Deploy, Scale and Extend Jaeger

Louis-Etienne Dorval

CloudNativeCon Europe 2019 - Barcelona, Spain

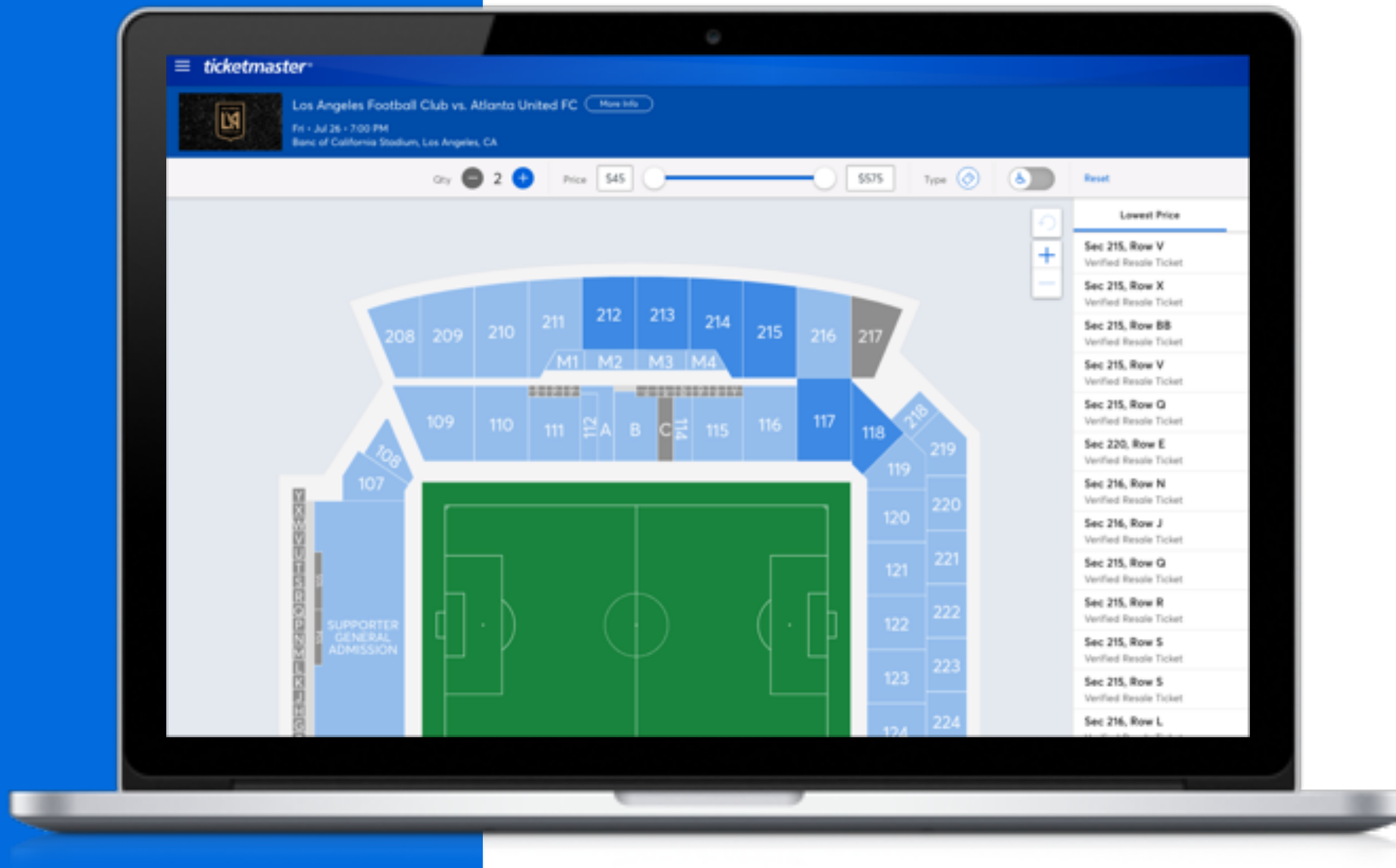
ticketmaster

What is Ticketmaster?

Power unforgettable moments of joy

What is Ticketmaster?

On sale



What is Ticketmaster?

Entry

Behind the scene

40 years of innovation

Hybrid Cloud

20,000+ VMs

7 data centers

15,000+ network endpoints

27 Ticketing Systems and over 300 products

1,400+ peoples in Product and Tech

What is Ticketmaster?

Technology

Custom VAX OS on Emulated VAX

What is Ticketmaster?

Technology

Custom VAX OS on Emulated VAX in Kubernetes (minikube)

Technology

Java

Perl

C

NodeJS

Rust

Go

C#

And probably more!

JAEGER

Jaeger

Current Scale

55+ services instrumented

Java
NodeJS
Go
C#
Envoy (C++)

Up to 8,000 spans per second

2.1B spans in Elasticsearch

1.2TB of data in Elasticsearch

Introduction

Why Tracing?

New Search

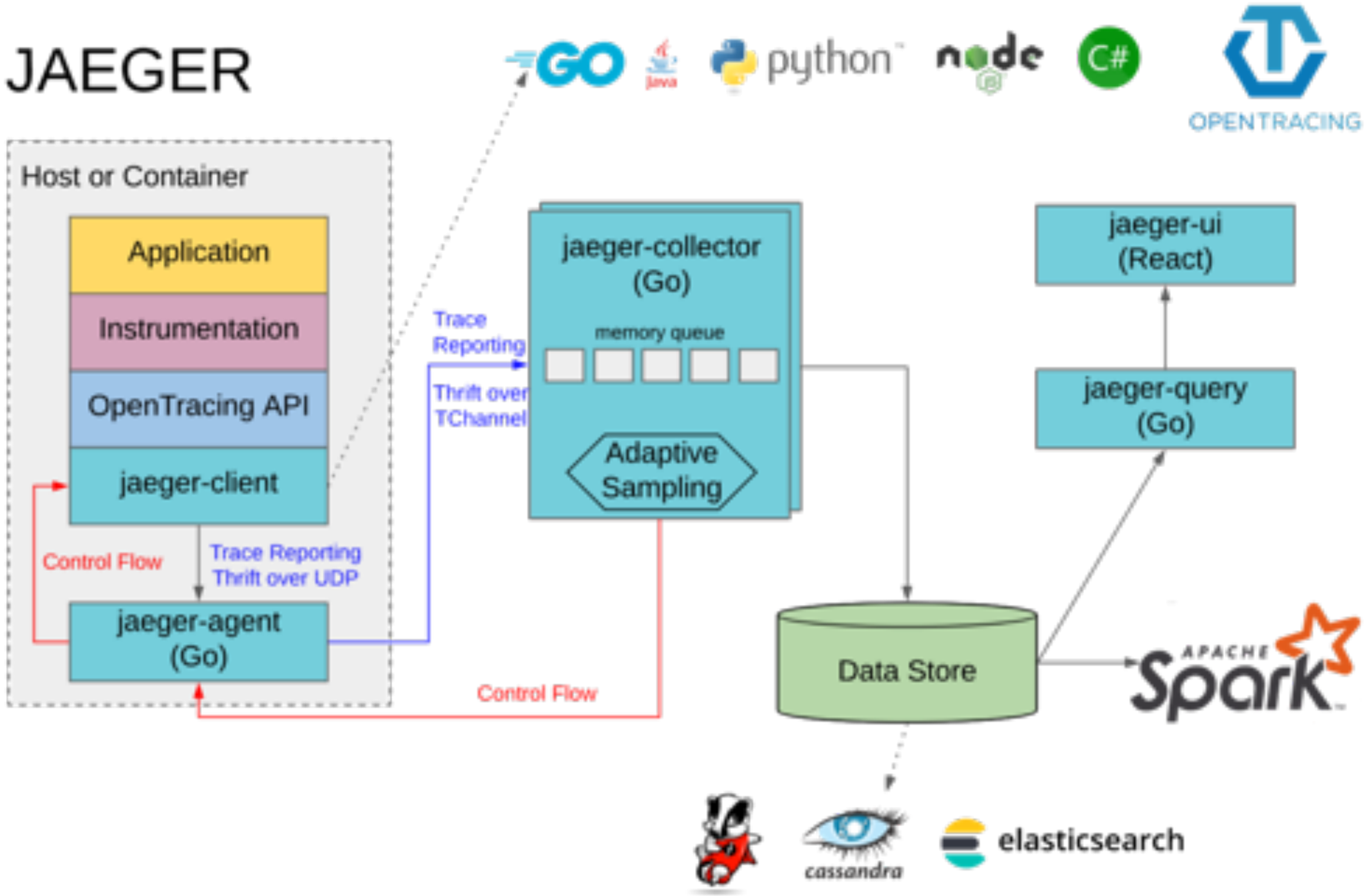
```
(index=[redacted] OR index=[redacted]) NOT(sourcetype=content.[redacted]) NOT(sourcetype=content.[redacted])

| rex field=url ".*[redacted]=(?<updatedBy>.*)&[redacted].*"
| rex field=desc ".*Updating version \[(?<attractionId>.*)\] \[.*"

| eval userClicksOnSave=IF(sourcetype="[redacted]" AND activity="Update and MarkAsReady an attraction" AND  appCode="inbound.request.rest", _time, "")
| eval updatedIn[redacted]=IF(activity="UpdateAttractionVersions" AND appCode="outbound.response.rest" AND sourcetype="objects.[redacted]", _time, "")
| eval publishedBy[redacted]=IF(activity="MarkVersionsAsReady" AND appCode="outbound.response.rest" AND sourcetype="objects.[redacted]", _time, "")
| eval pushbackIn[redacted]=IF(activity="objectTo[redacted]" AND appCode="105" AND sourcetype="objects.[redacted]", _time, "")
| eval indexedInSolr=IF(activity="toSolrConversion" AND sourcetype="content.[redacted]", _time, "")
| eval availableIn[redacted]=IF(activity="PersistUnifiedAttraction" AND sourcetype="content.[redacted]" , _time, "")

| join Correlation_ID [search activity="Update and MarkAsReady an Attraction" appCode="inbound.request.rest" index=auth sourcetype="objects.[redacted]"]
| transaction Correlation_ID

| fieldformat userClicksOnSave= strftime(userClicksOnSave, "%F %T.%3N")
| fieldformat updatedIn[redacted]= strftime(updatedIn[redacted], "%F %T.%3N")
| fieldformat publishedBy[redacted]= strftime(publishedBy[redacted], "%F %T.%3N")
| fieldformat pushbackIn[redacted]= strftime(pushbackIn[redacted], "%F %T.%3N")
| fieldformat indexedInSolr= strftime(indexedInSolr, "%F %T.%3N")
| fieldformat availableIn[redacted]= strftime(availableIn[redacted], "%F %T.%3N")

| table _time, Correlation_ID, attractionId, updatedBy, duration,
        userClicksOnSave, updatedIn[redacted], publishedBy[redacted], pushbackIn[redacted], indexedInSolr, availableIn[redacted]
```

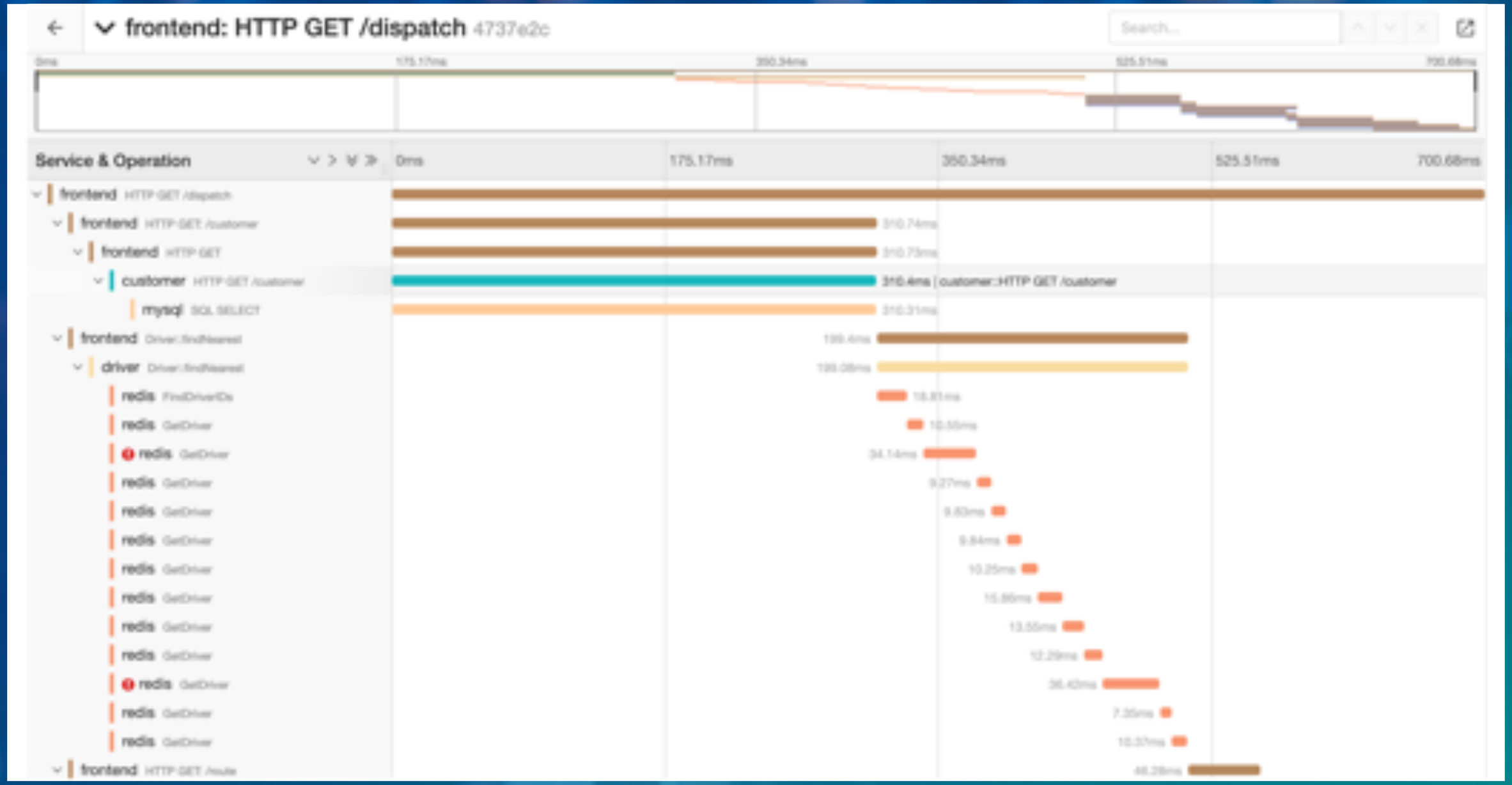
Why Tracing?

- Would You Like Some Tracing With Your Monitoring? - Yuri Shkuro, Uber Technologies
  - CloudNativeCon 2017 (https://sched.co/CU8f)
- Service Meshes and Observability - Ben Sigelman, Lightstep
  - CloudNativeCon 2017 (https://sched.co/CUCX)
- Distributed Tracing in Serverless Systems - Nitzan Shapira, Epsagon
  - CloudNativeCon 2018 (https://sched.co/GrXp)
- Understanding Microservices with Distributed Tracing - Lita Cho, Lyft
  - CloudNativeCon 2018 (https://sched.co/GrXj)

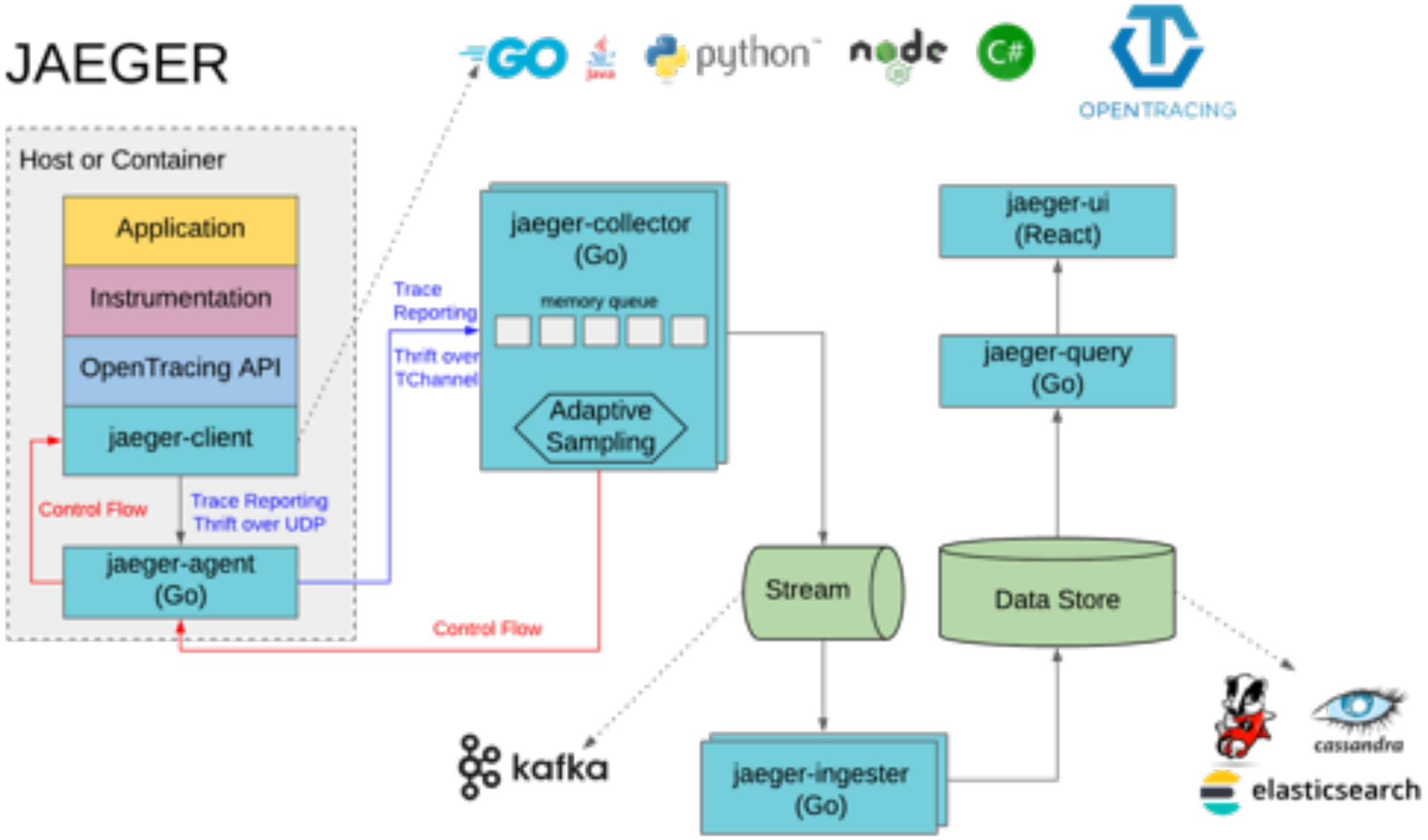
Jaeger Architecture

JAEGER

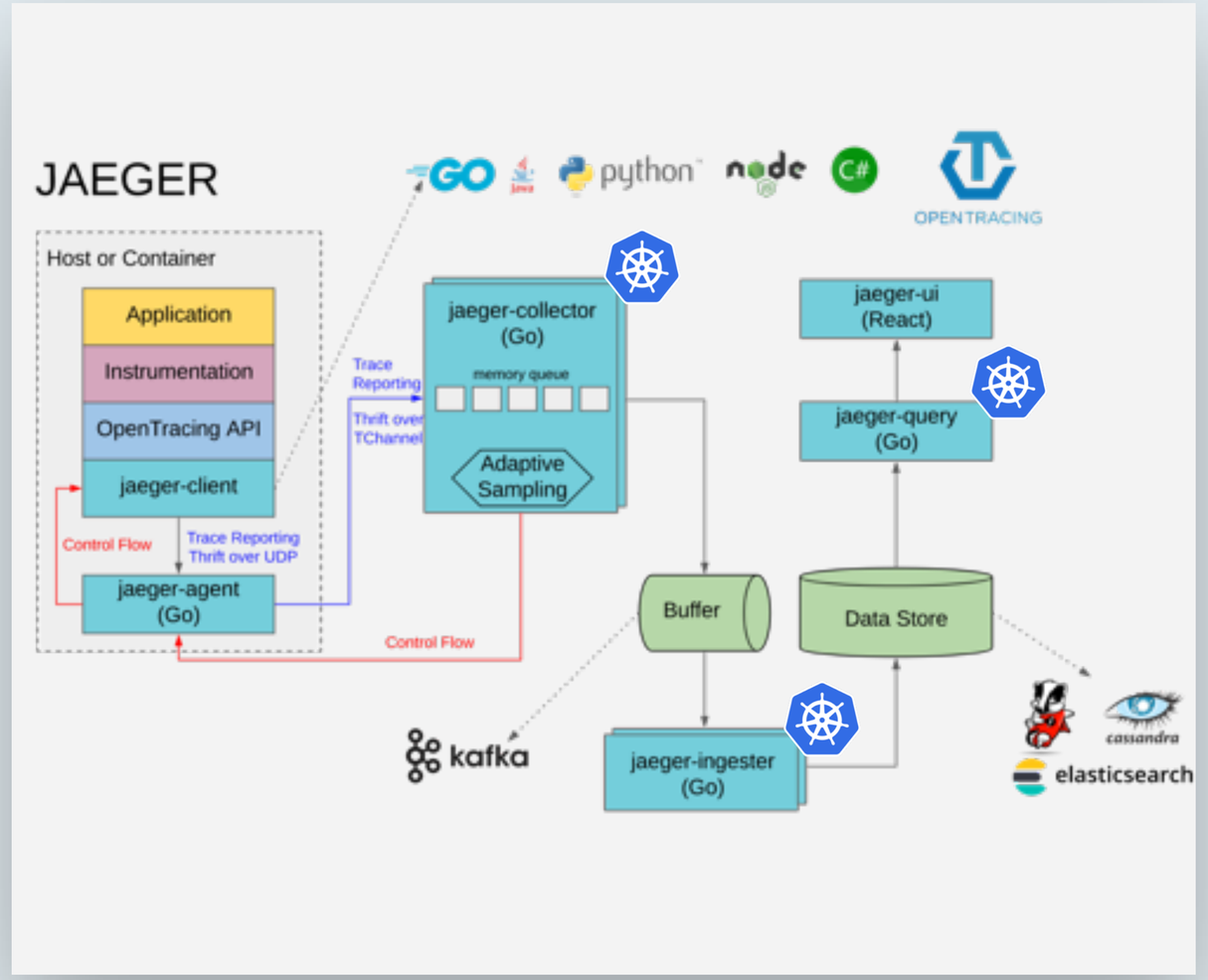

Host or Container
Application
Instrumentation
OpenTracing API
jaeger-client
Control Flow
Trace Reporting
Thrift over UDP
jaeger-agent
(Go)
Trace
Reporting
Thrift over
TChannel
jaeger-collector
(Go)
memory queue
Adaptive
Sampling
Control Flow
jaeger-ui
(React)
jaeger-query
(Go)
Data Store
APACHE
Spark
cassandra
elasticsearch


frontend: HTTP GET /dispatch 4737e2c
Search...
0ms
175.17ms
350.34ms
525.51ms
700.68ms
Service & Operation
frontend HTTP GET /dispatch
frontend HTTP GET /customer
frontend HTTP GET
customer HTTP GET /customer
310.4ms | customer::HTTP GET /customer
mysql SQL SELECT
frontend Driver::findNearest
driver Driver::findNearest
redis FindDriverIDs
redis GetDriver
redis GetDriver
redis GetDriver
redis GetDriver
redis GetDriver
redis GetDriver
redis GetDriver
redis GetDriver
redis GetDriver
redis GetDriver
redis GetDriver
redis GetDriver
frontend HTTP GET /route

JAEGER


Host or Container
Application
Instrumentation
OpenTracing API
jaeger-client
Control Flow
Trace Reporting
Thrift over UDP
jaeger-agent
(Go)
Trace
Reporting
Thrift over
TChannel
jaeger-collector
(Go)
memory queue
Adaptive
Sampling
Control Flow
jaeger-ui
(React)
jaeger-query
(Go)
Stream
Data Store
kafka
jaeger-ingester
(Go)
cassandra
elasticsearch

Deploy Jaeger



Deployment

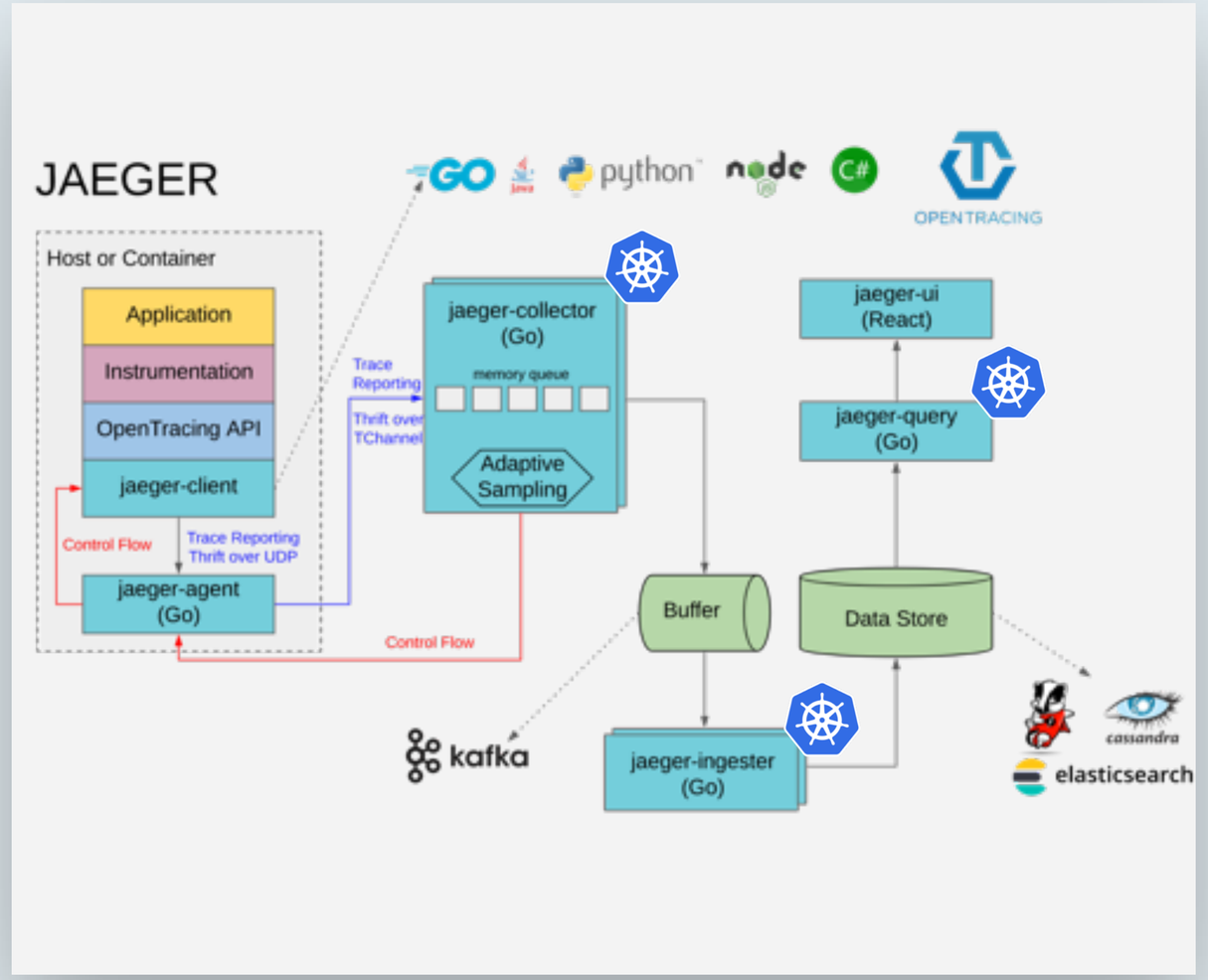
Kubernetes

Deployment (apps/v1)

- Jaeger Collector
- Jaeger Query
- Jaeger Ingester

CronJob (batch/v1beta1)

- es-index-cleaner
- spark-dependencies



Deployment

Kubernetes

Service (v1) - Type LoadBalancer

- Jaeger Collector
  - ELB in AWS

Ingress (extensions/v1beta1)

- Jaeger Query
  - ALB in AWS

Deployment

Jaeger Agent

Linux and Windows VM
- Binary → Daemon

Kubernetes
- Docker → Sidecar

AWS EC2
- Docker → Sidecar
  (docker run --link)

1. If the the Jaeger Client can't reach the Jaeger-Agent on localhost, you need to set:
   a. JAEGER_AGENT_HOST
   b. JAEGER_SAMPLER_MANAGER_HOST_PORT

2. The value of the 1b above vary from one language to another
   [jaeger-client-go #362]

Sidecar

⊕ Isolation

Security

⊖ Resource utilization

Added complexity outside Kubernetes 

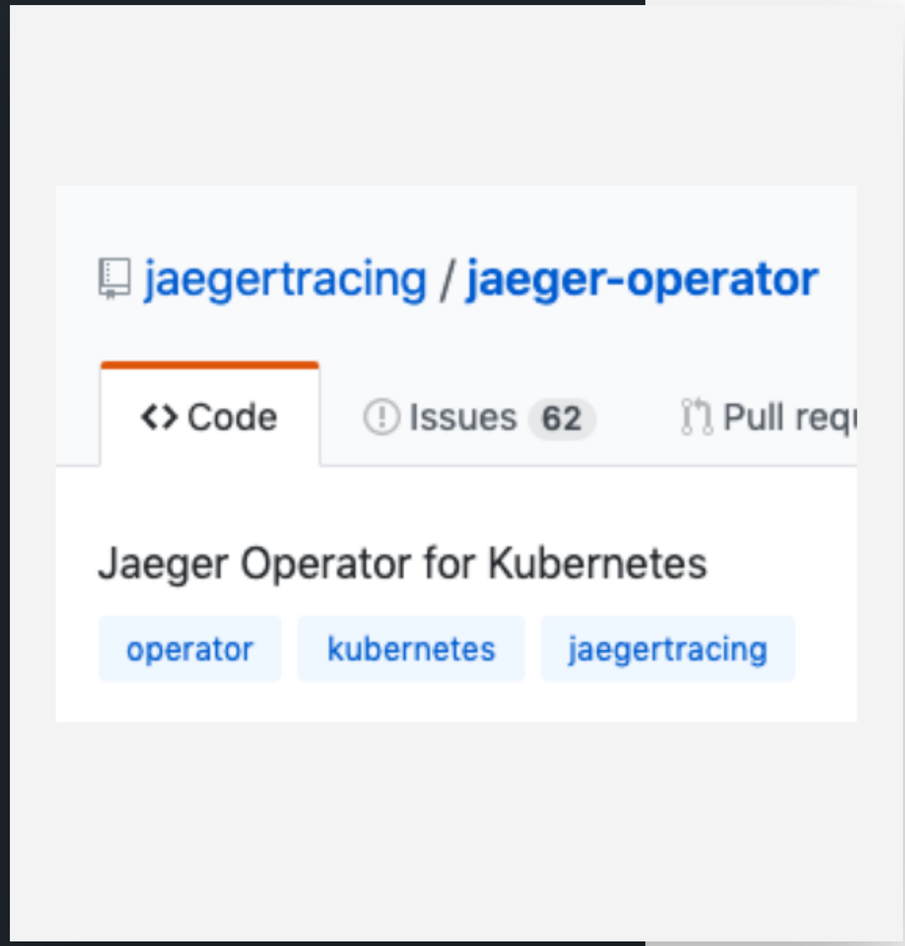
Daemon

⊕ Resource utilization

Added complexity inside Kubernetes 

⊖ Isolation

Security



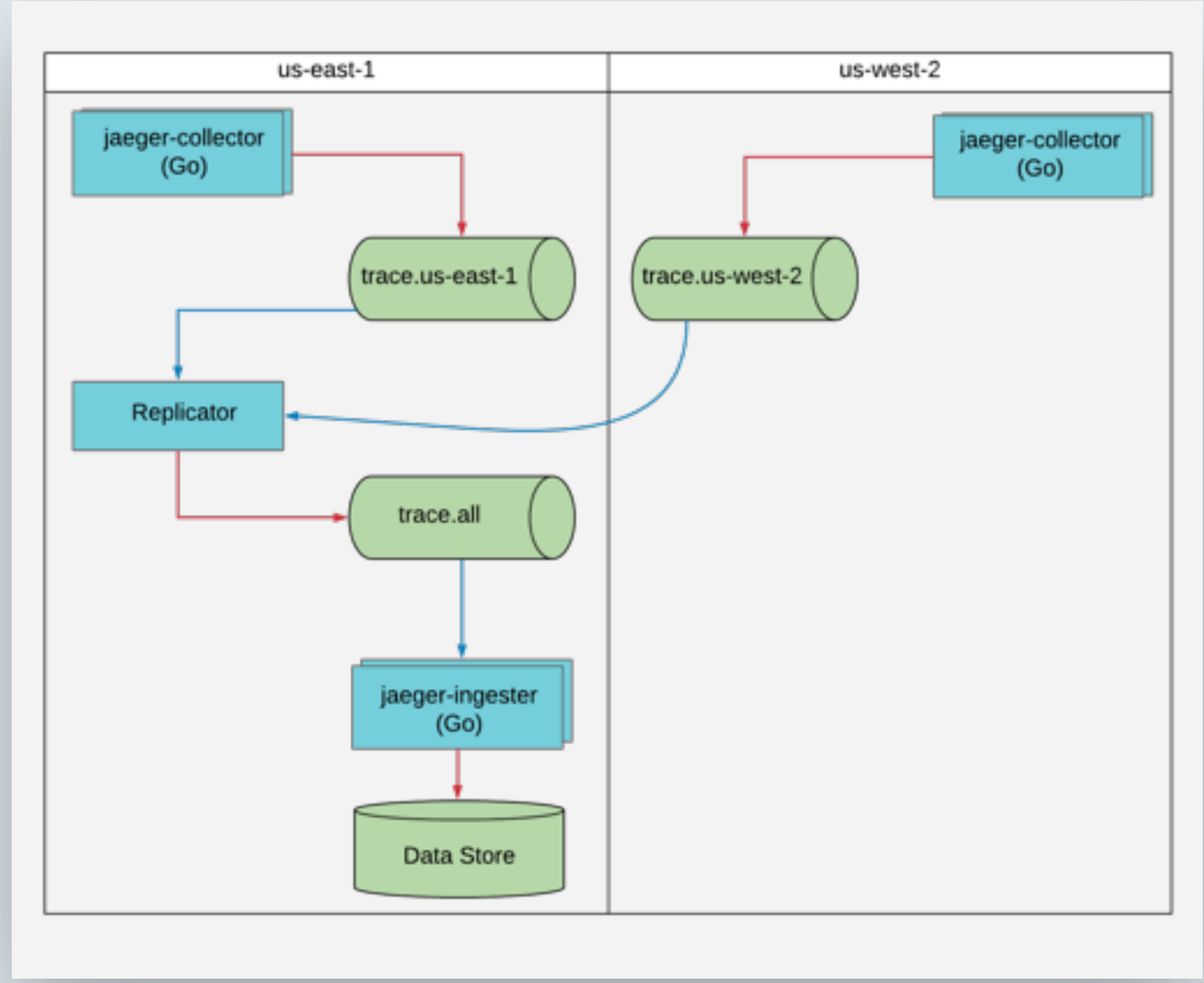
Deployment

Jaeger Operator for Kubernetes

- Deployment strategies
  - All In One
  - Production
  - Streaming

- Jaeger Agent strategies
  - Sidecar (auto injection supported)
  - DaemonSet

- Version upgrades

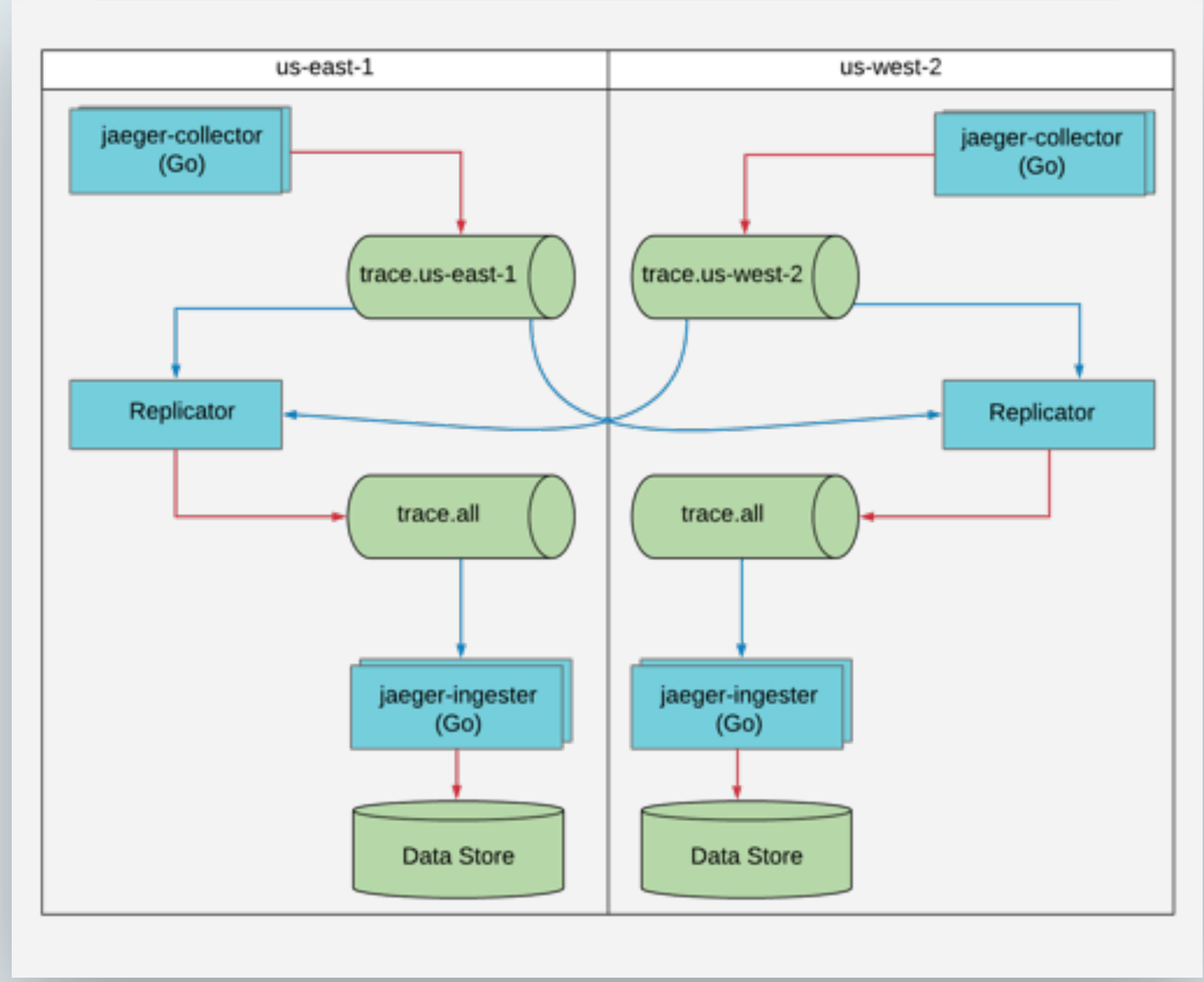


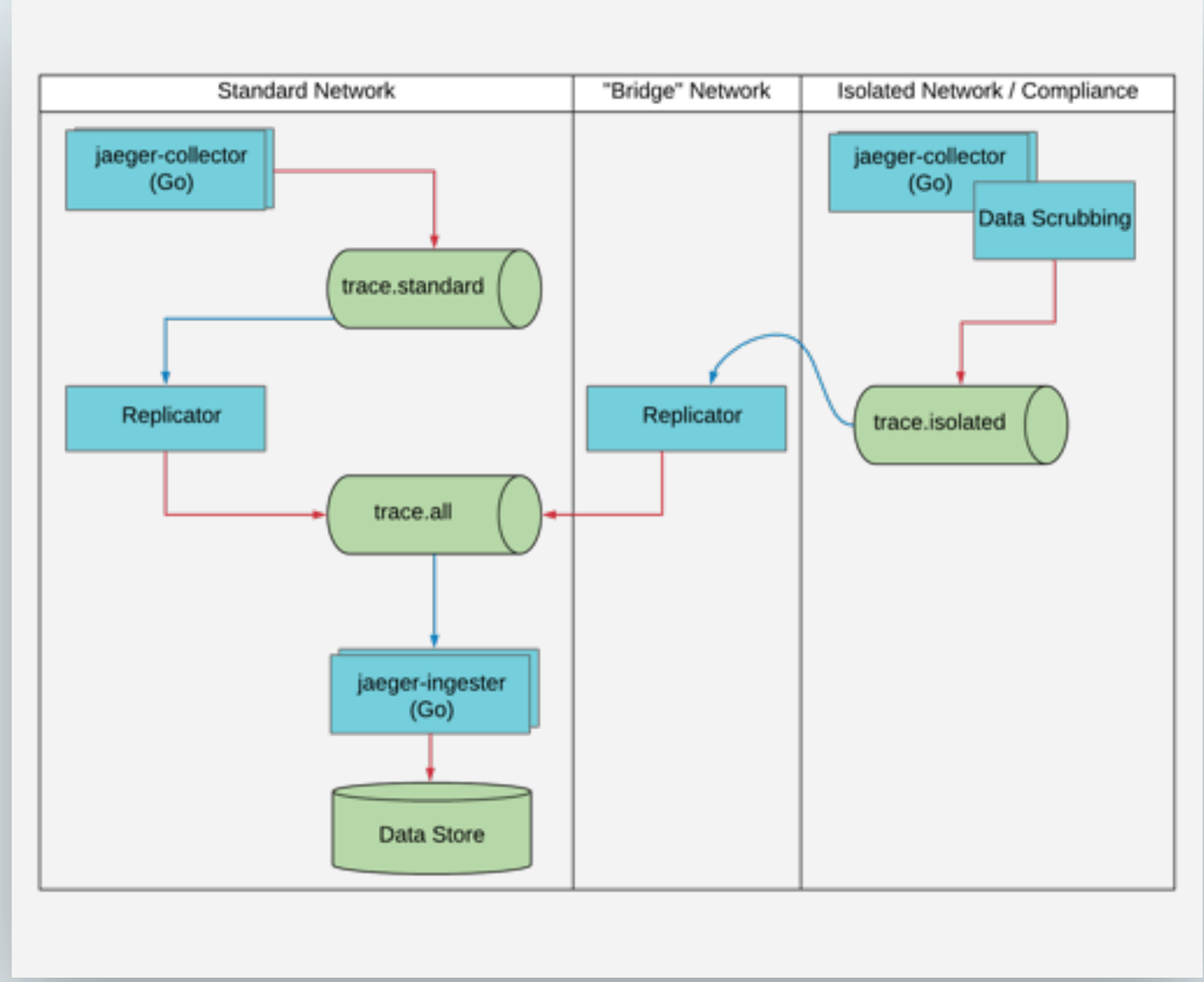
Deployment

Multi-Region (single datastore)

Deployment

Multi-region (multiple datastores)



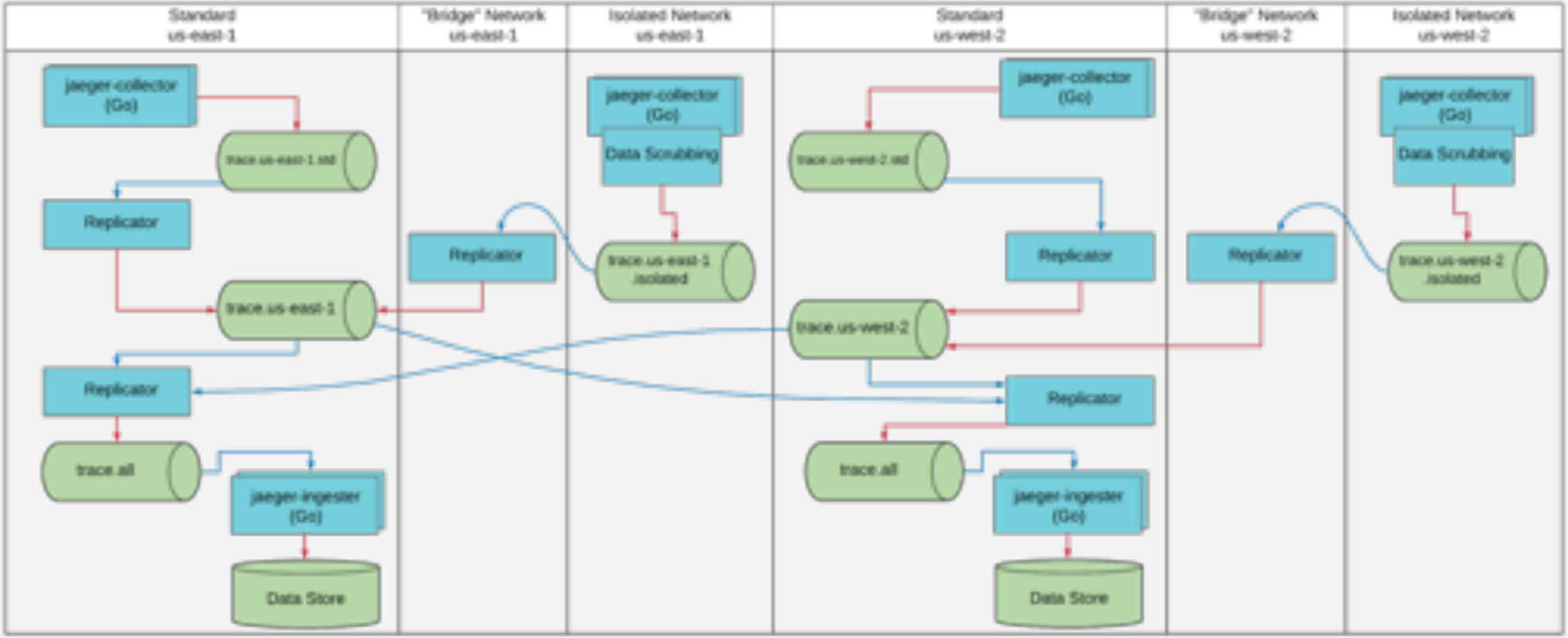


Deployment

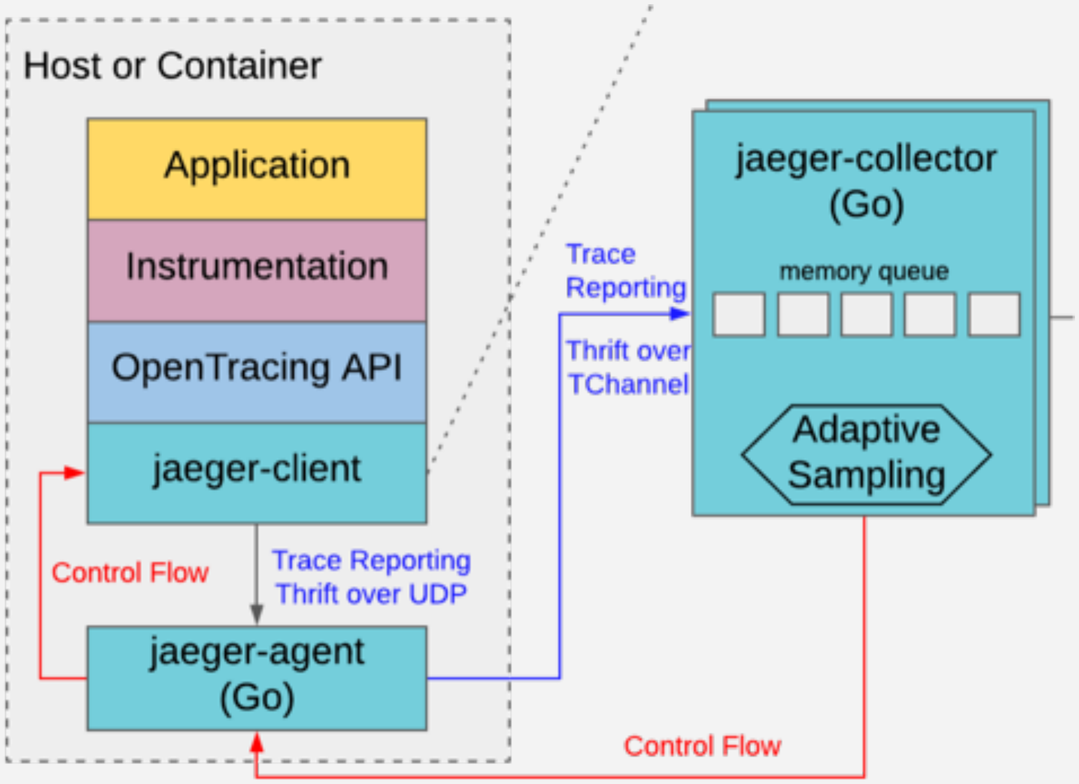
Isolated network / Compliance

Deployment

Kafka



Scale Jaeger



Scaling

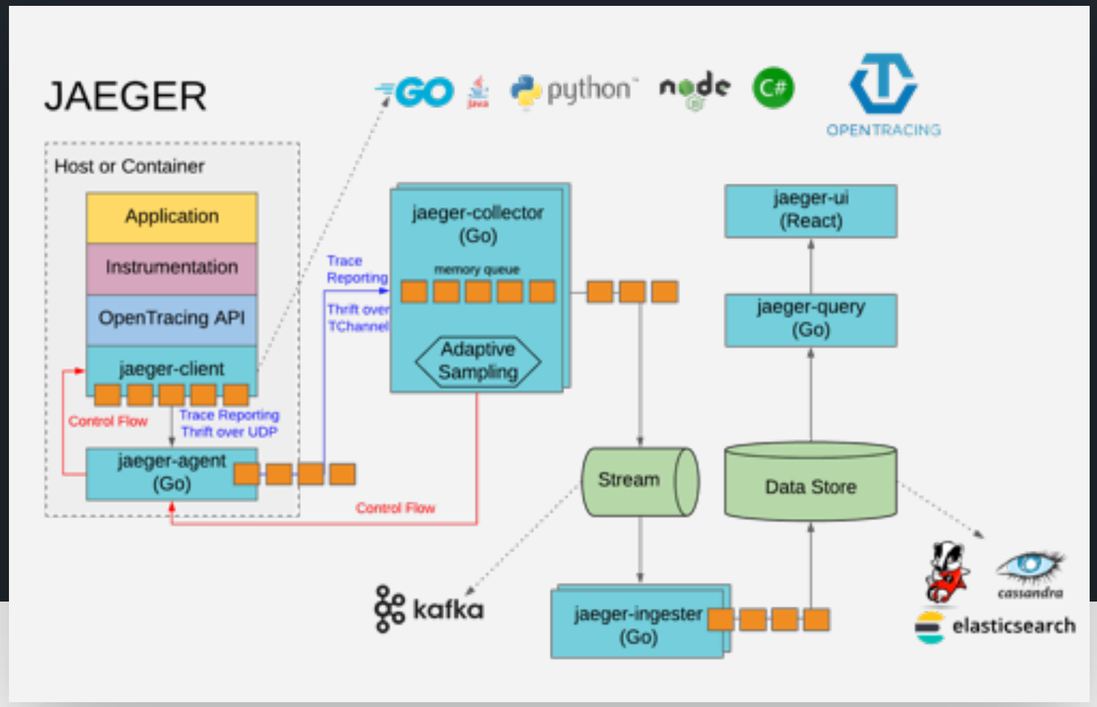
Start Small with Sampling

```
{
  "default_strategy": {
    "type": "ratelimiting",
    "param": 1
  }
}
```

Scale

Queues

https://medium.com/jaegertracing/tuning-jaegers-performance-7a60864cf3b1



Jaeger Agent

--processor.*.server-queue-size

Jaeger Collector

--collector.queue-size

--es.bulk.*

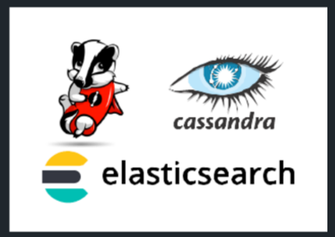
Kafka AsyncProducer
(based on max.message.bytes)

Jaeger Ingester

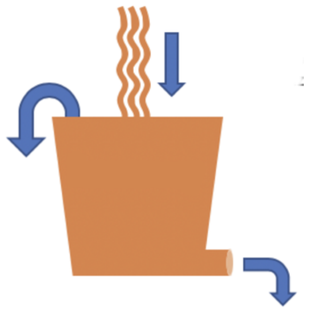
--collector.queue-size

--es.bulk.*

Monitoring - Collector / Ingestor



jaeger_collector_
spans_saved_
by_svc_total
{result="err”}

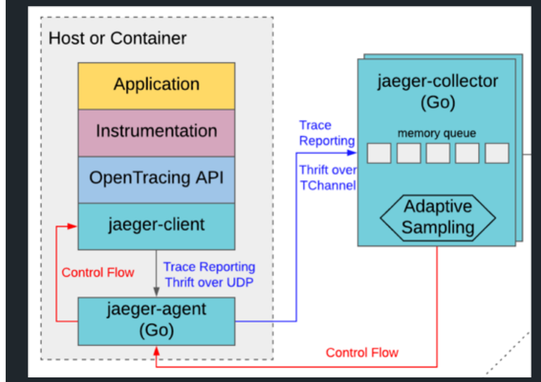


jaeger_collector_
spans_dropped_total

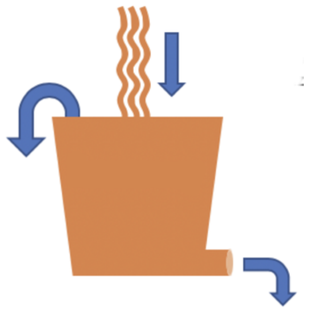

```
{
  "level":"error",
  "caller":"config/config.go:121",
  "msg":"Elasticsearch part of bulk request failed",
  "response":{
    "status":400,
    "error":{
      "caused_by":{
        "reason":"Numeric value (18446744073709343616) out of range of long
        "type":"json_parse_exception"
      }
    }
  }
```


Logs!

Monitoring - Agent



jaeger_agent_
collector_proxy_total
{result="err"}



jaeger_agent_*_
dropped_total


```
{
  "level":"error",
  "ts":1557538276.4659522,
  "caller":"peerlistmgr/peer_list_mgr.go:171",
  "msg":"Unable to connect",
  "host:port":"<...>",
  "connCheckTimeout":0.25,
  "error":"tchannel error ErrCodeTimeout: timeout"
  "stacktrace":"github.com/jaegertracing/jaeger/pk
```


Logs!

And...

Scaling

Current infrastructure

Elasticsearch

3x i3.xlarge

➔4 CPUs

➔30.5 GiB

➔950 GB (NVMe SSD)

Jaeger Collector

4x Pods

➔2 CPUs

➔4 GiB

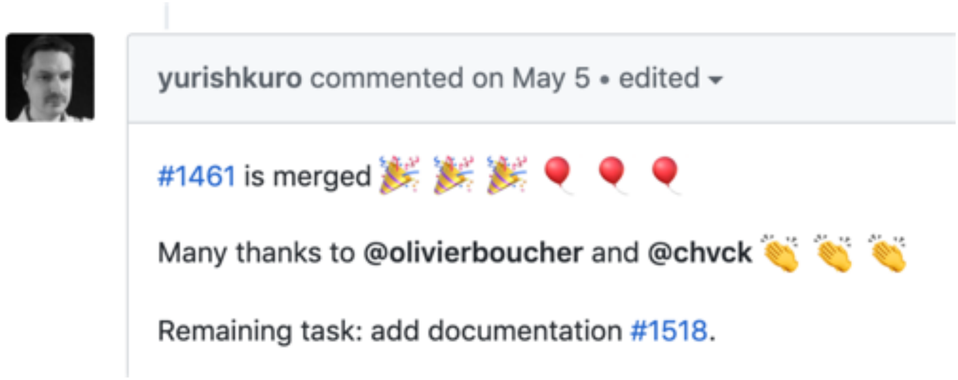
➔Queue Size: 300,000

➔Workers: 50 (default)

Extend Jaeger

Extend

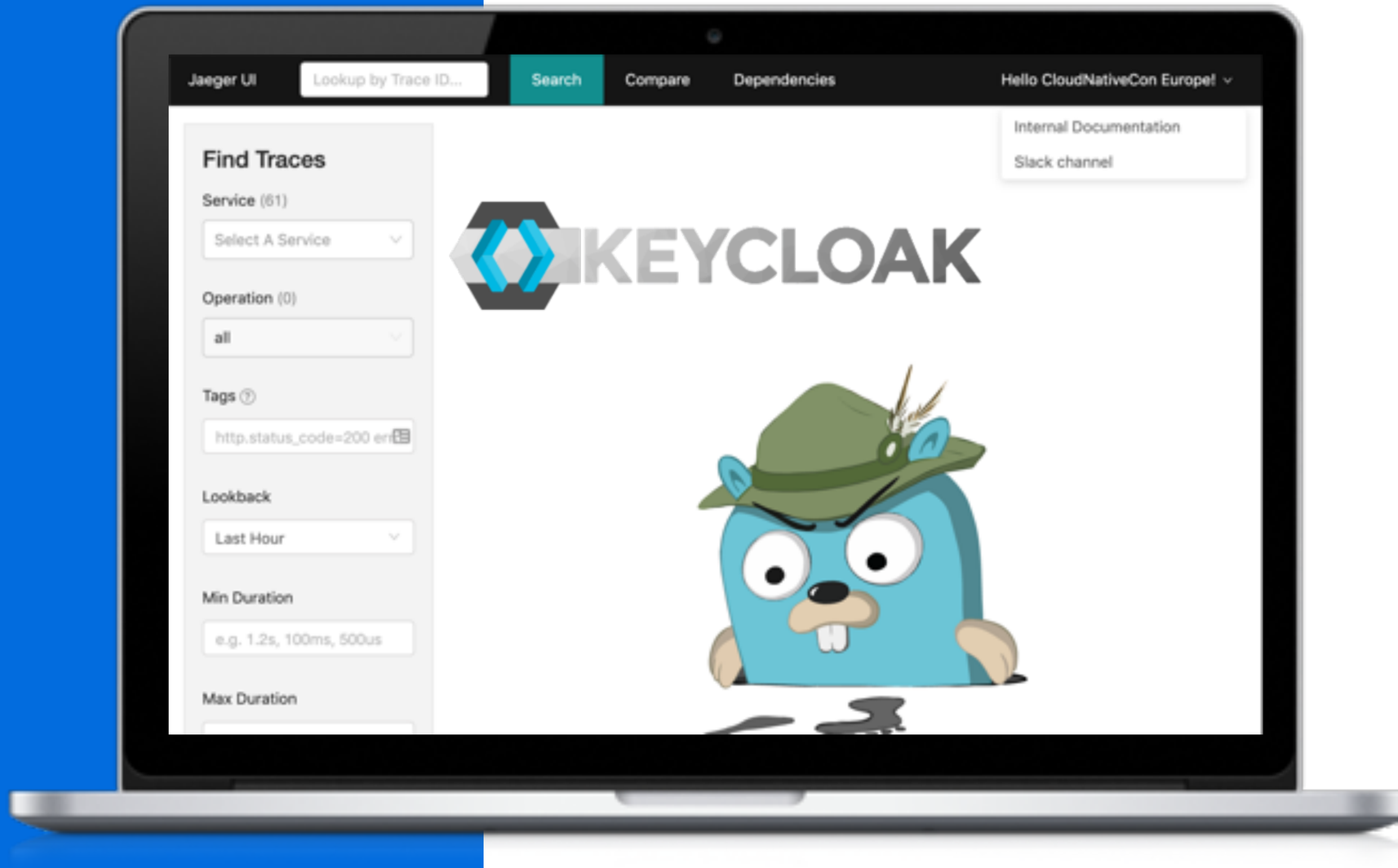
Storage plugin



Extend

Protect Jaeger UI

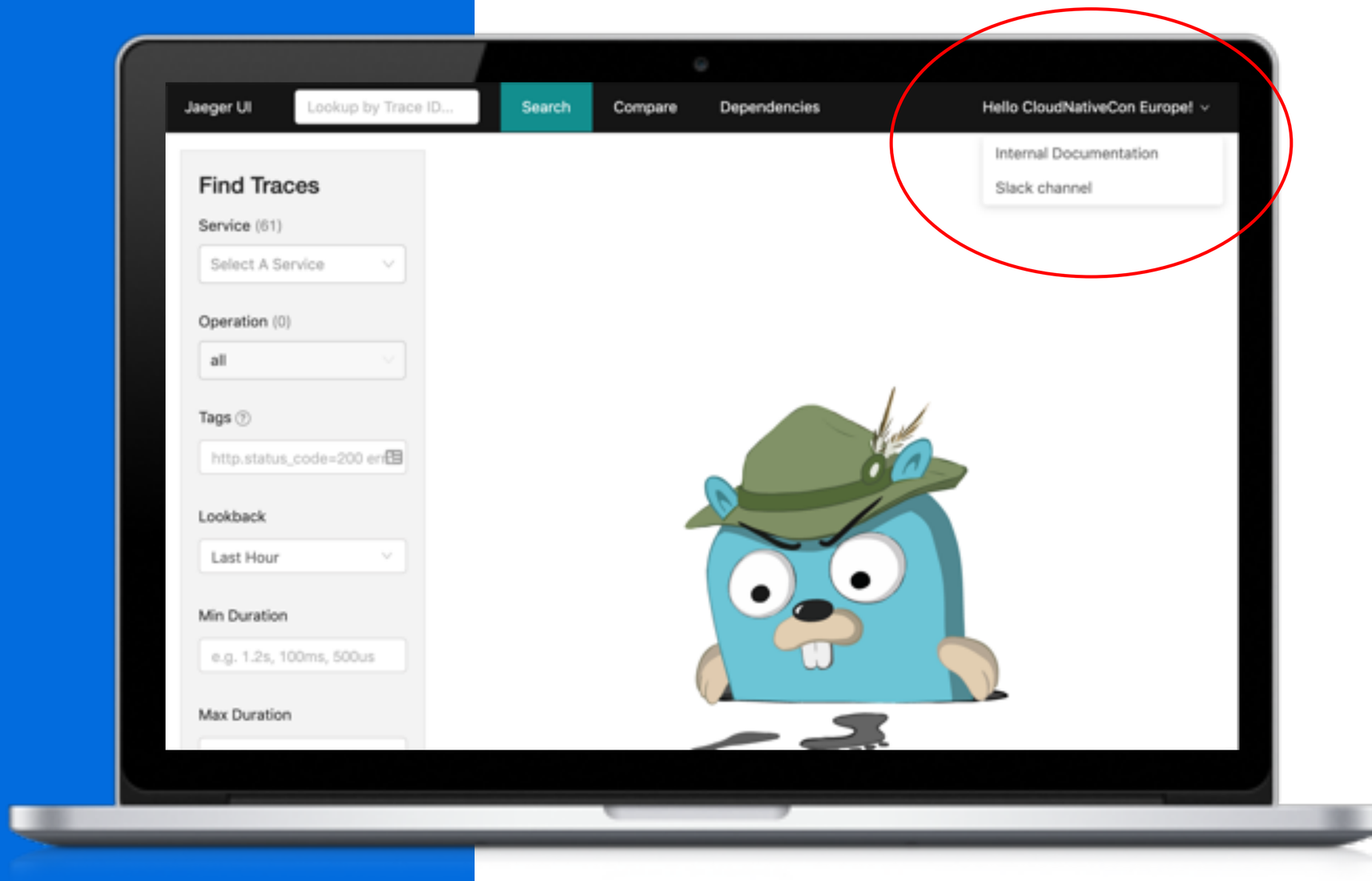
https://medium.com/jaegertracing/protecting-jaeger-ui-with-an-oauth-sidecar-proxy-34205cca4bb1



Extend

Custom Menu

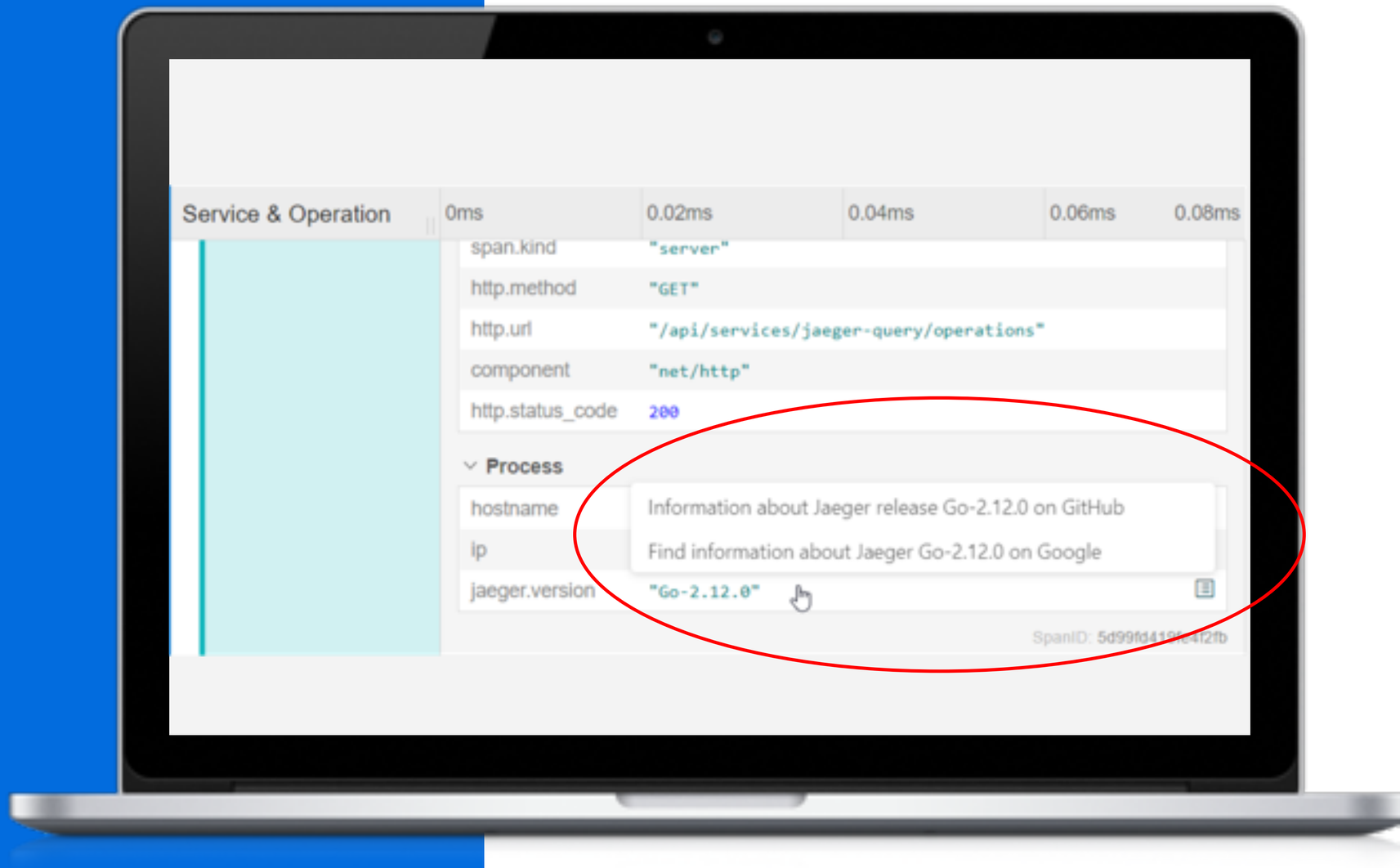
- ➔ Internal documentation
- ➔ Support Slack channel
- ➔ Add a Logout link
  - ◆ jaegertracing/jaeger-ui/pull/223
- ➔ Architecture diagram
- ➔ PagerDuty information



Extend

Links on Tag

- ➔ Inventory Asset Management
  - ◆ OnCall information
  - ◆ Slack channel
  - ◆ Source code

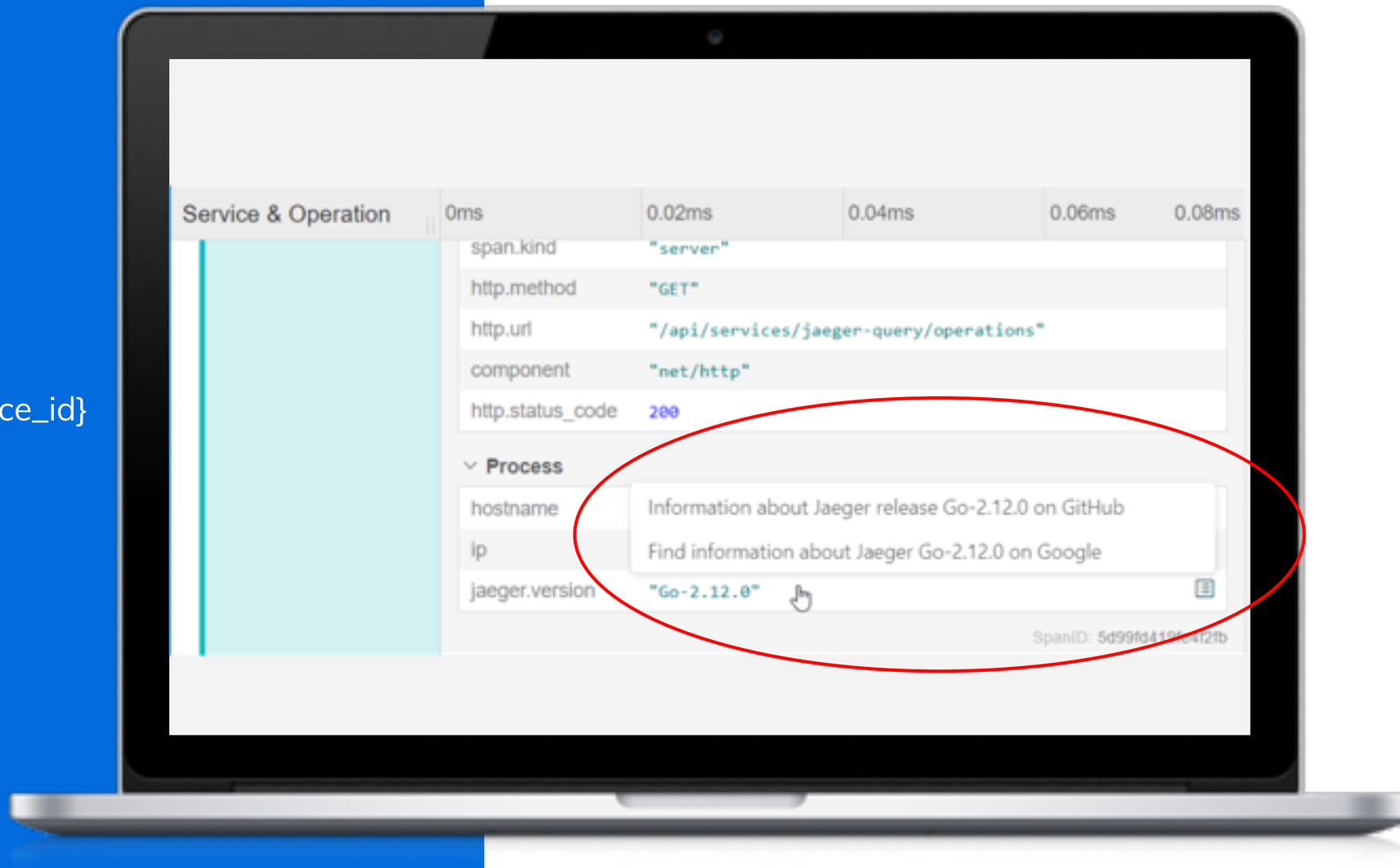


Extend

Links on Tag

➔ AWS Console

https://console.aws.amazon.com/ec2/v2/home?#Instances:search=#{aws.instance_id}



Extend

Links on Tag

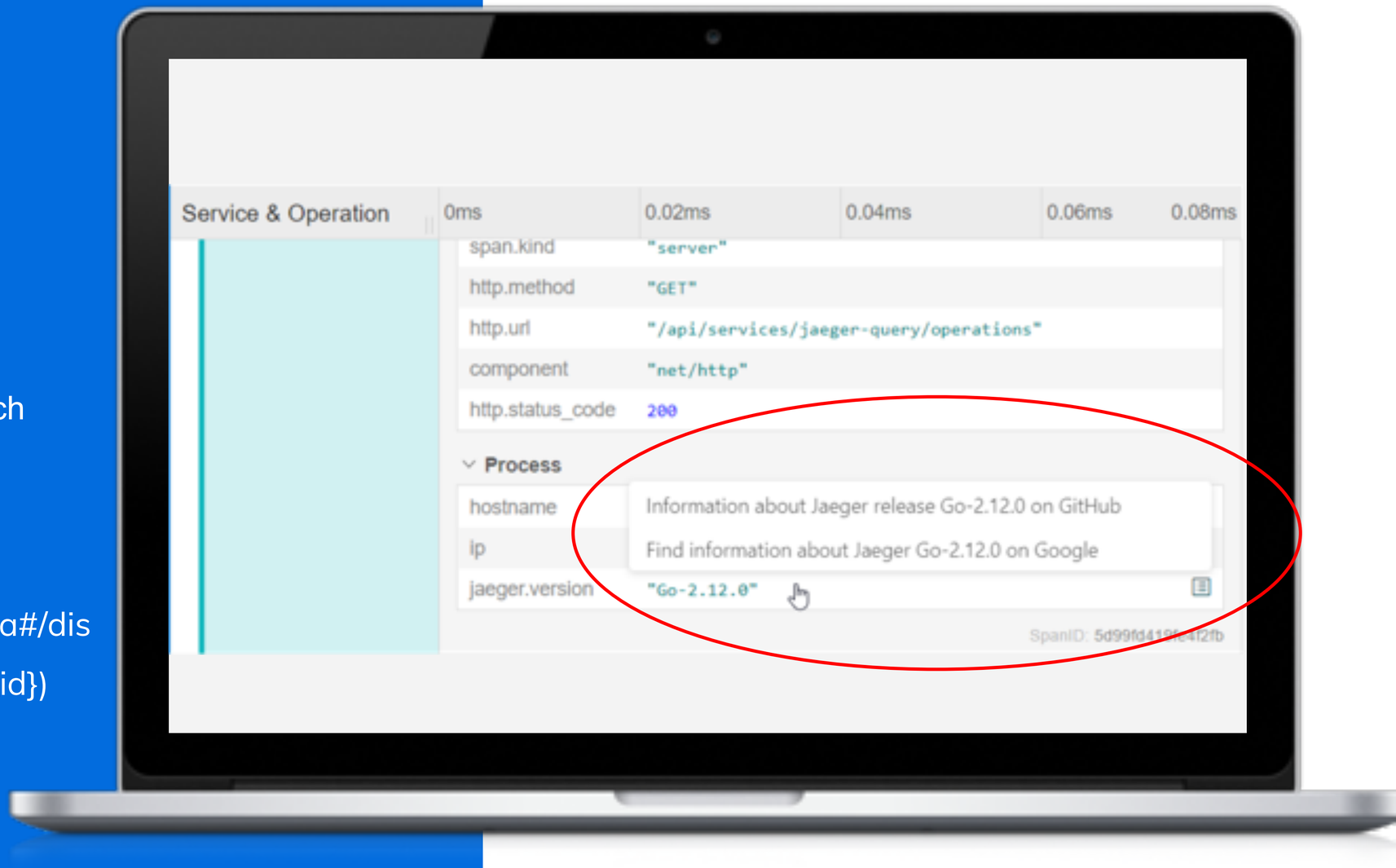
➔ Find the associated logs

- ◆ Splunk:

https://splunk/app/search/search?q=search index=* session_id=#{session_id}

- ◆ Kibana:

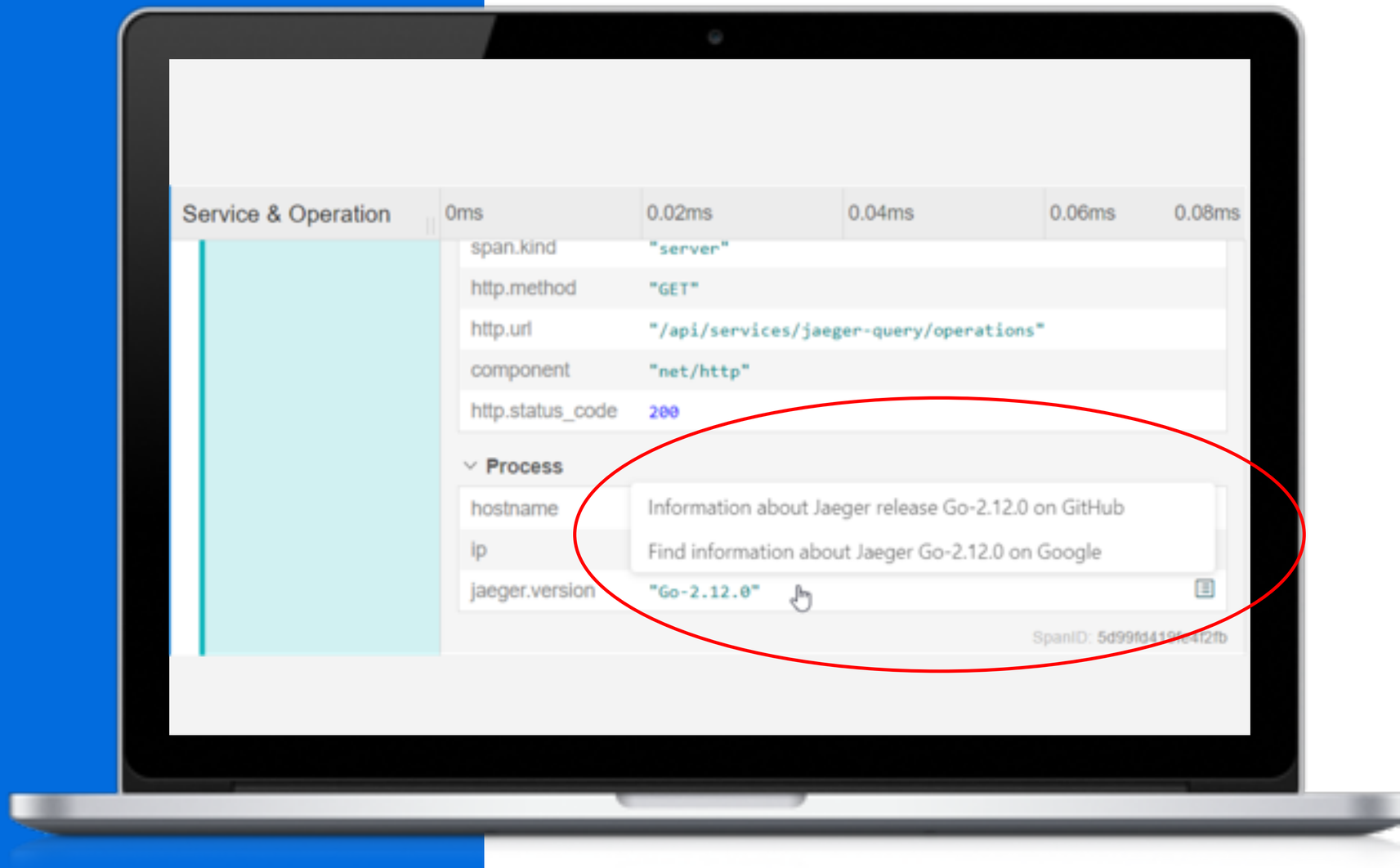
https://kibana/_plugin/kibana/app/kibana#/discover?_a=(query:'session_id=#{session_id})

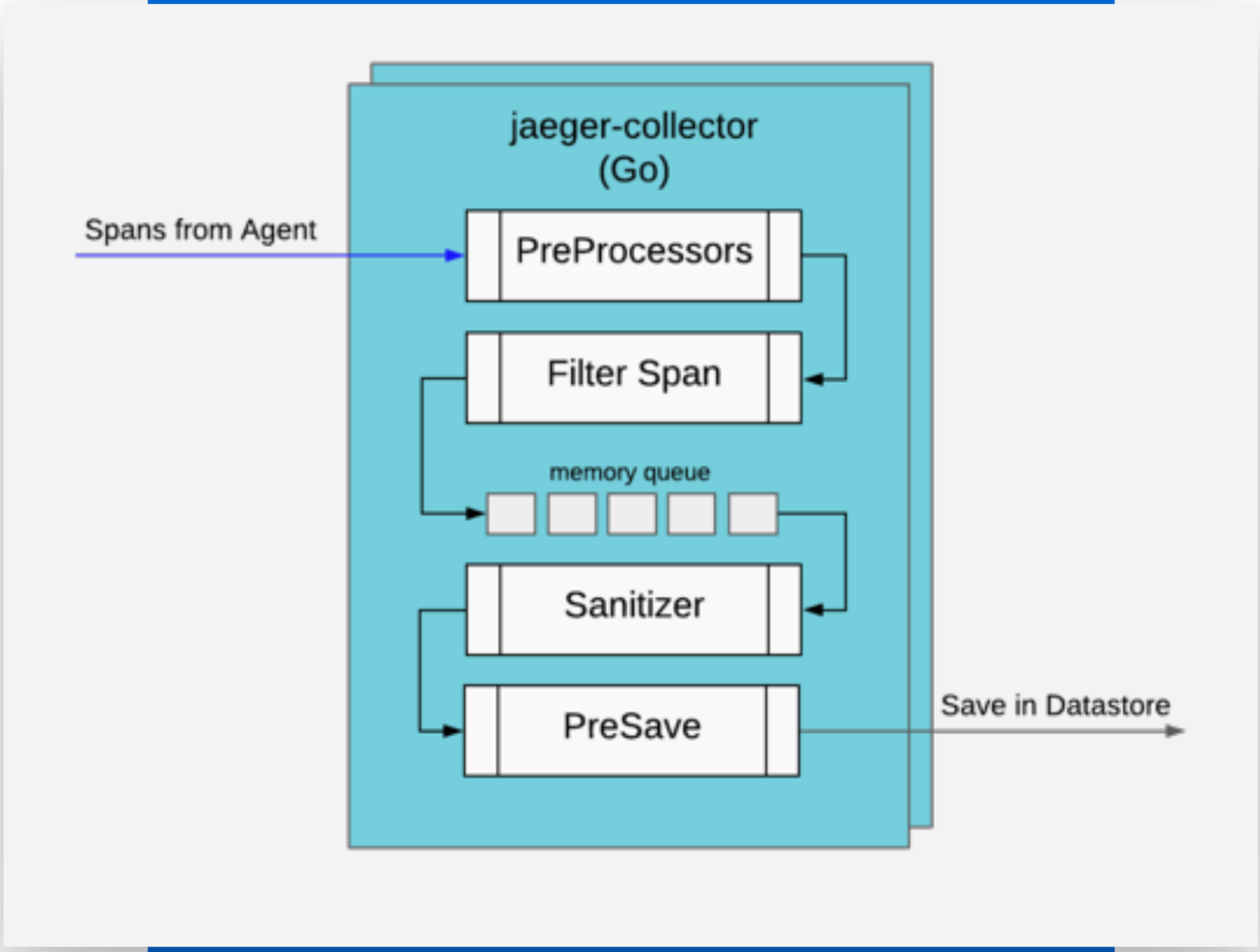


Extend

Links on Tag

- ➔ Monitoring Dashboards
- ➔ Git
- ➔ Runbook
- ➔ etc.



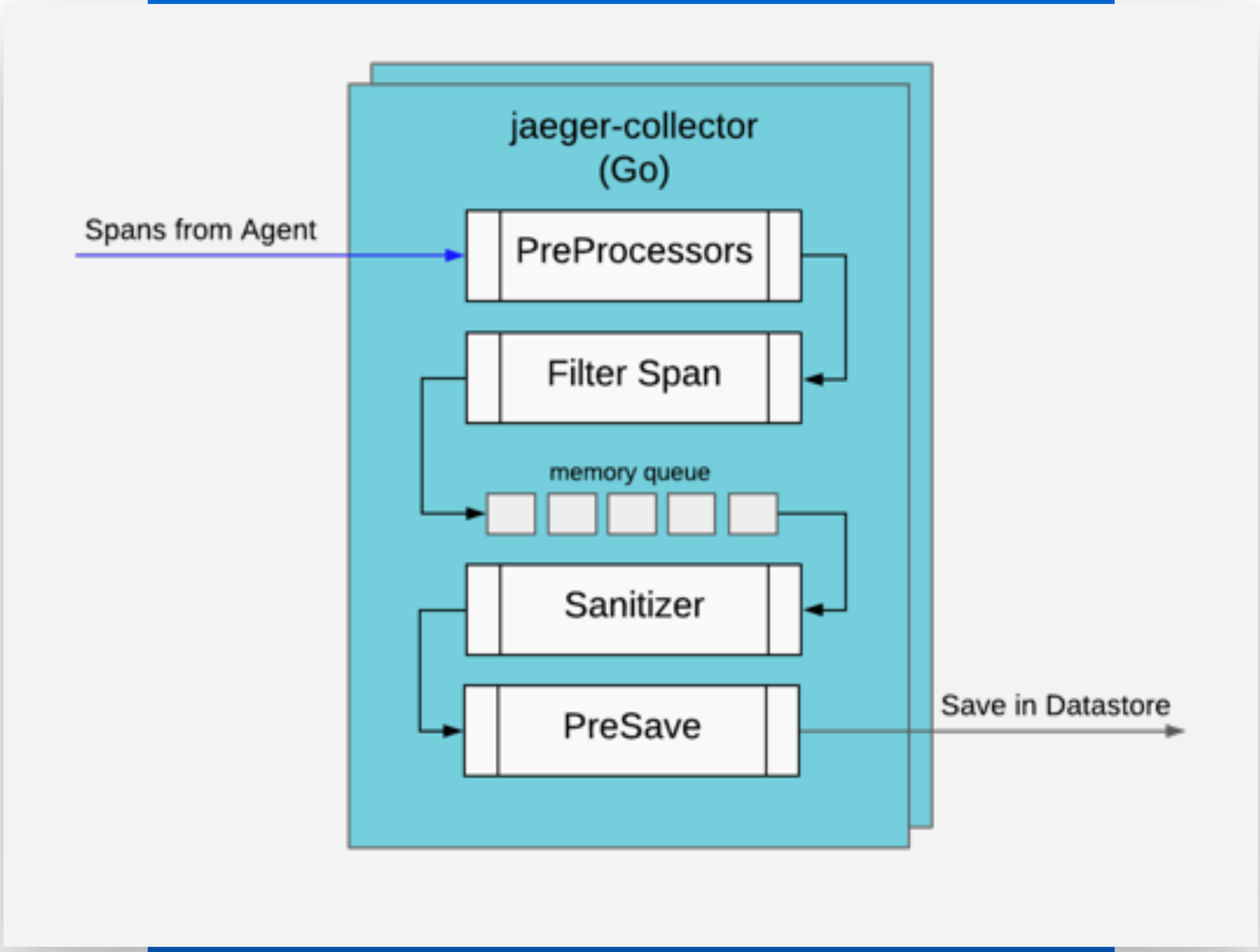


Extend

Processors

Currently not configurable

[jaegertracing/jaeger/issues/1530]



Extend

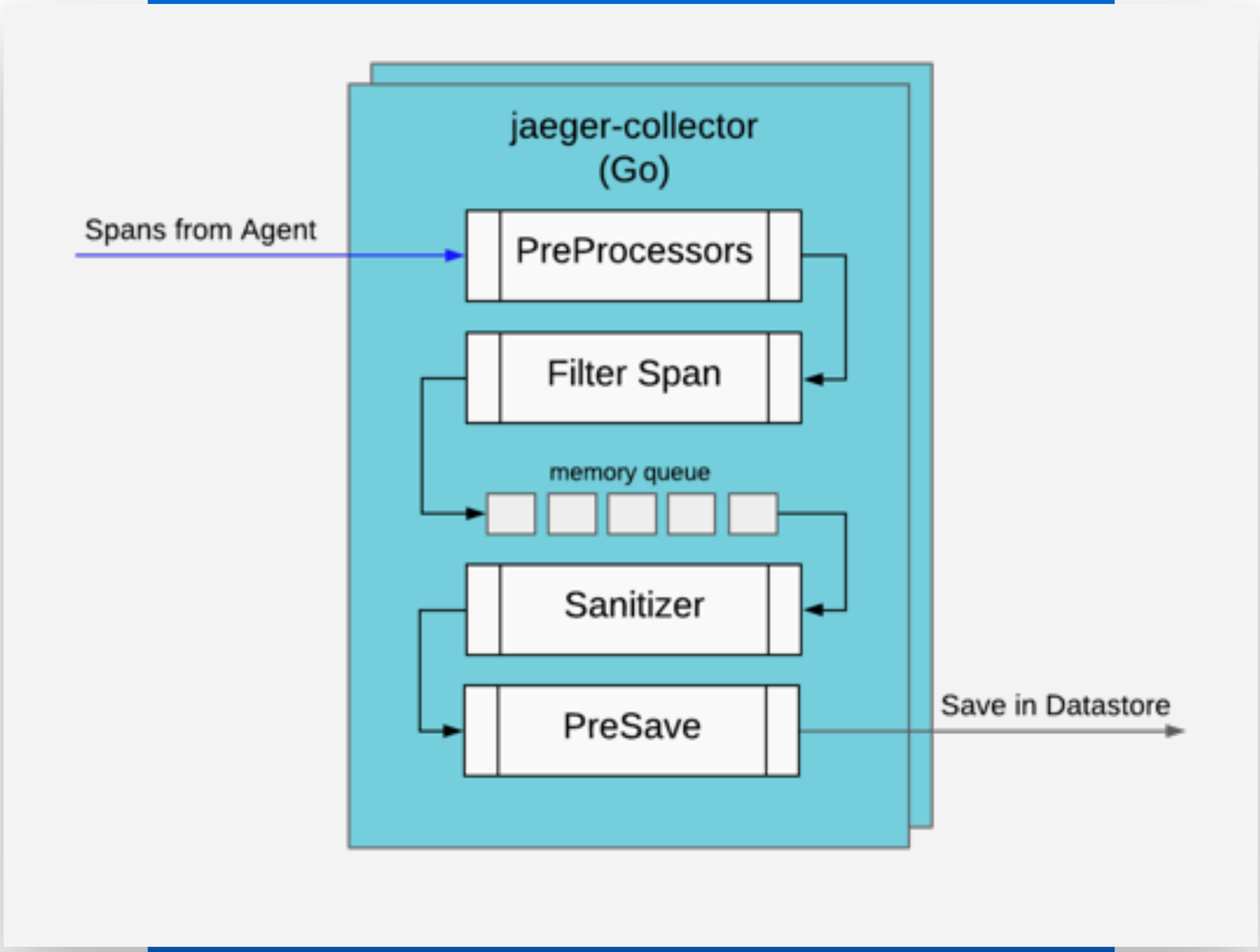
Spans PreProcessors

➔Extract Metrics per batch

➔Normalize Tags that are used in Filter Span

➔Best practice

- ◆ No Tags
- ◆ Minimum client version



Extend

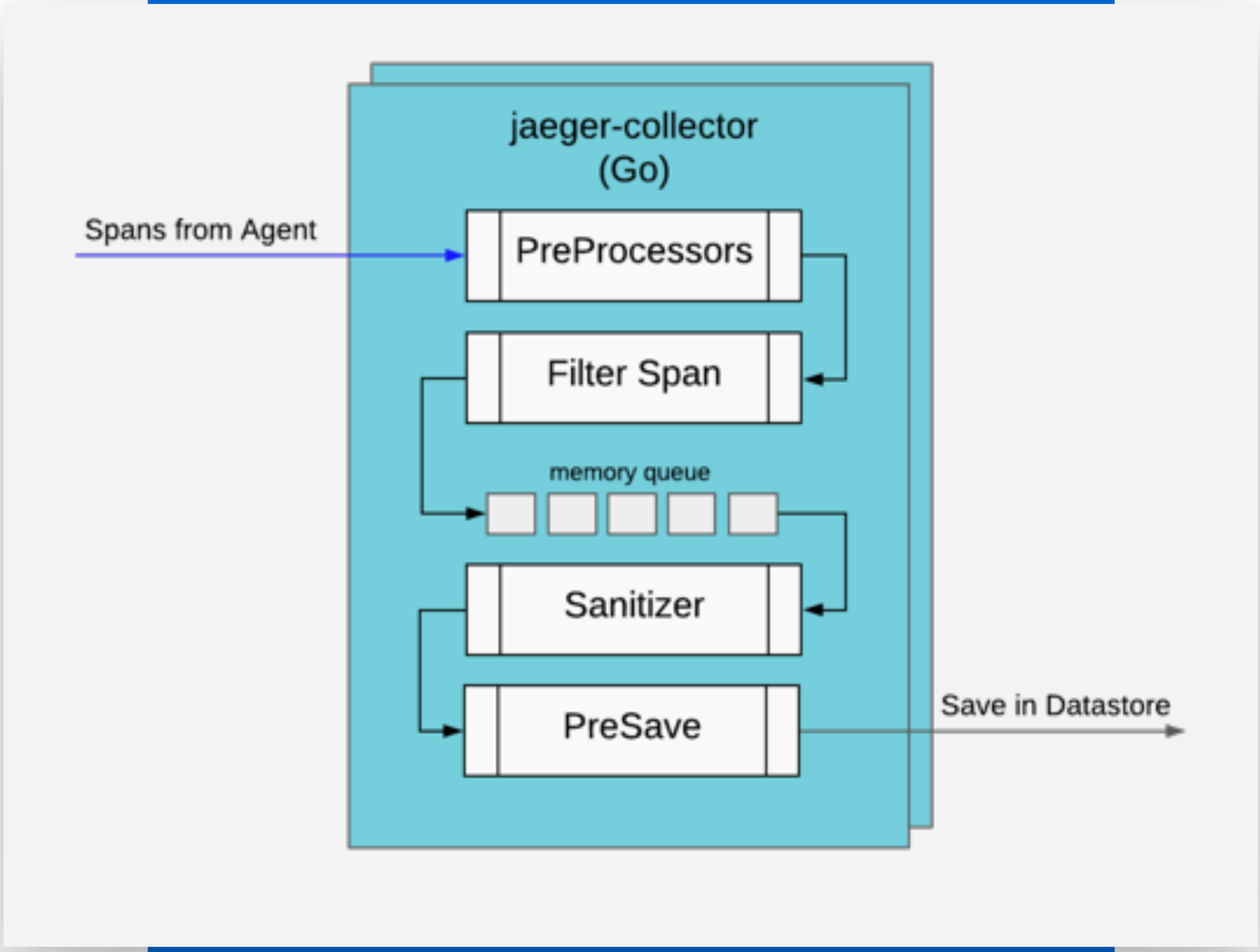
Span Filter

➔Blacklist / whitelist service

➔Enforce mandatory tags

➔Enforce Remote Sampler

- jaegertracing/jaeger/issues/1287



Extend

Span Sanitizer

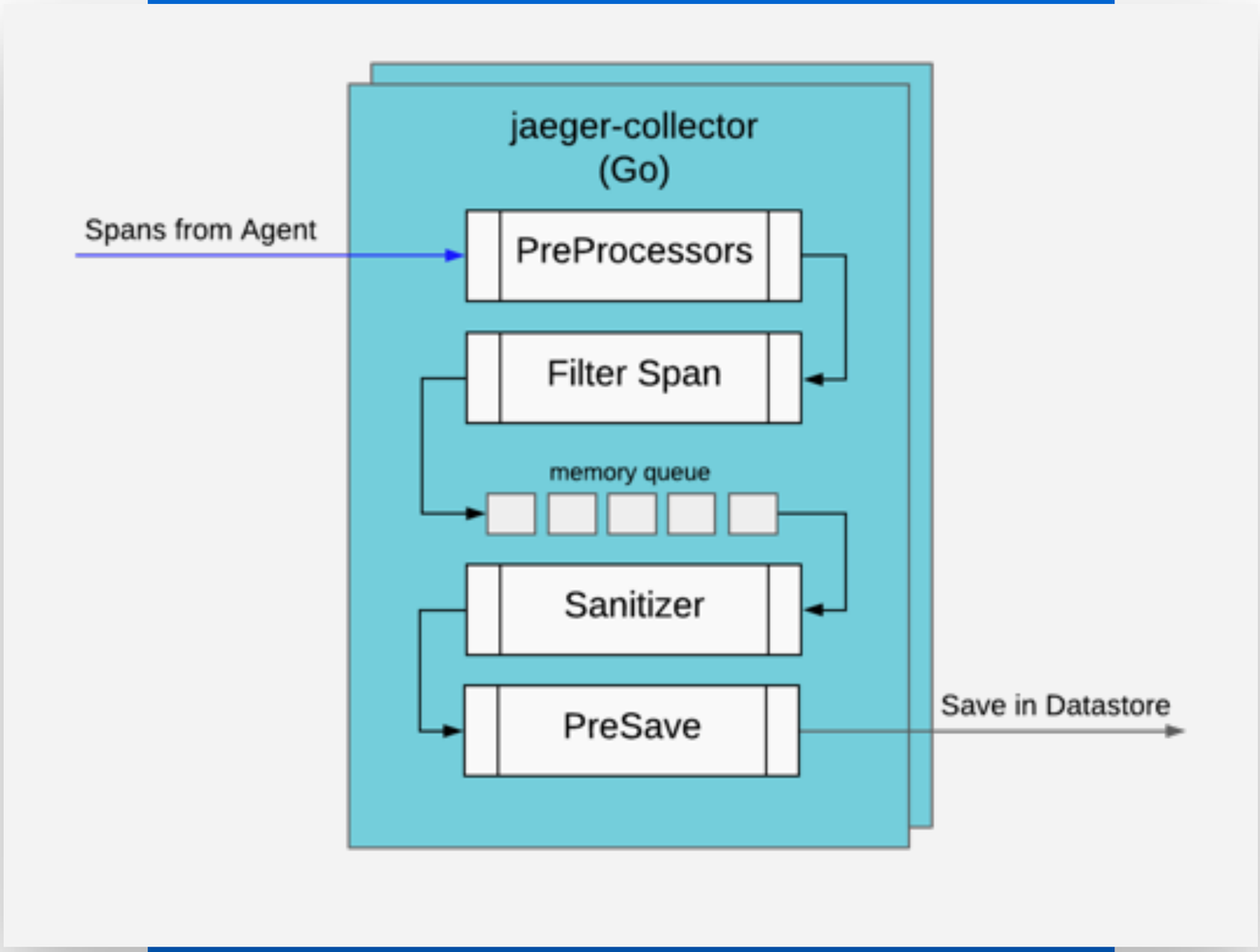
➔Normalize Tags format

- ◆ http_status, http.status, http.status_code

➔Data scrubbing

➔Add information from other system

- ◆ Inventory Asset Management



Extend

Pre Save

?!

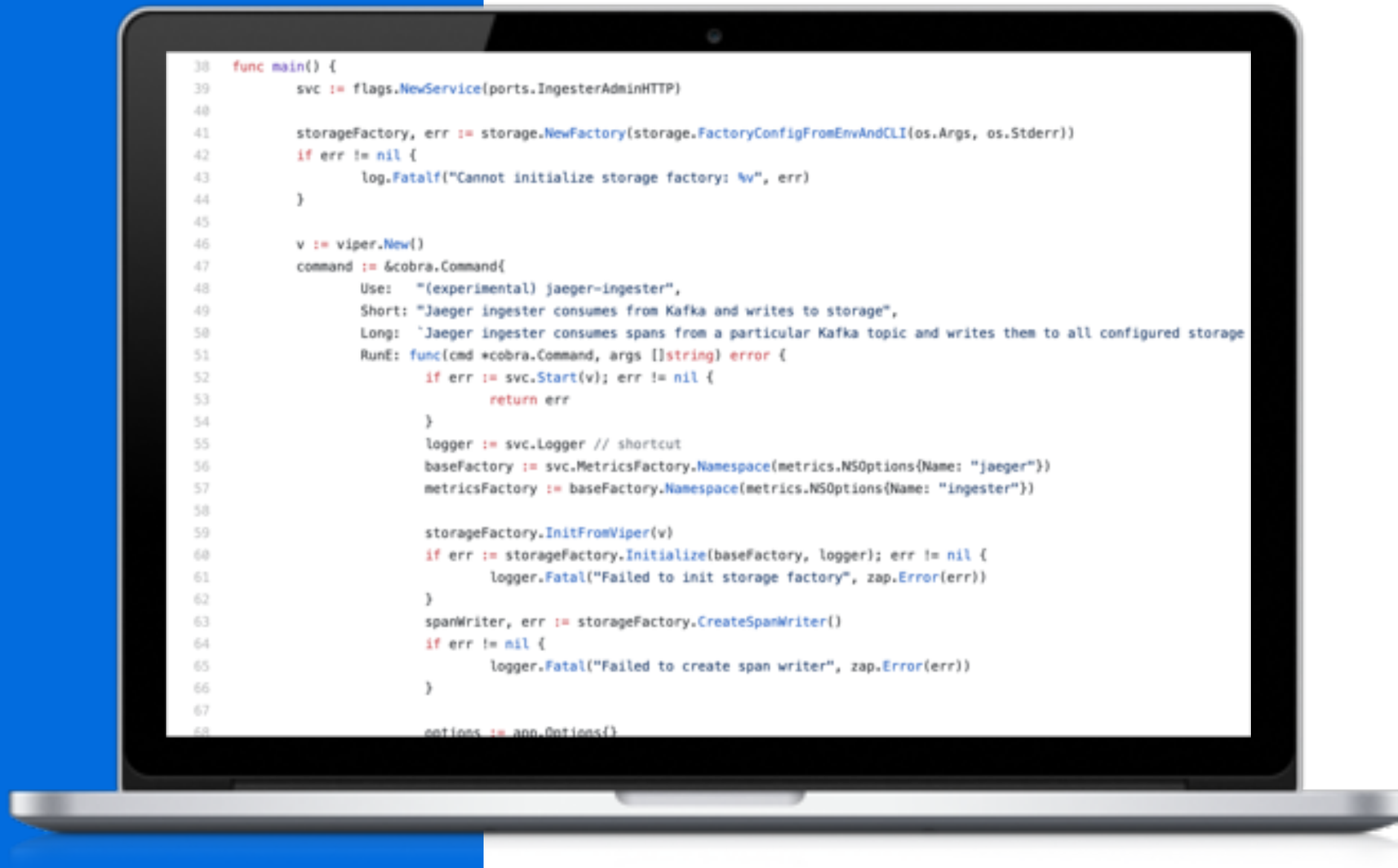
Any ideas? Comment here:

jaegertracing/jaeger/issues/1530

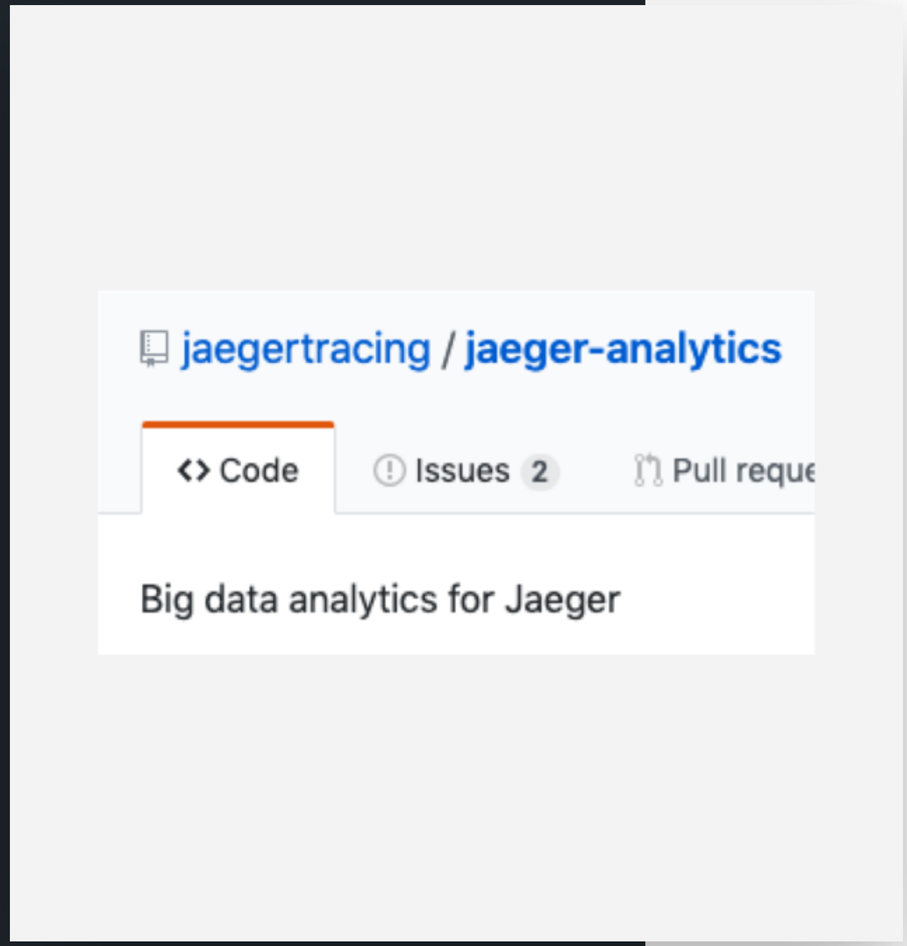
Extend

Don’t fork

Implement your own main()



```
func main() {
        svc := flags.NewService(ports.IngesterAdminHTTP)

        storageFactory, err := storage.NewFactory(storage.FactoryConfigFromEnvAndCLI(os.Args, os.Stderr))
        if err != nil {
                log.Fatalf("Cannot initialize storage factory: %v", err)
        }

        v := viper.New()
        command := &cobra.Command{
                Use:   "(experimental) jaeger-ingester",
                Short: "Jaeger ingester consumes from Kafka and writes to storage",
                Long:  `Jaeger ingester consumes spans from a particular Kafka topic and writes them to all configured storage
                RunE: func(cmd *cobra.Command, args []string) error {
                        if err := svc.Start(v); err != nil {
                                return err
                        }
                        logger := svc.Logger // shortcut
                        baseFactory := svc.MetricsFactory.Namespace(metrics.NSOptions{Name: "jaeger"})
                        metricsFactory := baseFactory.Namespace(metrics.NSOptions{Name: "ingester"})

                        storageFactory.InitFromViper(v)
                        if err := storageFactory.Initialize(baseFactory, logger); err != nil {
                                logger.Fatal("Failed to init storage factory", zap.Error(err))
                        }
                        spanWriter, err := storageFactory.CreateSpanWriter()
                        if err != nil {
                                logger.Fatal("Failed to create span writer", zap.Error(err))
                        }

                        options := app.Options{}
```



Extend

Analytics

- ➔Dependency Map
  - ◆ Show services
    - AWS API
    - Google Maps API
  - ◆ External component
    - Redis
    - SQL/NoSQL
- ➔Trace Quality
- ➔Latency Histogram

References

- https://medium.com/jaegertracing/deployment-strategies-for-the-jaeger-agent-1d6f91796d09
- https://medium.com/jaegertracing/tuning-jaegers-performance-7a60864cf3b1
- https://medium.com/jaegertracing/running-jaeger-agent-on-bare-metal-d1fc47d31fab
- https://www.jaegertracing.io/docs/latest/architecture/
- https://github.com/jaegertracing/jaeger
- https://github.com/jaegertracing/jaeger-analytics

Thank you!

Louis-Etienne Dorval

github.com/ledor473

twitter.com/ledor473