ENTERPRISE DEVOPS TECHCON

Debugging Microservices in Kubernetes

Sander Rodenhuis

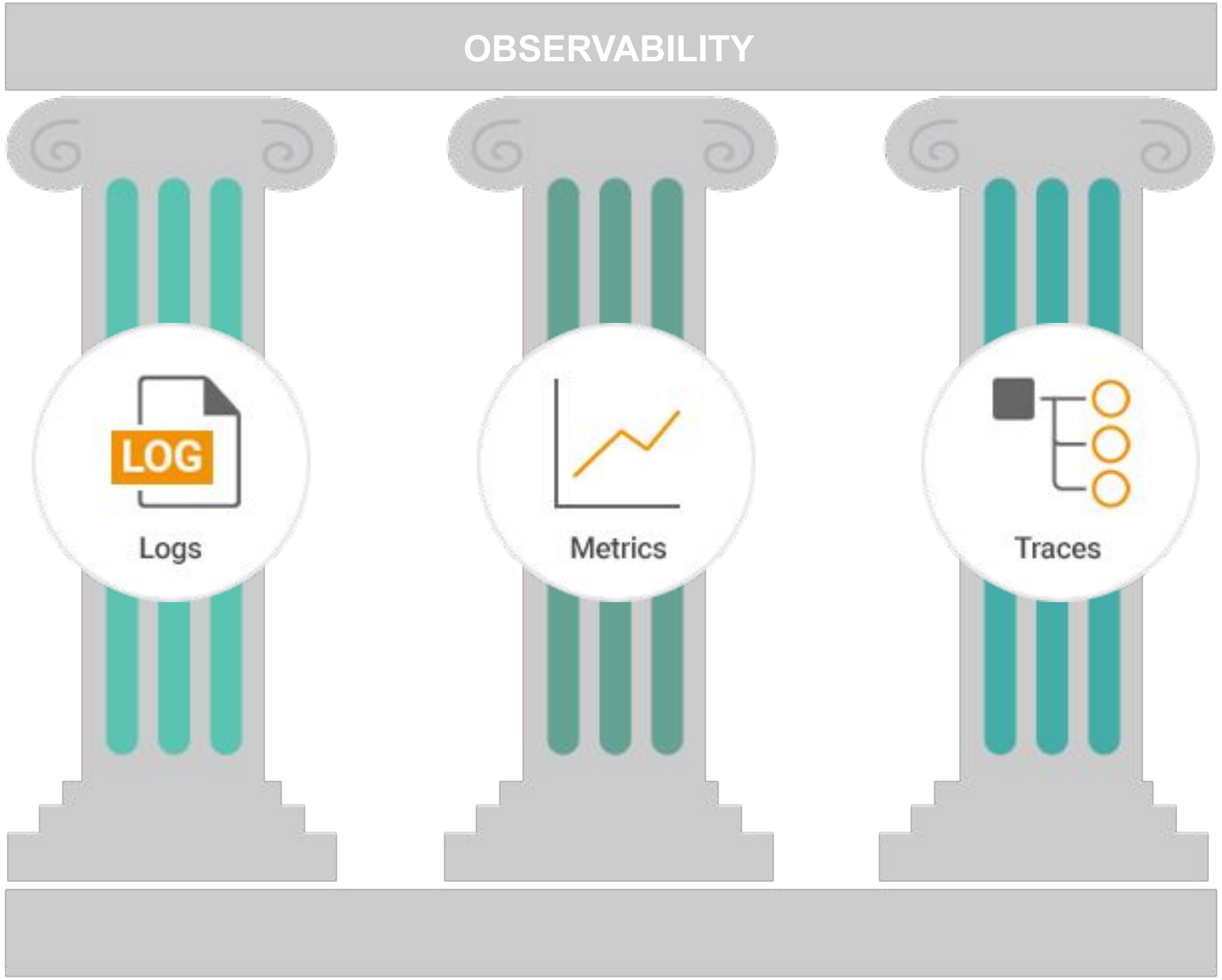
Debugging Microservices in Kubernetes

Using Istio, OpenTelemetry and Grafana Tempo

Agenda

- The 3 pillars of observability
- The what and why of Distributed tracing
- Using Istio for tracing (the good, the .. and the …)
- Introducing OpenTelemetry
- Things to consider when getting started
- Observability with distributed tracing in Otomi
- Demo: Tracing with Nginx, Istio, OpenTelemetry, Prometheus, Grafana Loki and Tempo

The 3 pillars of Observability



The 3 pillars of Observability

Logs

Records of events, warnings and errors within a microservice.

Provide insights into events and errors during the lifecycle of microservices.

Metrics

Quantifiable measurements that reflect the health and performance of a microservice.

Provide real-time insight into the state of microservices.

Traces

Data that tracks a request as it flows through various microservices.

Research the root cause of a problem with a microservices architecture consisting out of multiple microservices.

Logs, Metrics and Traces each provide valuable but limited visibility. Only by combining them you'll get the complete picture. And everything is about CONTEXT!

Distributed tracing

- A method of tracking application requests as they flow between microservices
- Understand the behaviour of an application consisting out of multiple microservices
- Troubleshooting performance bottlenecks
- Fix errors, and other issues that could impact the user experience
- But you can’t trace everything!

The basics

Trace

A complete **end-to-end path of a request** or transaction as it flows through a distributed system.

Represents the journey of a specific operation as it traverses various components and services in a distributed architecture.

Span

A single operation or **unit of work** within a distributed system.

Captures the timing and metadata associated with a specific operation and provides a way to track and understand the behavior of individual components and services.

Context Propagation

Passing contextual information between different components or services within a distributed system.

Crucial for connecting and correlating spans to construct a complete trace of a request or transaction as it flows through various services.

Why tracing with Istio

- Responsible for managing traffic, it can also report logs, metrics, and traces
- Leverages Envoy's distributed tracing feature to provide tracing integration out of the box
- A service mesh can introduce its own delays and issues
- Visibility into this layer of infrastructure is useful in troubleshooting
- Easy to get started

Tracing using Istio (defaults)

The good

- Easy to setup
- No coding required

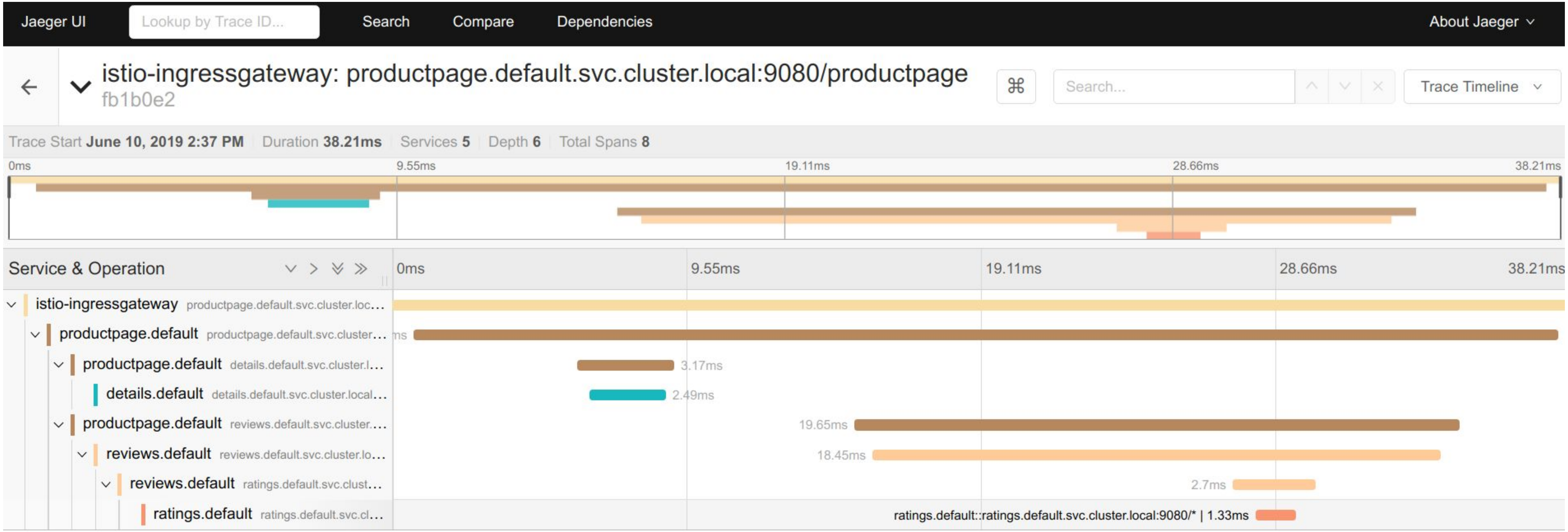
The bad

- Only collects partial data with partial context
- One sampling rate for all traffic

The ugly

- Uses Jaeger and Zipkin format
- No active development for Jaeger SDK (needed for instrumentation)

What we see in the demo



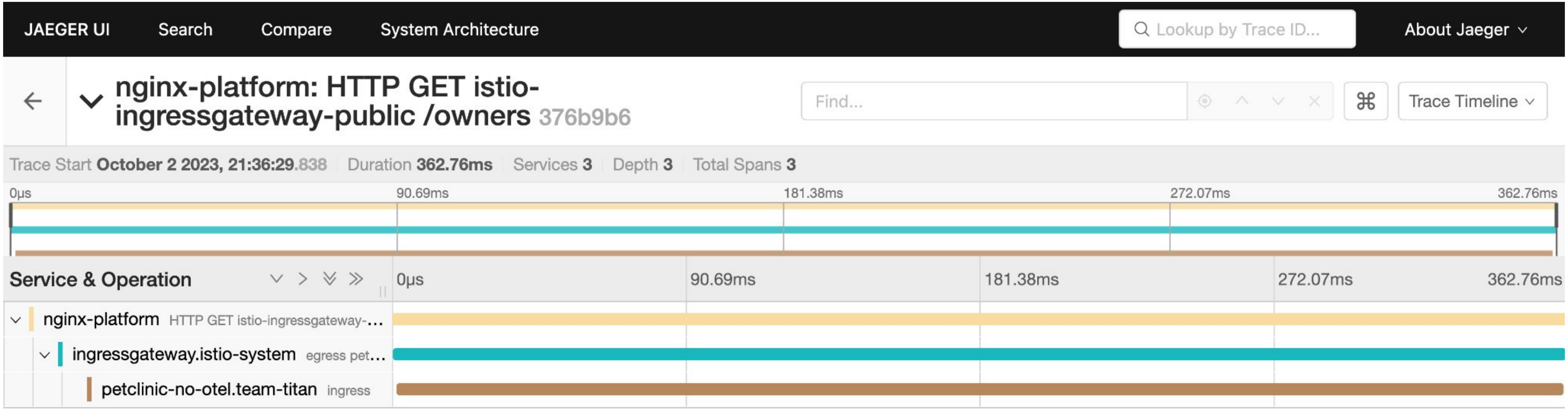
istio-ingressgateway: productpage.default.svc.cluster.local:9080/productpage
fb1b0e2

Trace Start **June 10, 2019 2:37 PM** | Duration **38.21ms** | Services **5** | Depth **6** | Total Spans **8**

Service & Operation

istio-ingressgateway productpage.default.svc.cluster.loc...
productpage.default productpage.default.svc.cluster...
productpage.default details.default.svc.cluster.l... 3.17ms
details.default details.default.svc.cluster.local... 2.49ms
productpage.default reviews.default.svc.cluster.... 19.65ms
reviews.default reviews.default.svc.cluster.lo... 18.45ms
reviews.default ratings.default.svc.clust... 2.7ms
ratings.default ratings.default.svc.cl... ratings.default::ratings.default.svc.cluster.local:9080/* | 1.33ms

What we see of our own app

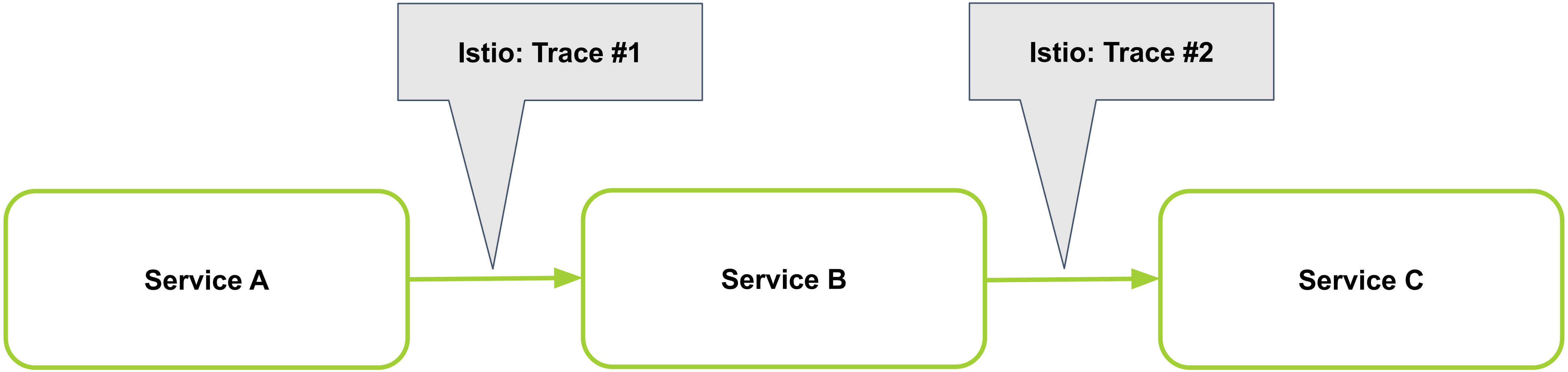


nginx-platform: HTTP GET istio-ingressgateway-public /owners 376b9b6

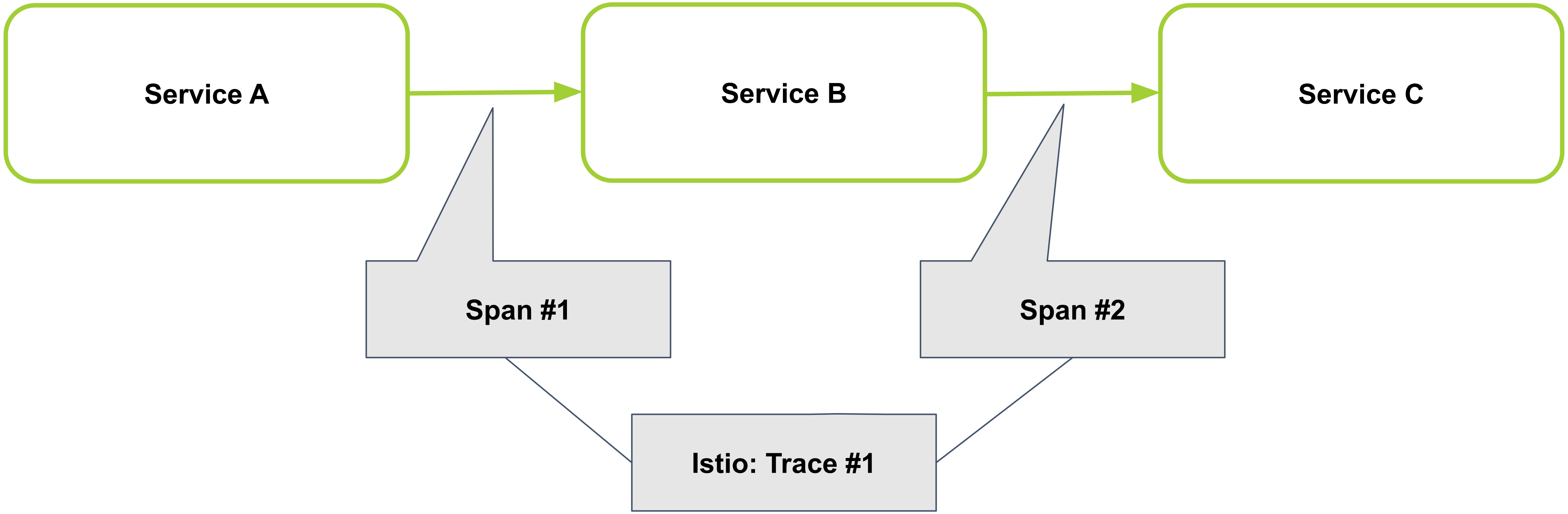
Trace Start **October 2 2023, 21:36:29**.838 | Duration **362.76ms** | Services **3** | Depth **3** | Total Spans **3**

Service & Operation	0μs	90.69ms	181.38ms	272.07ms	362.76ms
nginx-platform HTTP GET istio-ingressgateway-…					
ingressgateway.istio-system egress pet…					
petclinic-no-otel.team-titan ingress					

Without context propagation



With context propagation



The good news: Istio now supports OTEL

```
meshConfig:
   extensionProviders:
  - name: otel-tracing
    opentelemetry:
        port: 4317
        service: otel-collector.svc.cluster.local
```

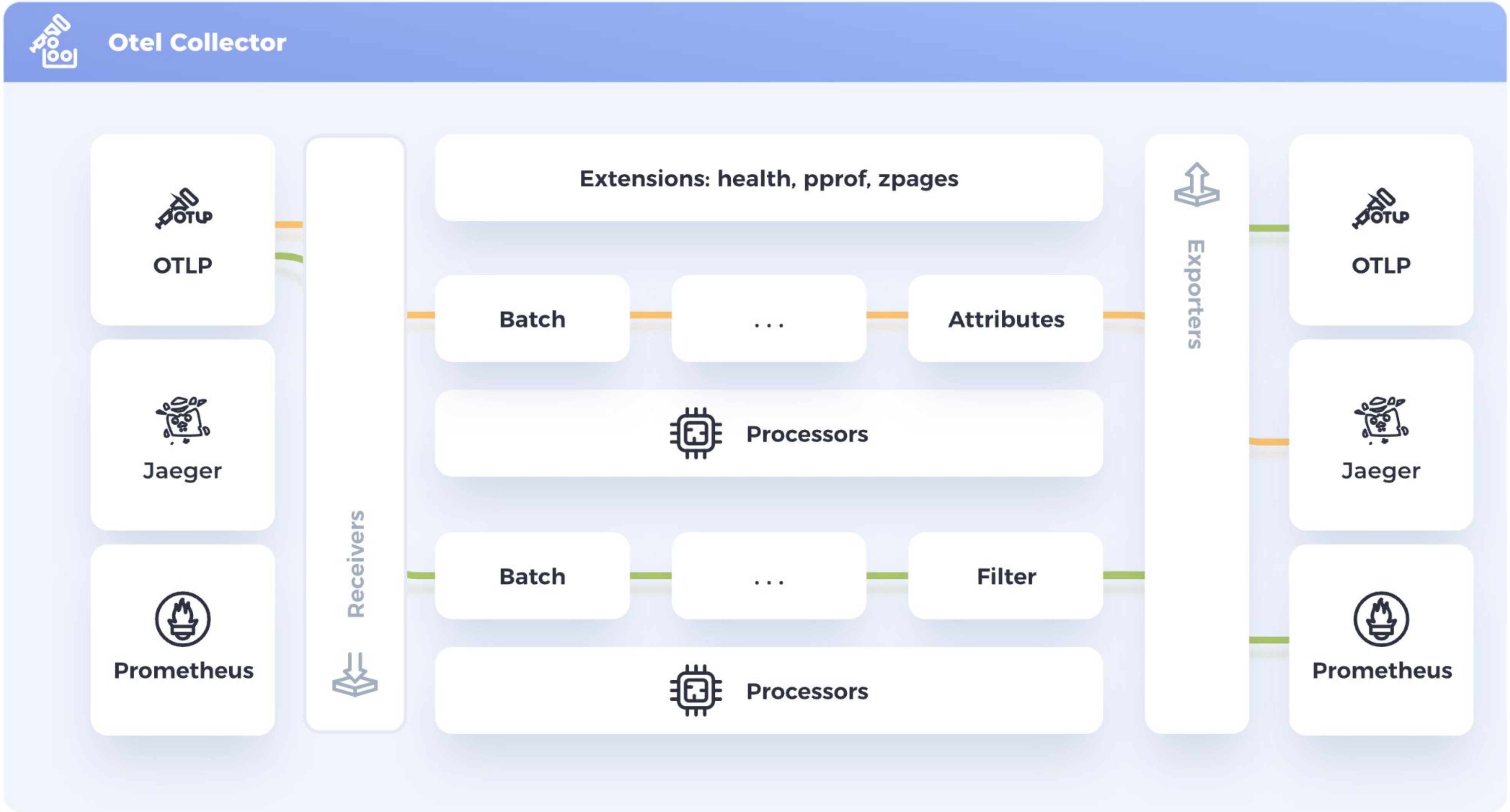
OpenTELemetry

- Framework and toolkit to create and manage telemetry data
- Can be used with a broad variety of observability backends

Components:

- Collector to receive, process and export telemetry data
- Code Instrumentation support for many languages

Otel Collector

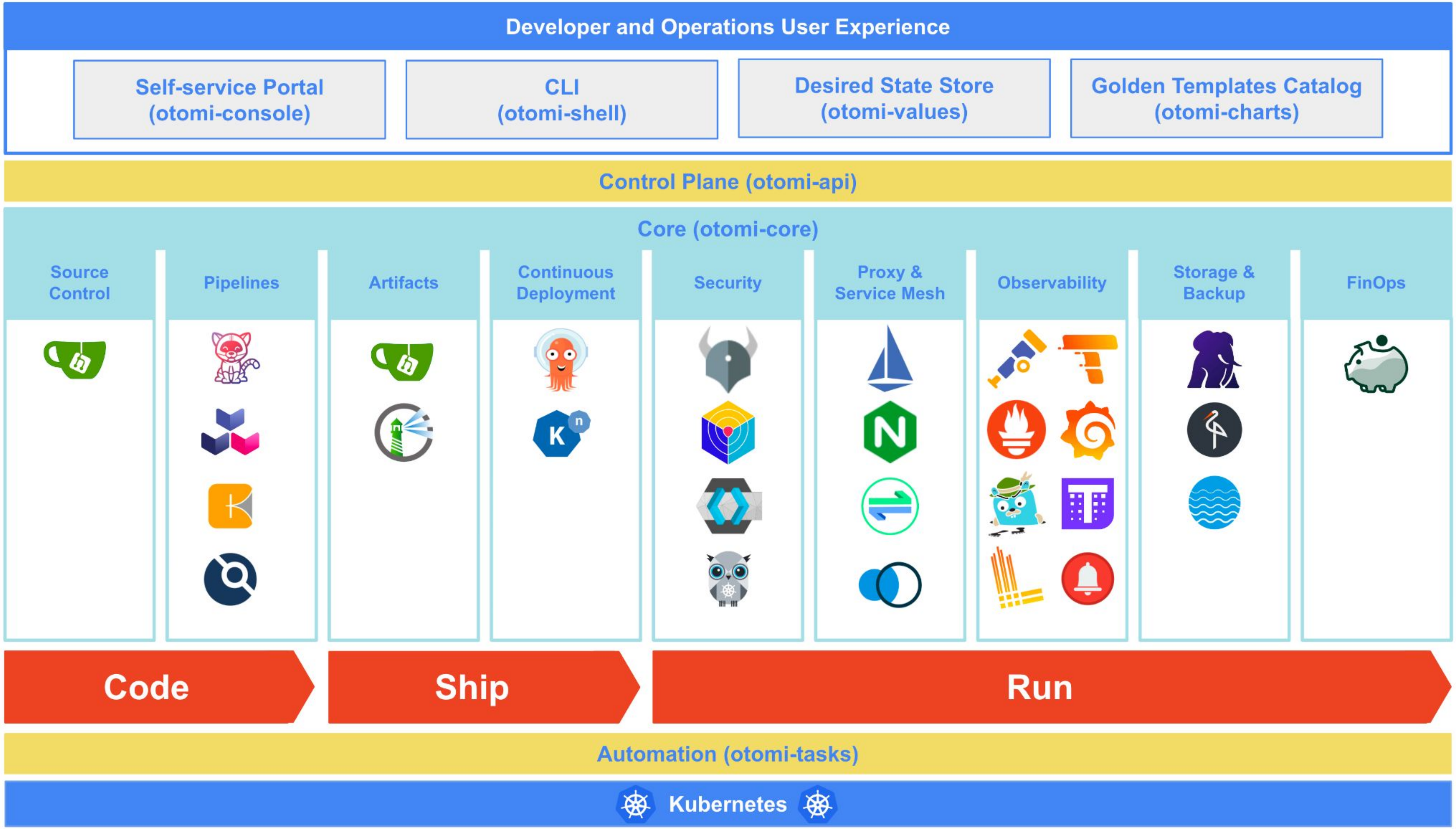


Otel auto instrumentation

```
apiVersion: opentelemetry.io/v1alpha1
kind: Instrumentation
metadata:
  name: my-app-instrumentation
spec:
  propagators:
    - tracecontext
    - baggage
  sampler:
    type: always_on
  java:
    env:
      - name: OTEL_EXPORTER_OTLP_ENDPOINT
        value: http://otel-collector.otel.svc.cluster.local:4317
```

Considerations when getting started

- Istio is difficult, so is OpenTelemetry
- What backend to use?
- Use auto instrumentation or the SDK?
- Use Otel for everything (metrics, logs, traces)?
- Scaling of the Collector
- Sampling rates
- How to correlate telemetry data?
- How to offer a seamless experience to developers?



Developer and Operations User Experience

Self-service Portal (otomi-console)

CLI (otomi-shell)

Desired State Store (otomi-values)

Golden Templates Catalog (otomi-charts)

Control Plane (otomi-api)

Core (otomi-core)

Source Control

Pipelines

Artifacts

Continuous Deployment

Security

Proxy & Service Mesh

Observability

Storage & Backup

FinOps

Code

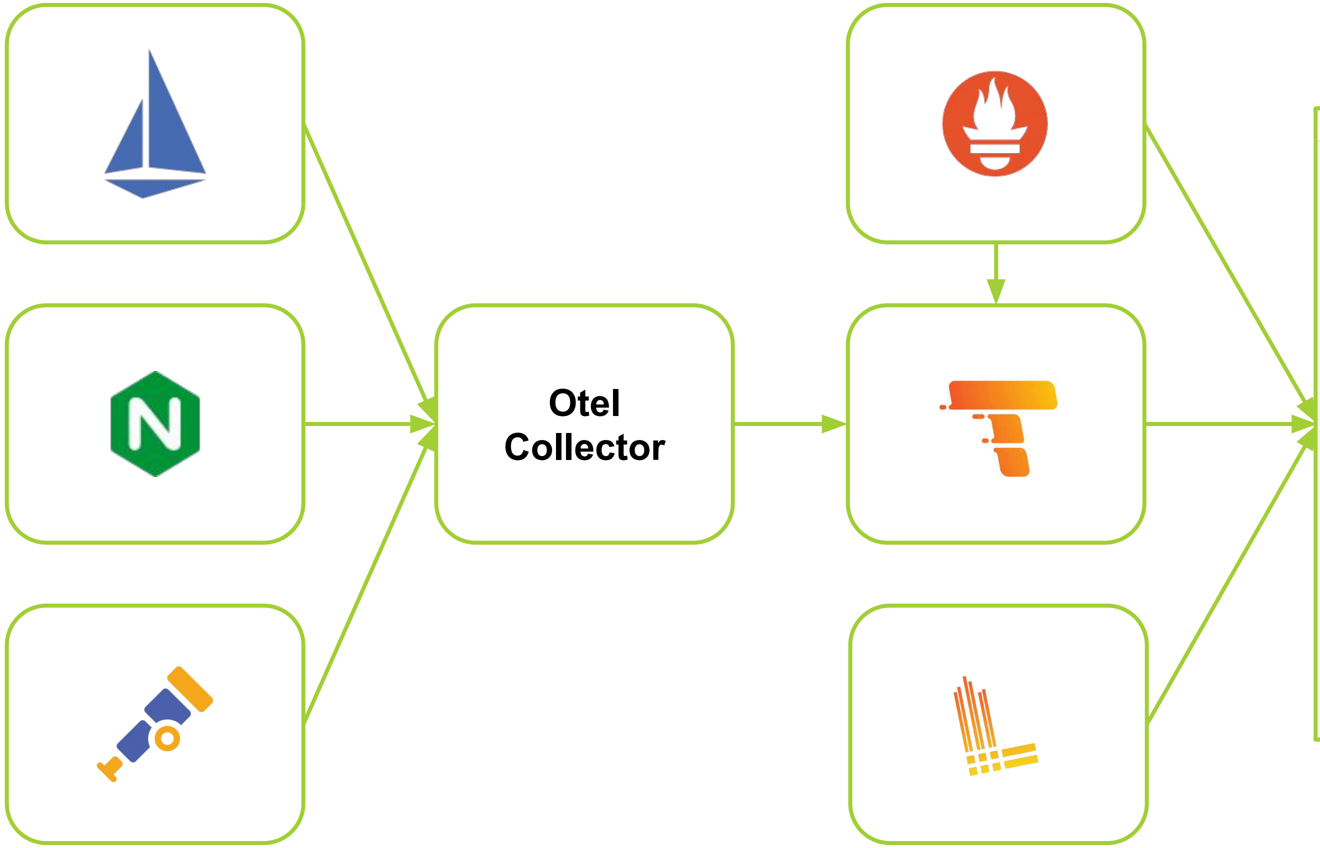
Ship

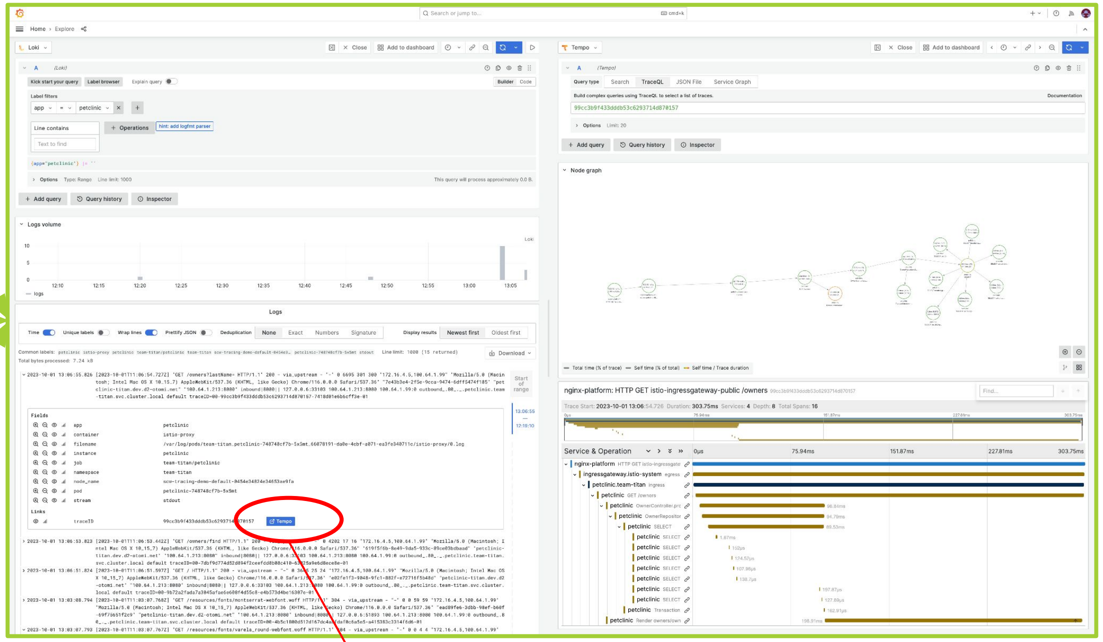
Run

Automation (otomi-tasks)

Kubernetes

Observability in Otomi





Loki (A):

Label filters: app = petclinic

```
{app="petclinic"} |= ``
```

Options Type: Range Line limit: 1000 This query will process approximately 0.0 B.

Logs volume: 10, 5, 0; 12:10, 12:15, 12:20, 12:25, 12:30, 12:35, 12:40, 12:45, 12:50, 12:55, 13:00, 13:05

Logs: Time ● Unique labels ○ Wrap lines ● Prettify JSON ○ Deduplication: None Exact Numbers Signature Display results: Newest first Oldest first Download

Common labels: petclinic istio-proxy petclinic team-titan/petclinic team-titan scw-tracing-demo-default-0454e3… petclinic-748748cf7b-5x5mt stdout Line limit: 1000 (15 returned)
Total bytes processed: 7.24 kB

```
2023-10-01 13:06:55.826 [2023-10-01T11:06:54.727Z] "GET /owners?lastName= HTTP/1.1" 200 - via_upstream - "-" 0 6695 301 300 "172.16.4.5,100.64.1.99" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/116.0.0.0 Safari/537.36" "7e43b3e4-2f5e-9cca-9474-6dff5474f185" "petclinic-titan.dev.d2-otomi.net" "100.64.1.213:8080" inbound|8080|| 127.0.0.6:33103 100.64.1.213:8080 100.64.1.99:0 outbound_.80_._.petclinic.team-titan.svc.cluster.local default traceID=00-99cc3b9f433dddb53c6293714d870157-7418d01e6b6cff3e-01
```

Start of range 13:06:55 — 12:19:10

Fields

app	petclinic
container	istio-proxy
filename	/var/log/pods/team-titan_petclinic-748748cf7b-5x5mt_66078191-da0e-4cbf-a071-ea3fe340711c/istio-proxy/0.log
instance	petclinic
job	team-titan/petclinic
namespace	team-titan
node_name	scw-tracing-demo-default-0454e34824e34653ae9fa
pod	petclinic-748748cf7b-5x5mt
stream	stdout

Links

traceID 99cc3b9f433dddb53c6293714d870157 Tempo

```
2023-10-01 13:06:53.823 [2023-10-01T11:06:53.442Z] "GET /owners/find HTTP/1.1" 200 - [illegible] "-" 0 4202 17 16 "172.16.4.5,100.64.1.99" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/116.0.0.0 Safari/537.36" "619f5f6b-8e49-9da5-933c-89ce03bdbaad" "petclinic-titan.dev.d2-otomi.net" "100.64.1.213:8080" inbound|8080|| 127.0.0.6:33103 100.64.1.213:8080 100.64.1.99:0 outbound_.80_._.petclinic.team-titan.svc.cluster.local default traceID=00-7dbf9d774d52d894f2ceefdd8b08c410-63825a9e6d8ece8e-01
2023-10-01 13:06:51.824 [2023-10-01T11:06:51.597Z] "GET / HTTP/1.1" 200 - via_upstream - "-" 0 3645 25 24 "172.16.4.5,100.64.1.99" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/116.0.0.0 Safari/537.36" "e02fe1f3-9048-9fc1-882f-e72716f5b48d" "petclinic-titan.dev.d2-otomi.net" "100.64.1.213:8080" inbound|8080|| 127.0.0.6:33103 100.64.1.213:8080 100.64.1.99:0 outbound_.80_._.petclinic.team-titan.svc.cluster.local default traceID=00-9b72a2fada7a3045afae6e608f4d55c8-e4b573d4be16307e-01
2023-10-01 13:03:08.794 [2023-10-01T11:03:07.768Z] "GET /resources/fonts/montserrat-webfont.woff HTTP/1.1" 304 - via_upstream - "-" 0 0 59 59 "172.16.4.5,100.64.1.99" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/116.0.0.0 Safari/537.36" "ead89fe6-3dbb-98ef-b60f-69f75651f2c9" "petclinic-titan.dev.d2-otomi.net" "100.64.1.213:8080" inbound|8080|| 127.0.0.6:51893 100.64.1.213:8080 100.64.1.99:0 outbound_.80_._.petclinic.team-titan.svc.cluster.local default traceID=00-4b5c1800d512d167dc4a[illegible]daf0c6a5e5-a415383c3314f6d6-01
2023-10-01 13:03:07.793 [2023-10-01T11:03:07.767Z] "GET /resources/fonts/varela_round-webfont.woff HTTP/1.1" 304 - via_upstream - "-" 0 0 4 4 "172.16.4.5,100.64.1.99"
```

Tempo (A):

Query type: Search TraceQL JSON File Service Graph

Build complex queries using TraceQL to select a list of traces. Documentation

```
99cc3b9f433dddb53c6293714d870157
```

Options Limit: 20

nginx-platform: HTTP GET istio-ingressgateway-public /owners 99cc3b9f433dddb53c6293714d870157

Trace Start: 2023-10-01 13:06:54.726 Duration: 303.75ms Services: 4 Depth: 8 Total Spans: 16

0µs 75.94ms 151.87ms 227.81ms 303.75ms

Service & Operation	
nginx-platform HTTP GET istio-ingressgate	
ingressgateway.istio-system egress	
petclinic.team-titan ingress	
petclinic GET /owners	
petclinic OwnerController.prc	96.84ms
petclinic OwnerRepositor	94.79ms
petclinic SELECT	89.53ms
petclinic SELECT	1.67ms
petclinic SELECT	152µs
petclinic SELECT	124.52µs
petclinic SELECT	107.96µs
petclinic SELECT	138.7µs
petclinic SELECT	197.87µs
petclinic SELECT	127.89µs
petclinic Transaction	162.91µs
petclinic Render owners/own	198.91ms

Using Derived Fields

Otomi on GitHub

https://github.com/redkubes/otomi-core

And if you like the otomi project, give it a ★

ENTERPRISE
DEVOPS
TECHCON