
OpenTelemetry in Operate First

Sally O'Malley
@sallyom



Agenda

- What's OpenTelemetry (OTLP)?
 - Why OpenTelemetry?
 - Tracing in Operate First
 - Conclusion
-

What is Telemetry?

- **Tracing, Metrics, Logging, Events, Contexts**
 - Outputs of a system used to debug, optimize, understand, and monitor internal processes
- Trace and Metrics APIs are stable, Logs are experimental



What is OpenTelemetry

- An observability framework for cloud-native software
- Tools, APIs, and SDKs, and a Protocol to instrument, generate, collect, and export telemetry data (metrics, logs, and traces)
- **Vendor neutral** and **interoperable**
 - good for users, good for vendors



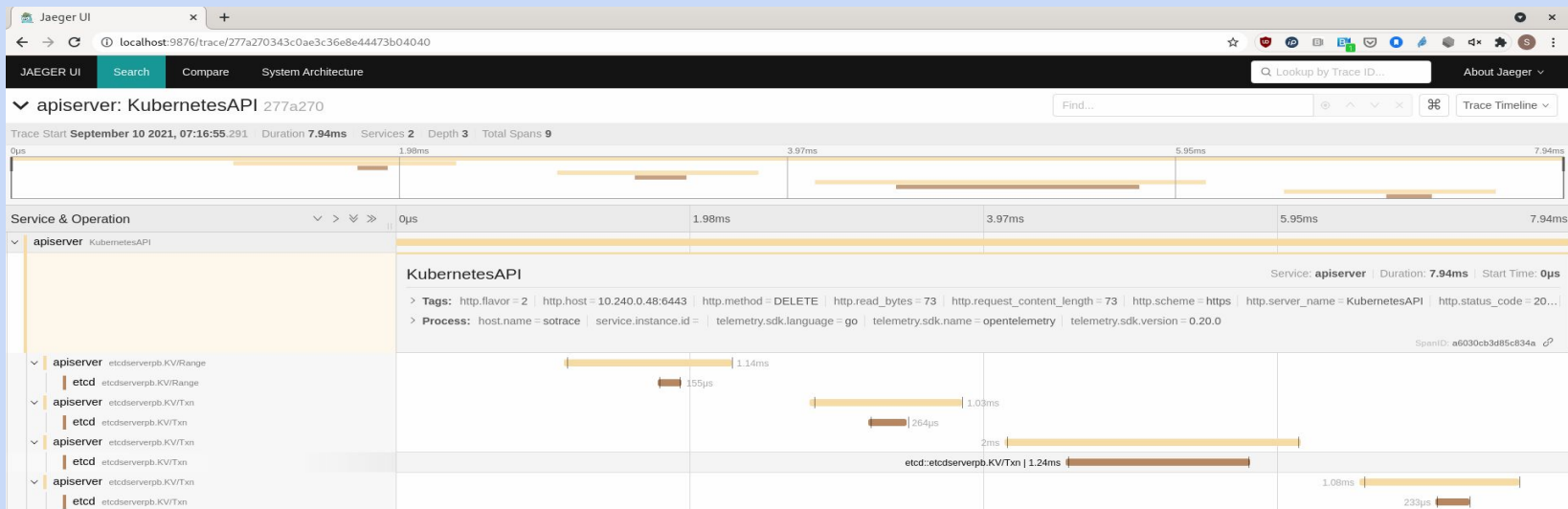
What's a Trace?

- Represents **work being done**
 - Measures end-to-end latency for a request
 - Click a button to add a product to the cart
 - how long did it take
 - did it complete or fail
 - what operation held it up?
 - [Example from KubeCon](#)
 - checkout -> payment svc latency
 - correlated host metrics, application metrics, traces, & logs to quickly determine root cause



What's a Span?

- **Span** - A single operation that takes place as part of a request
 - building blocks of a trace - a trace can include many spans
 - contains metadata like operation name, timestamps, attributes, events, and status plus custom key,value items



Let's Start Collecting!

- Instrumentation
 - Import and configure the OpenTelemetry API and SDK
 - Create Telemetry Data
 - Export Data



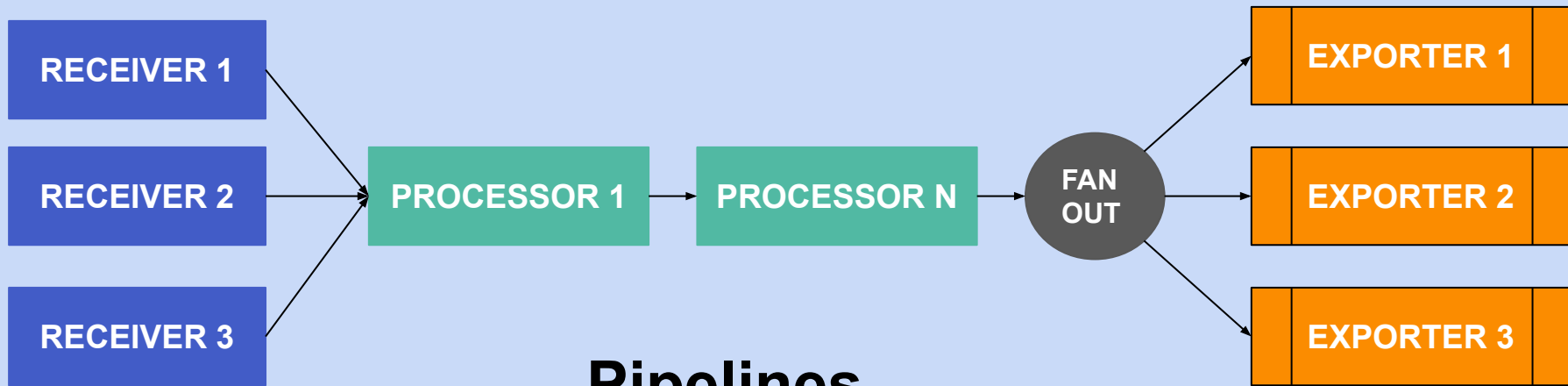
Let's Start Collecting!

- Instrumentation
 - OpenTelemetry Collector
 - Receive, process and export telemetry data
 - Supports several popular open-source protocols (**OTLP**)
 - Data receiving, transformation and sending is done using **Pipelines**.
-

Let's Start Collecting!

- Instrumentation libraries
 - OpenTelemetry Collector
 - Visualization (Jaeger)
 - Collector
 - Console
 - Query
-

Let's Start Collecting!



Pipelines

Let's Start Collecting!

RECEIVERS

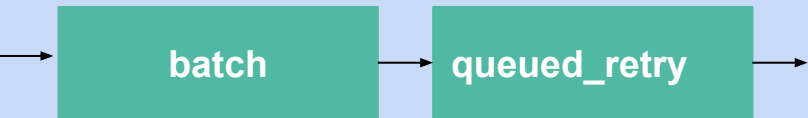
OTLP

```
receivers:  
  otlp:  
    protocols:  
      grpc:  
        max_recv_msg_size_mib: 999999999
```



Let's Start Collecting!

PROCESSORS



```
processors:  
  batch:  
    queued_retry:  
      size: 50  
      per-exporter: true  
      enabled: true
```



Let's Start Collecting!

EXPORTERS

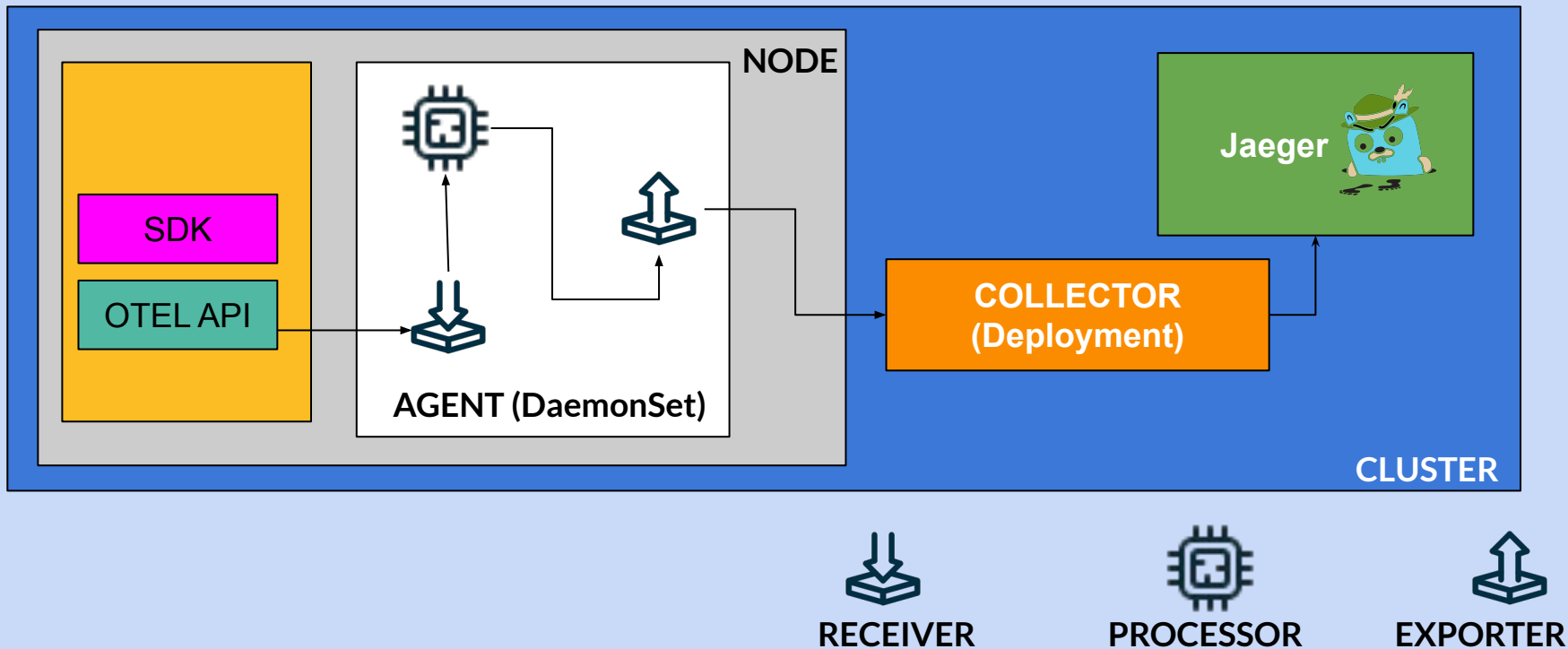
jaeger

logging

```
exporters:  
  jaeger:  
    protocols:  
      grpc:  
        endpoint: "0.0.0.0:14250"  
  
  logging:
```



Let's Start Collecting!



Tracing in Operate First

- [Smaug: Installed Operators](#)
 - [Smaug: otel-dev project](#)
 - [ArgoCD otel-dev application](#)
 - [hello-opentelemetry example instrumentation](#)
 - <https://www.operate-first.cloud/blog/distributed-tracing>
 - [More interesting example application](#)
-

Thank you!

- <https://opentelemetry.io/>
- <https://github.com/open-telemetry/opentelemetry-go>
- <https://github.com/open-telemetry/opentelemetry-collector>
- <https://www.jaegertracing.io/docs/1.38/getting-started/>

