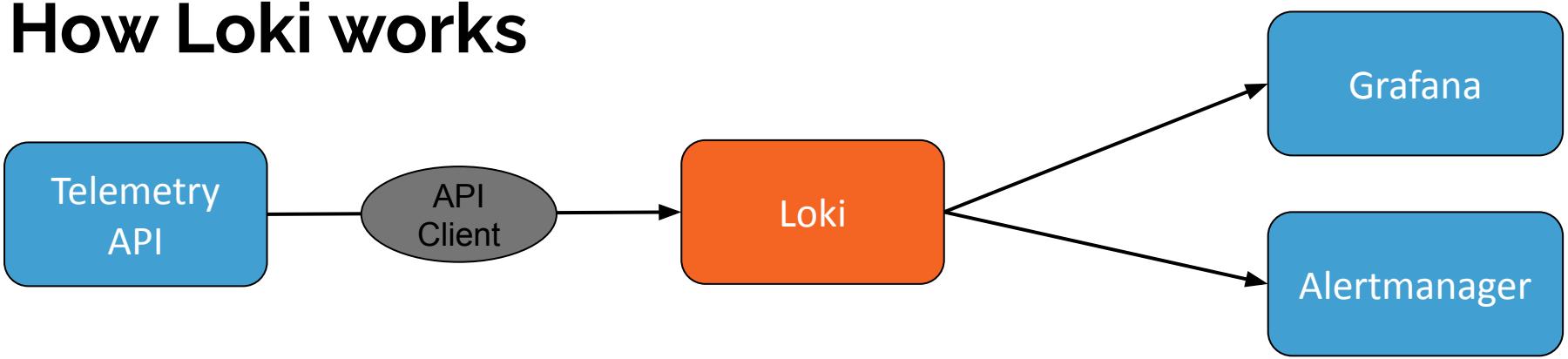
Data pipeline



- A log aggregation system inspired by Prometheus

- Like Prometheus, but for Logs

How Loki works



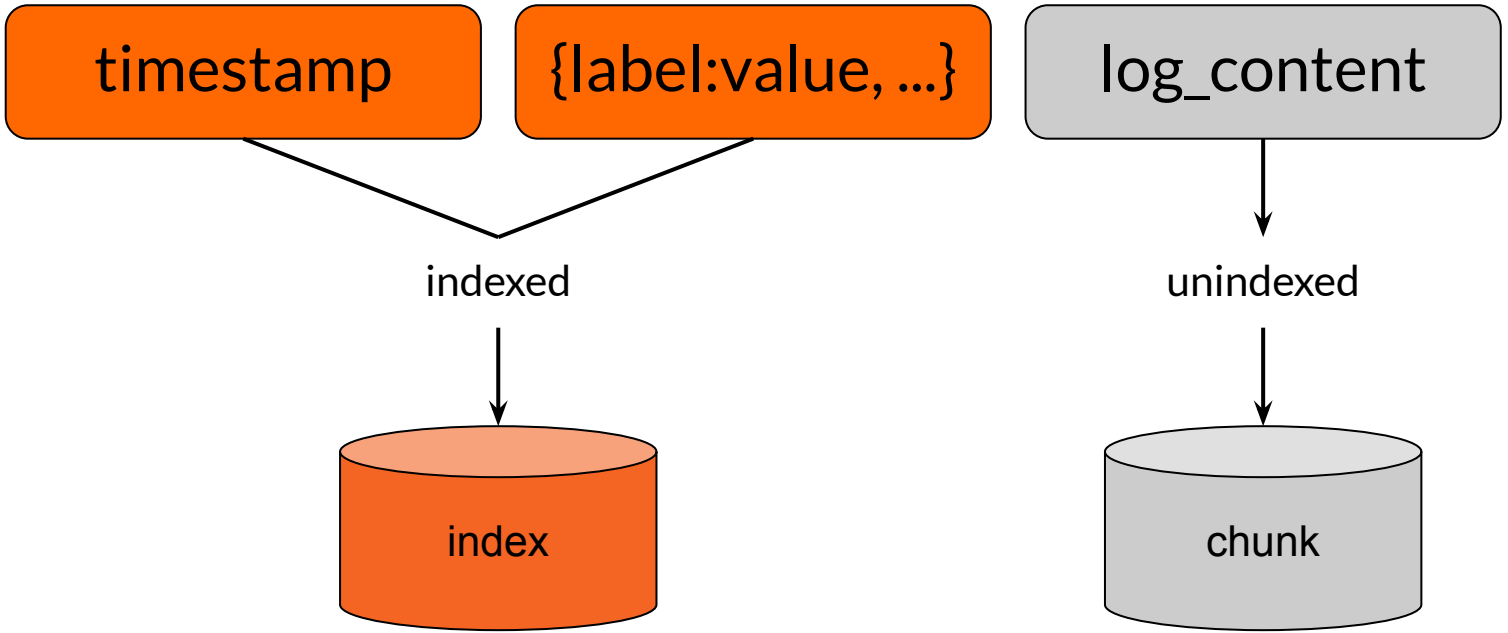

```
2021-08-02T13:58:50Z

{namesapce:"sma",pod:"rsyslog-aggregator-1"...}

rsyslogd: ... error ...
```

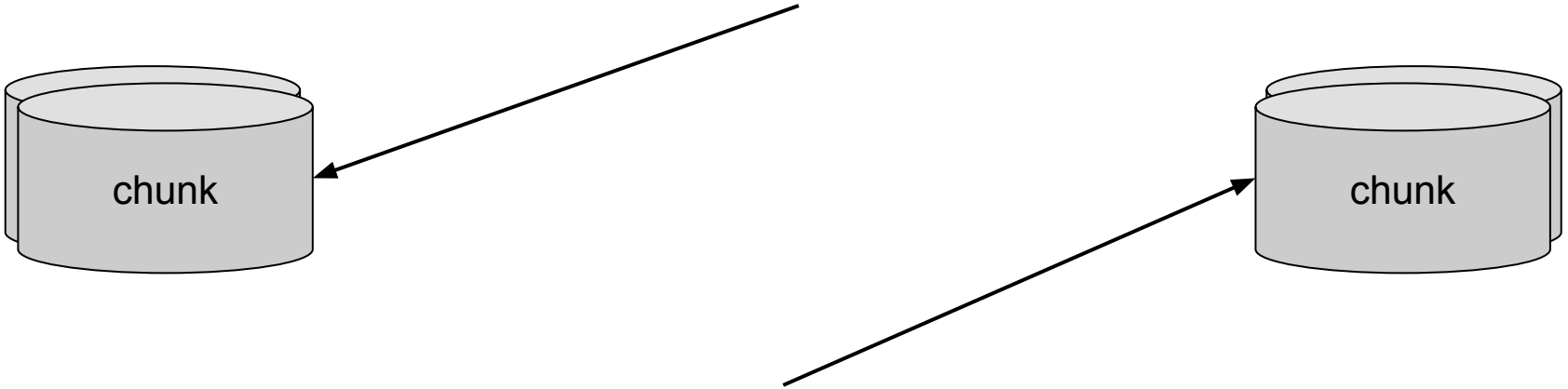
How Loki stores logs

```
2021-08-02T13:58:50Z  {namespace:"sma"...} rsyslogd: ...error...
```



Log streams and chunks

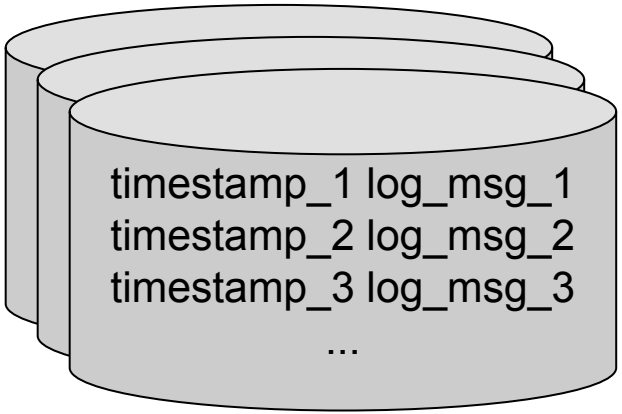
```
2021-08-02T13:58:50Z {namespace:"sma",pod:"rsyslog-aggregator-1"...} log_msg 1
2021-08-02T13:58:51Z {namespace:"sma",pod:"rsyslog-aggregator-1"...} log msg 2
2021-08-02T13:58:52Z {namespace:"sma",pod:"rsyslog-aggregator-1"...} log_msg 3
```



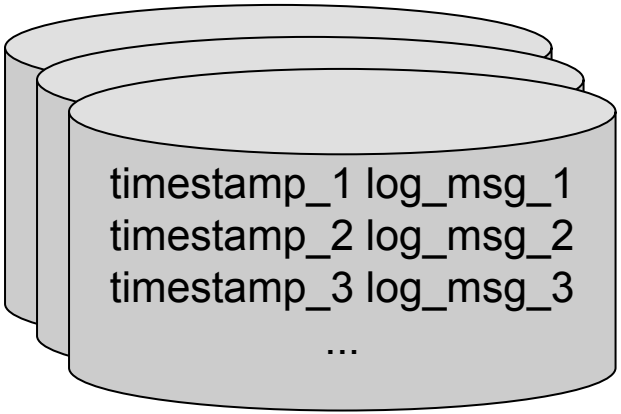

```
2021-08-02T13:57:10Z {namespace:"sma",pod:"cluster-kafka-0"...} log msg 1
2021-08-02T13:58:20Z {namespace:"sma",”pod:"cluster-kafka-0"...} log msg 2
2021-08-02T13:59:30Z {namespace:"sma",pod:"cluster-kafka-0"...} log_msg 3
```

Chunks

```
{namespace:"sma",
pod:"rsyslog-aggregator-1"...}
```




```
{namespace:"sma",
pod:"cluster-kafka-0"...}
```



How a query works

Logcli/Grafana

Label matchers [Filter expressions]

Query:

{namespace="sma"} |= "error"

between timestamp_2

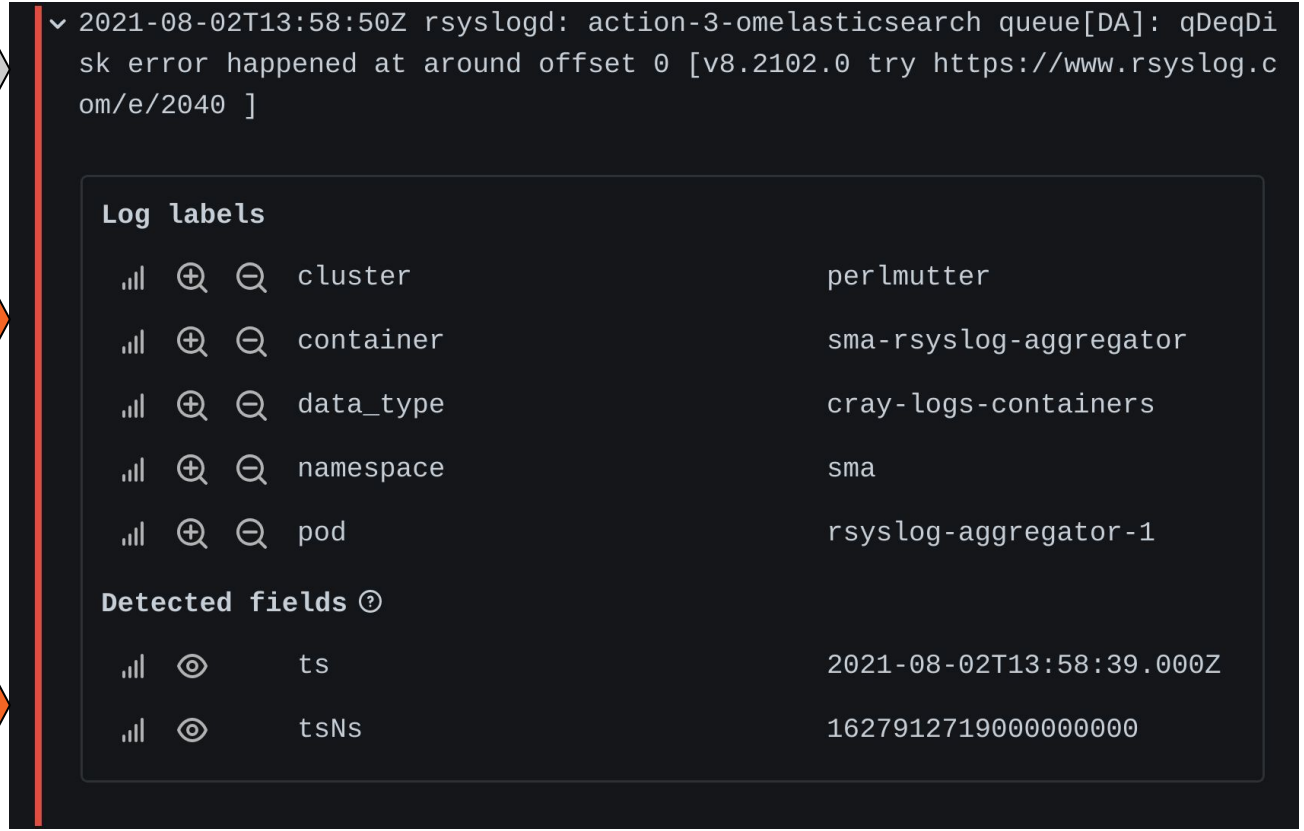
and timestamp_3

How logs look like in Loki/Grafana

log_content

{label:value, ...}

timestamp



Selecting log streams with LogQL

{namespace="sma", pod=~"rsyslog-*"}	|= "error" != "timeout"
Label matchers	Filter expressions

- = exactly equal to a string
- != not equal to a string
- =~ regex matches
- !~ regex does not match

- |= contain a string
- != does not contain a string
- |~ regex matches
- !~ regex does not match

LogQL functions

func({label matchers} filter expressions [time range])

Convert logs into metrics

count_over_time({pod="rsyslog-aggregator-1"}[10m])

- rate(log-range): calculates the number of entries per second
- absent_over_time(log-range): returns an empty vector if the range vector passed to it has any elements and a 1-element vector with the value 1 if the range vector passed to it has no elements

LogQL functions

Same functions from PromQL

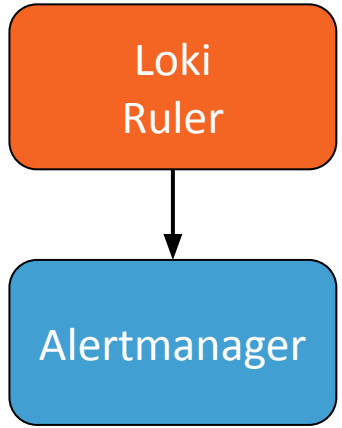
```
sum(count_over_time({namespace="sma"}[1m])) by (pod)
```

avg, max, min, topk, bottomk, etc ...

Alerting Rules in Loki

Query:

sum(count_over_time({hostname=~"ncn-.+"} |= "error"[1m])) by (hostname)



Alerting Rule:

```
- alert: HighLogErrorRate
  expr: sum(count_over_time({hostname=~"ncn-.+"} |= "error"[1m])) by (hostname) > 100
  for: 5m
```

Like Prometheus

- Has native support in Grafana
- Use the same labels
- Use PromQL-inspired language for queries
- Has native support for Alertmanager

What if

One Monitoring UI

- Grafana

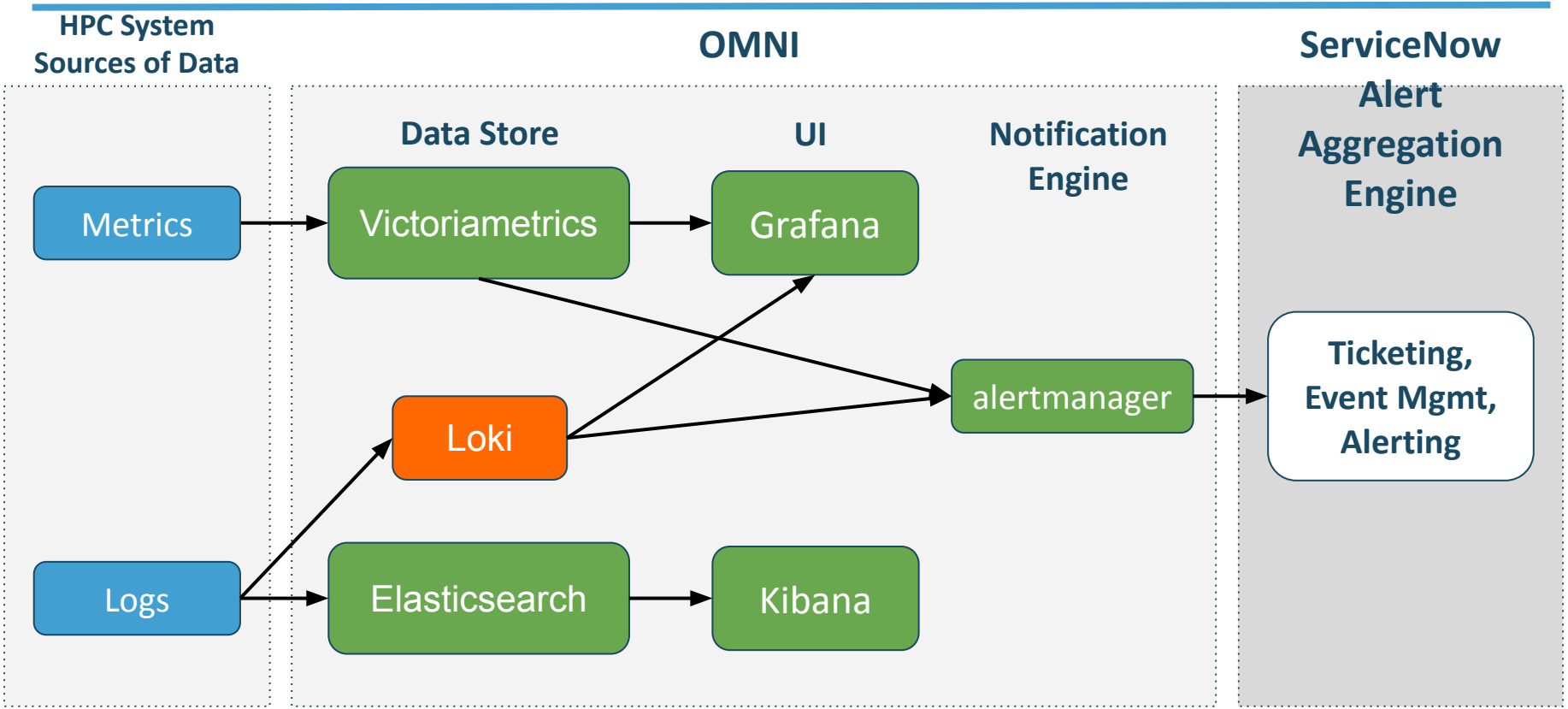
One Query Language

- PromQL/LogQL

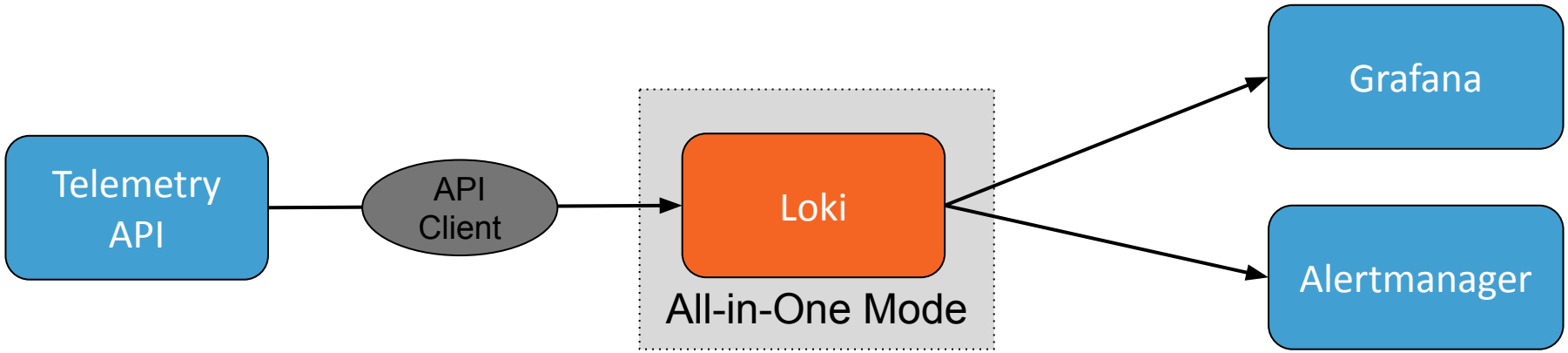
One Notification Engine

- Alertmanager

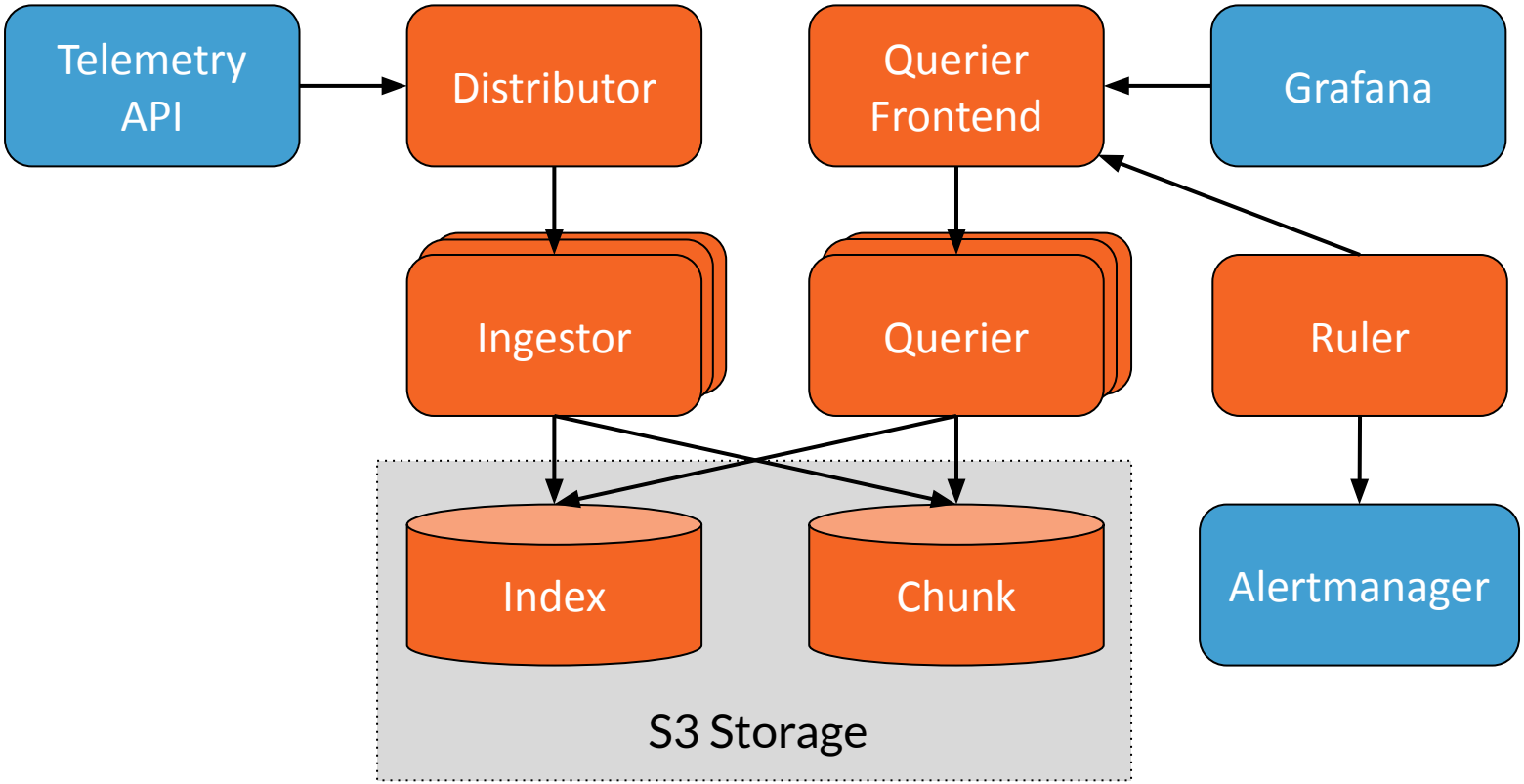
Data pipeline



Performance



Distributed Mode



Reducing Labels

Raw Data from Telemetry API:

{ "procid":"5374",
"timereported":"2021-06-11T03:41:15",
"message":" Auth returned 1:",
"hostname":"ncn-s017",
"tag":"python3[5374]",
"priority":"30",
"severity":"info", "
facility":"daemon" }

Log for Loki:

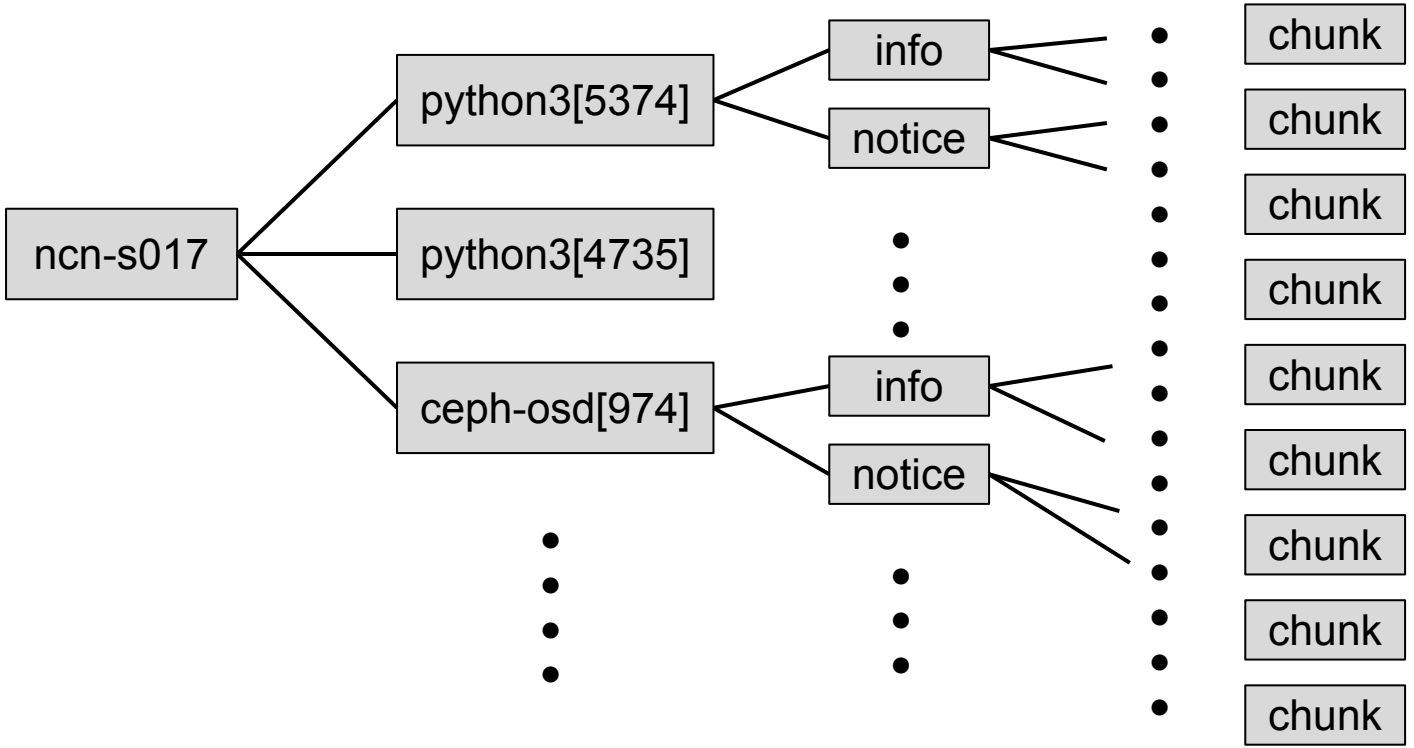
2021-06-11T03:41:15

{hostname :"ncn-s017",
tag:"python3[5374]",
priority:"30",
Severity:"info",
"facility":"daemon" }

" Auth returned 1:"

Overusing Labels

{hostname :"ncn-s017", tag:"python3[5374]", priority:"30", Severity:"info", "facility":"daemon" }



Reducing Labels

Before:

2021-08-11T03:41:15

{hostname :"ncn-s017",
tag:"python3[5374]",
priority:"30",
Severity:"info",
"facility":"daemon" }

" Auth returned 1:"

After:

2021-08-11T03:41:15

{hostname :"ncn-s017"}

"tag=python3[5374] priority=30
severity=info facility=daemon
Auth returned 1:"

Query: {hostname :"ncn-s017"} |= "tag=python3" |= "severity=info"

Connecting logs and metrics

Demo

For

- Ad-hoc analysis
- Short-term storage
- Visualization
- Alerting

Not for

- Replacing Elasticsearch

Beyond Shasta

Visualize & Monitor all logs with Loki

- **Logs from containers and hosts**
- **Syslog from Rosetta switches**
- **Syslog from Aruba switches**
- **Logs from other systems in CRT**