



OpenTelemetry

With aws

Luke Stoward





Senior Cloud Service Engineer 



@LukeStoward



@LukeStoward



4+ years



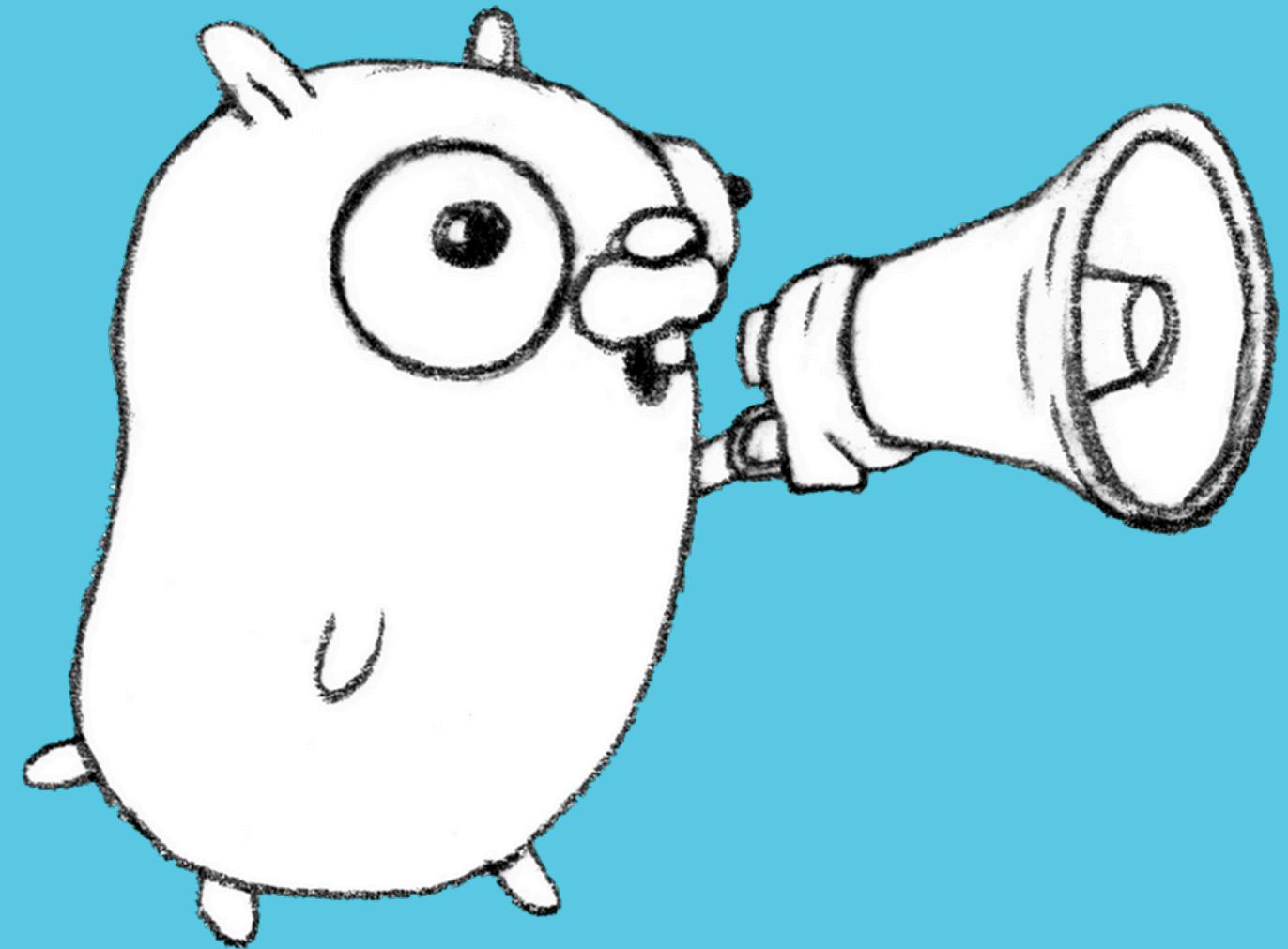
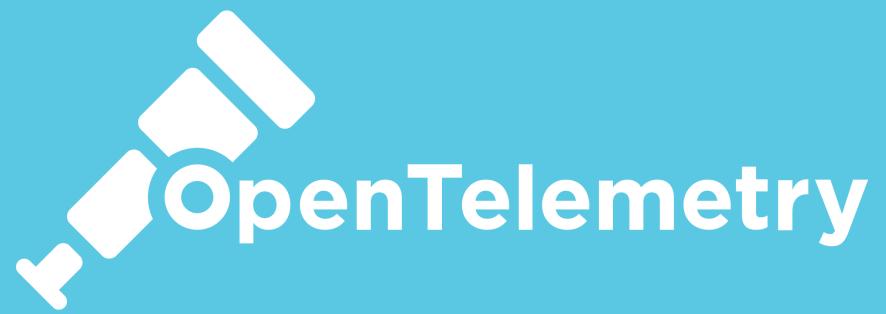


Illustration by Renee French (CC 3.0)

- What is OpenTelemetry?
- OpenTelemetry Tracing
- OpenTelemetry Collector
- AWS X-Ray





What is OpenTelemetry?



“OpenTelemetry is the mechanism by which code is instrumented, to help make a system observable”



OpenTelemetry

- Observability
- Standardisation
- Merger of OpenTracing & OpenCensus
- CNCF Incubating Project

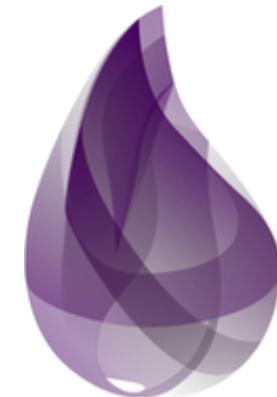
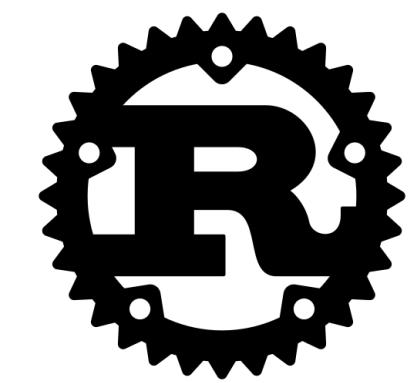
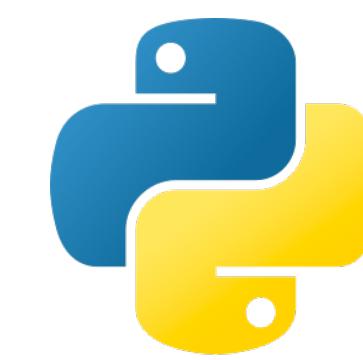


www.cncf.io/projects/opentelemetry





Supported Languages



opentelemetry.io/docs/instrumentation





Signals

Traces

Metrics

Logs



Signals

Traces

- ✓ API: Stable
- ✓ SDK: Stable

Metrics

- ✓ API: **Stable**
- ⚠ SDK: Mixed

Logs

- ✗ API: Draft
- ✗ SDK: Draft



Signals

Traces

- ✓ API: Stable
- ✓ SDK: Stable

Metrics

- ✓ API: **Stable**
- ⚠ SDK: Mixed

Logs

- ✗ API: Draft
- ✗ SDK: Draft



Baggage



OpenTelemetry Tracing





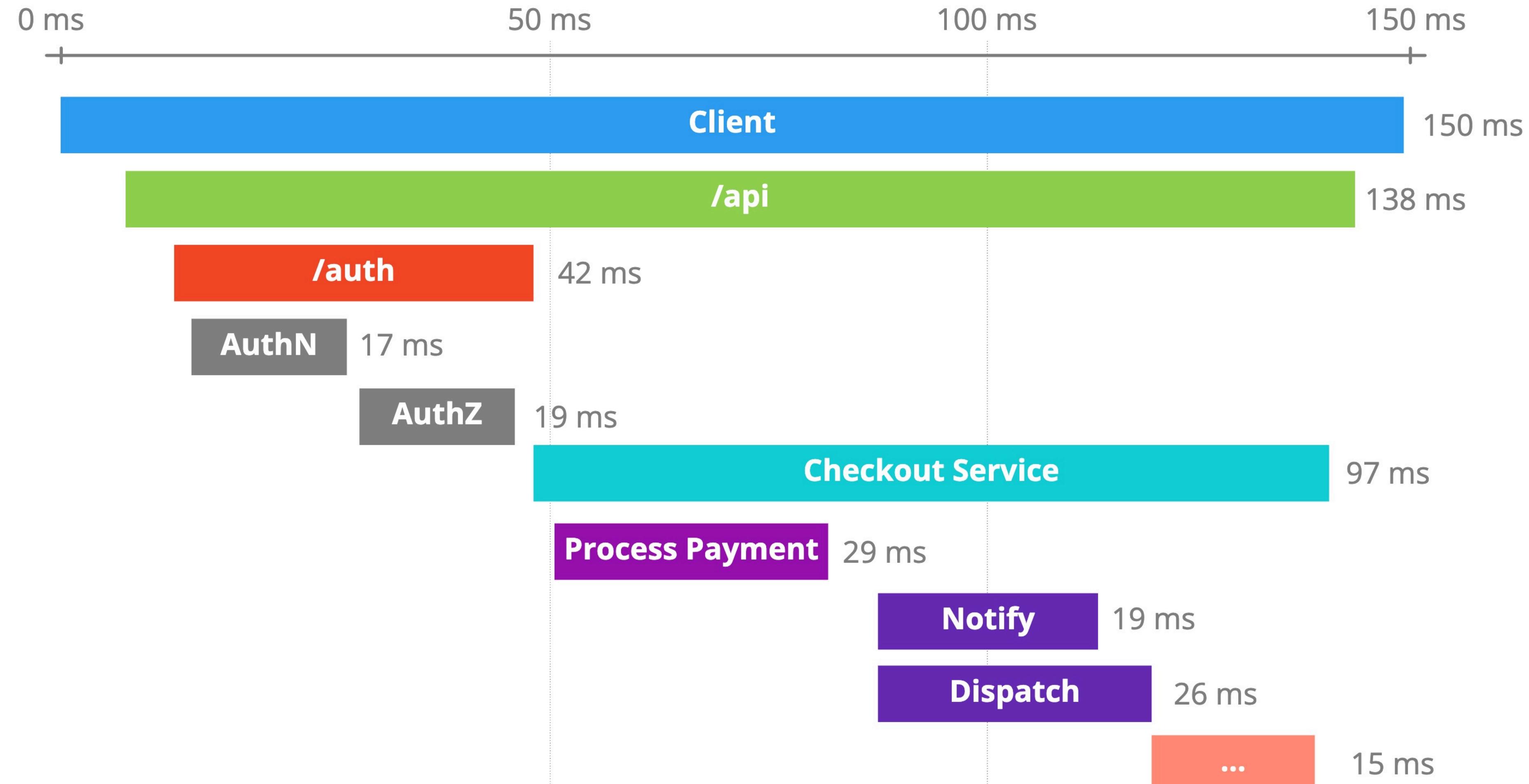
What is a Trace?

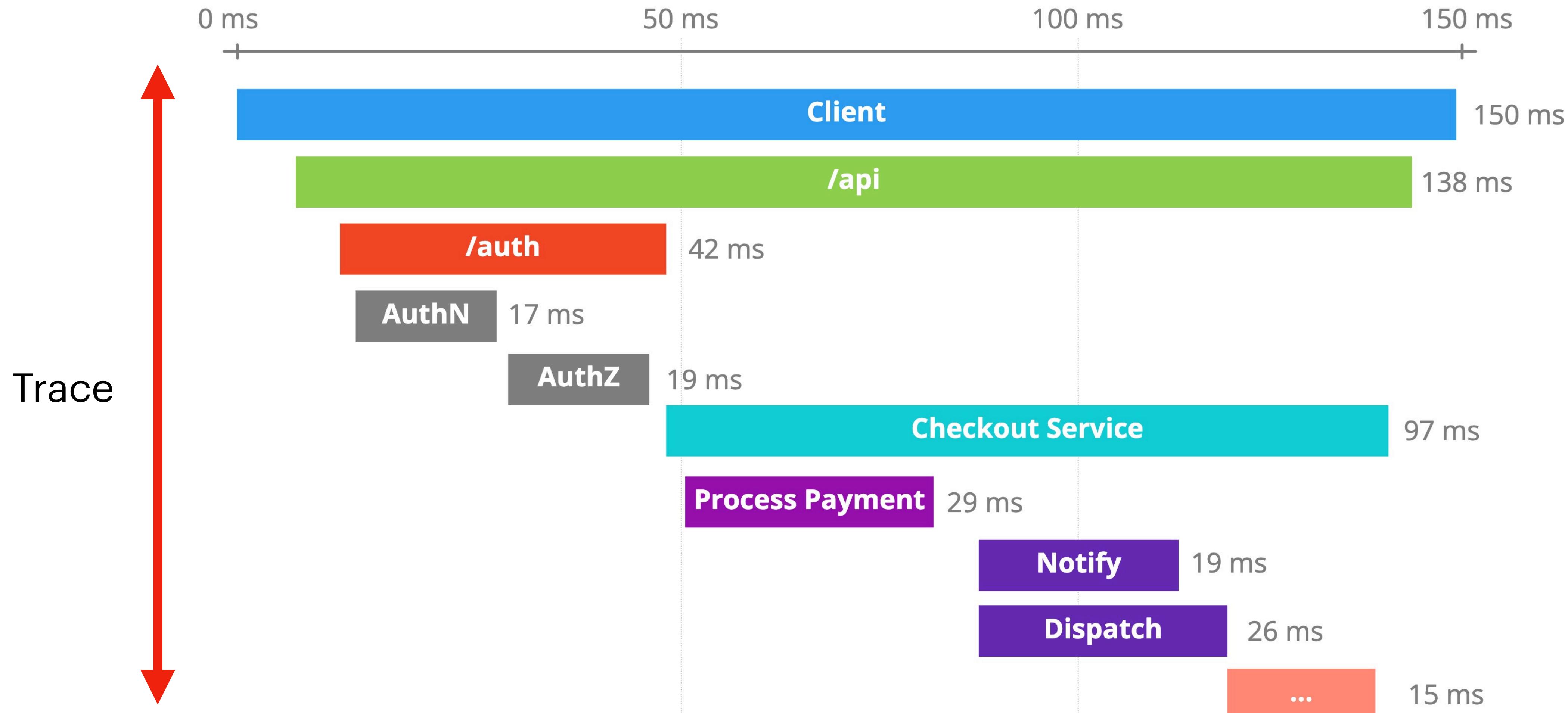
A **Trace** captures the sequence of events across all services for a given transaction.

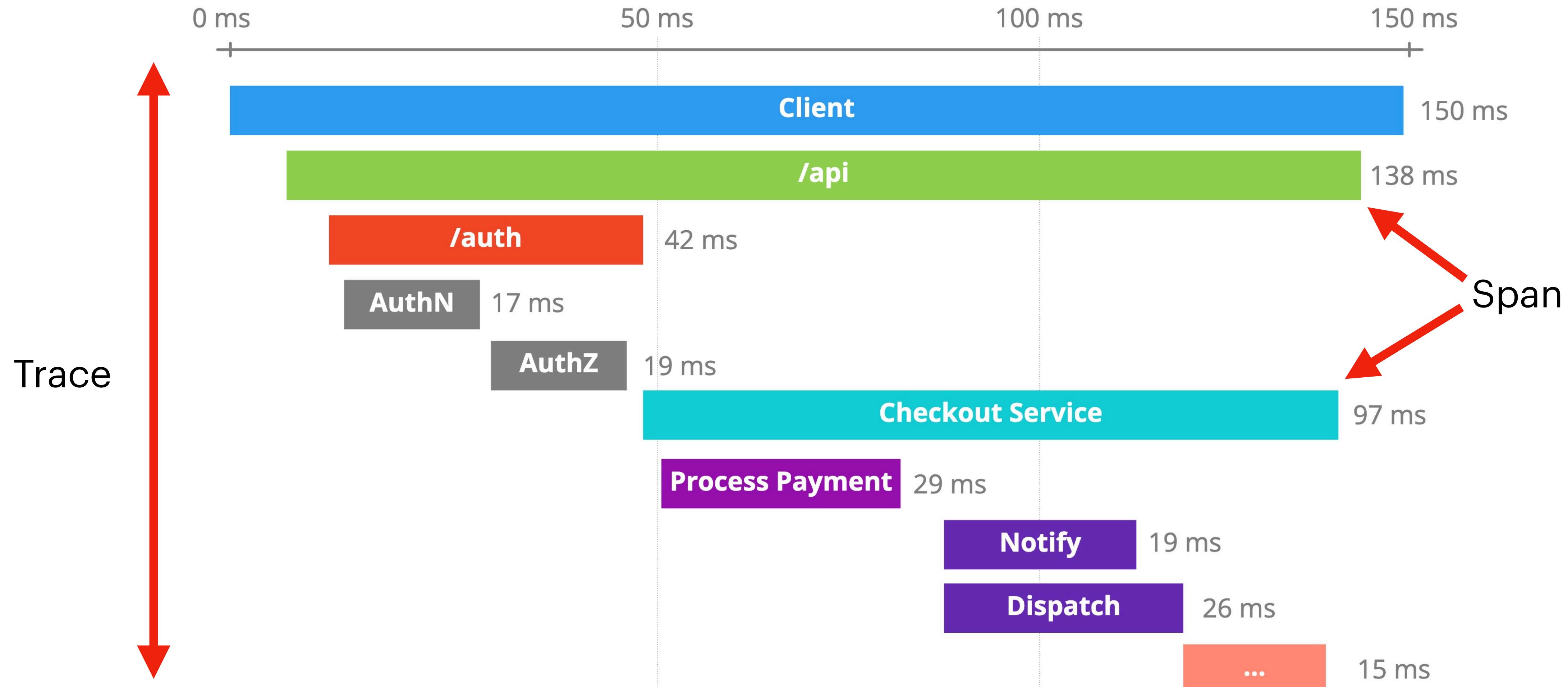
A **Trace** is comprised of one or more **Spans**.

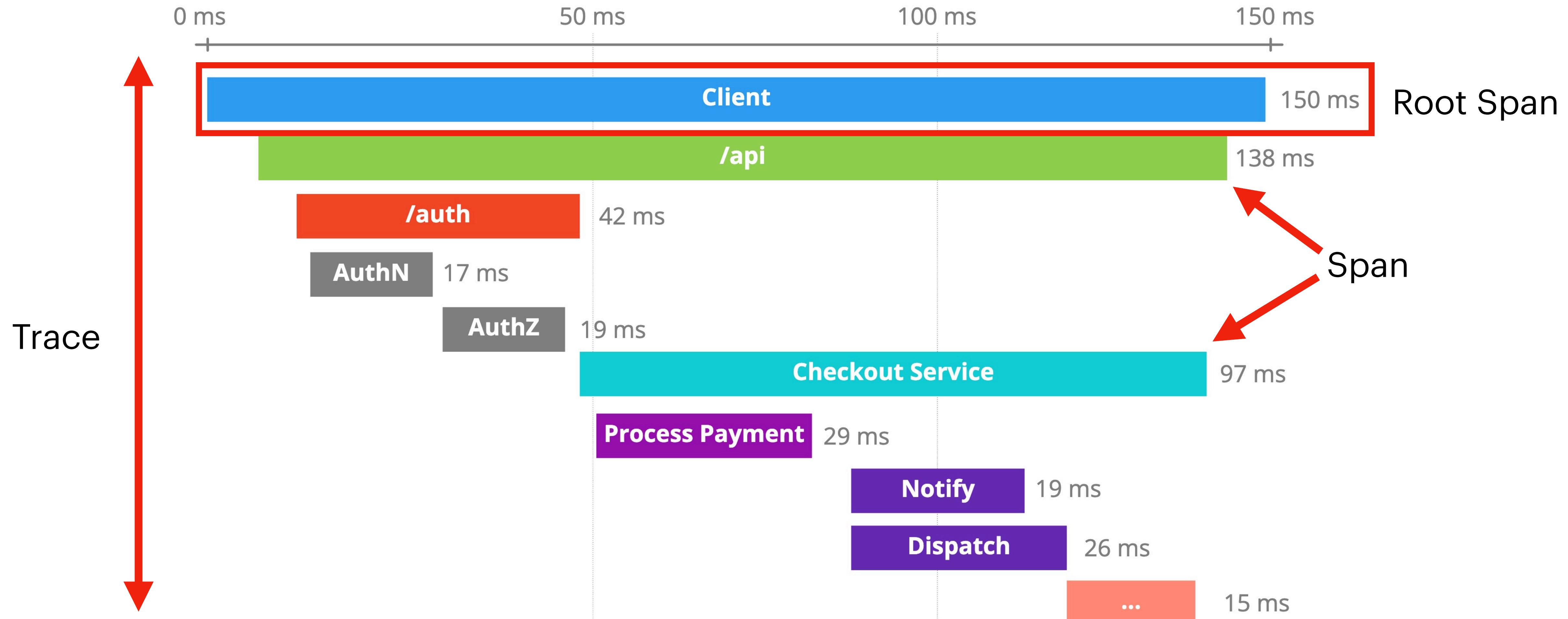
A **Span** represents a unit of work or operation and captures time-related data and attributes.

```
{  
  "trace_id": "7bba9f33312b3dbb8b2c2c62bb7abe2d",  
  "parent_id": "",  
  "span_id": "086e83747d0e381e",  
  "name": "/v1/sys/health",  
  "start_time": "2021-10-22 16:04:01 +0000 UTC",  
  "end_time": "2021-10-22 16:04:01 +0000 UTC",  
  "status_code": "STATUS_CODE_OK",  
  "status_message": "",  
  "attributes": {  
    "net.transport": "IP.TCP",  
    "net.peer.ip": "172.17.0.1",  
    "net.peer.port": "51820",  
    "net.host.ip": "10.177.2.152",  
    "net.host.port": "26040",  
    "http.method": "GET",  
    "http.target": "/v1/sys/health",  
    "http.server_name": "mortar-gateway",  
    "http.route": "/v1/sys/health",  
    "http.user_agent": "Consul Health Check",  
    "http.scheme": "http",  
    "http.host": "10.177.2.152:26040",  
    "http.flavor": "1.1"  
  }  
}
```











Why Should You Trace?

- End-to-end visibility of your distributed systems.
- Understand service dependencies & relationships.
- Identify performance bottlenecks.
- Allows teams to speak the same language.





Tracing Considerations



Overhead



Cost



Trace Sampling

“Sampling is a process that restricts the amount of trace data that is generated by a system”



Trace Sampling

Where: Head-based or Tail-based.

When: Samplers

“Always On”- Sample all spans (default).

“Parent Based” - Sample if the parent is sampled.

“TraceID Ratio Based” - Sample using a ratio value (1/10 spans).

“Remote Based” - Sample dynamically based on remote rule set.



Setup Demo

github.com/lukestoward/otel-with-aws-presentation-go





OpenTelemetry Collector

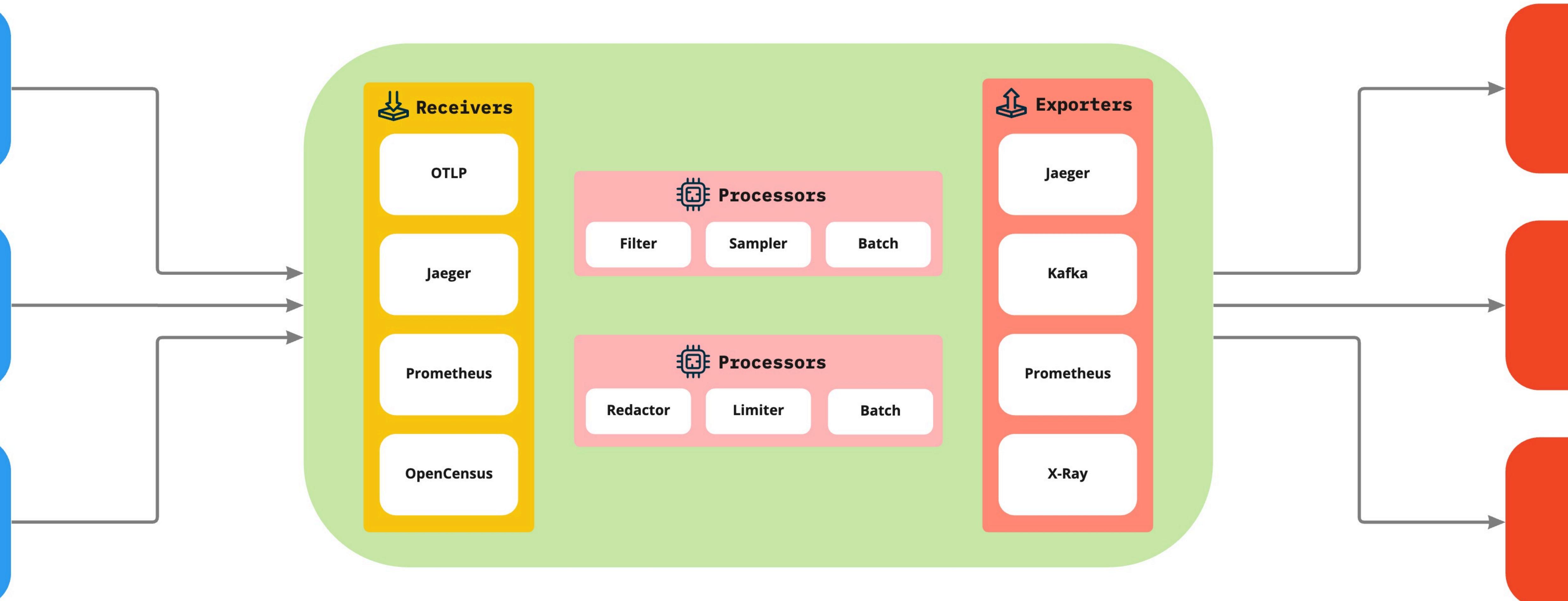




The Collector



The Collector Components



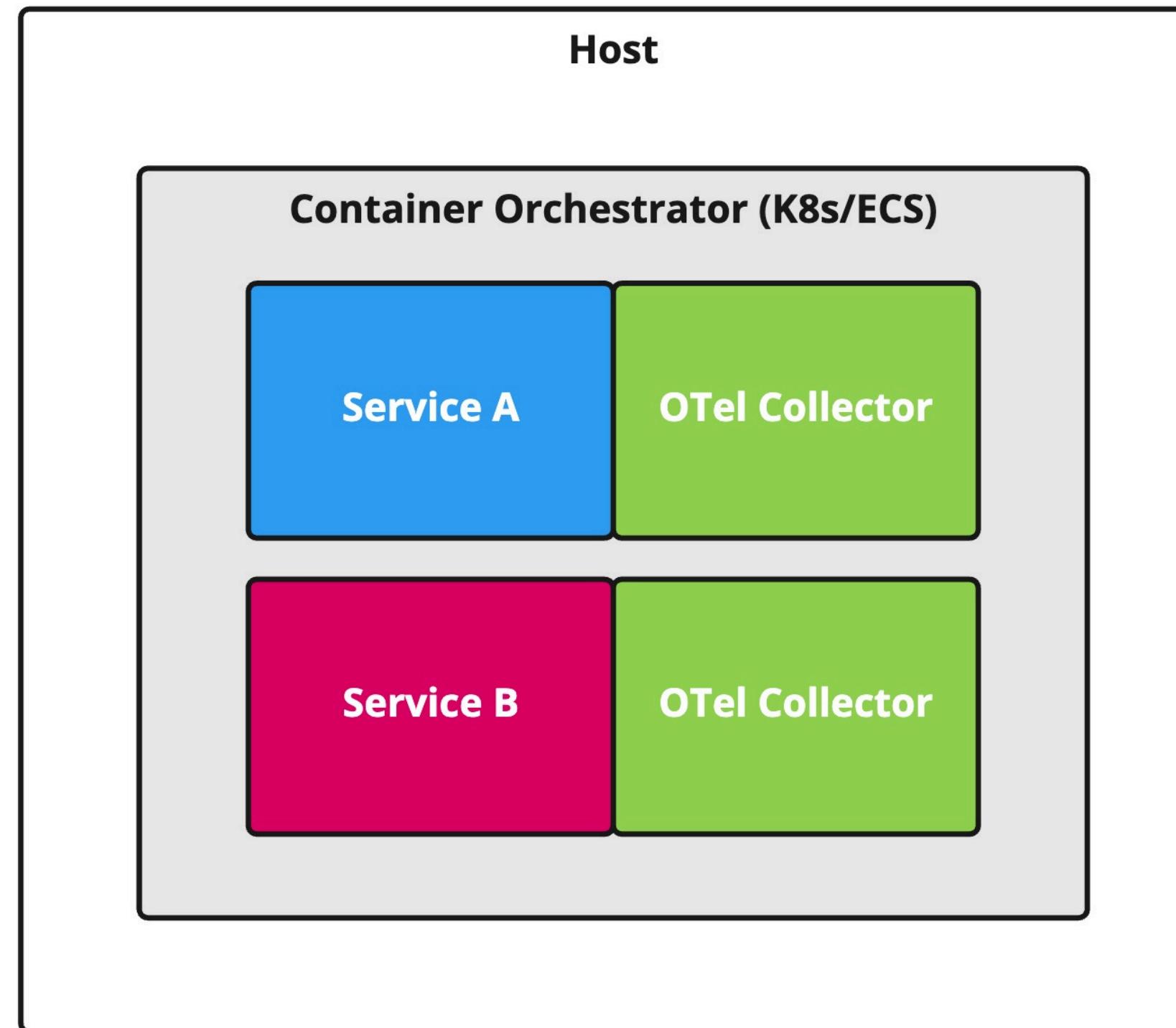


Why Use The Collector?

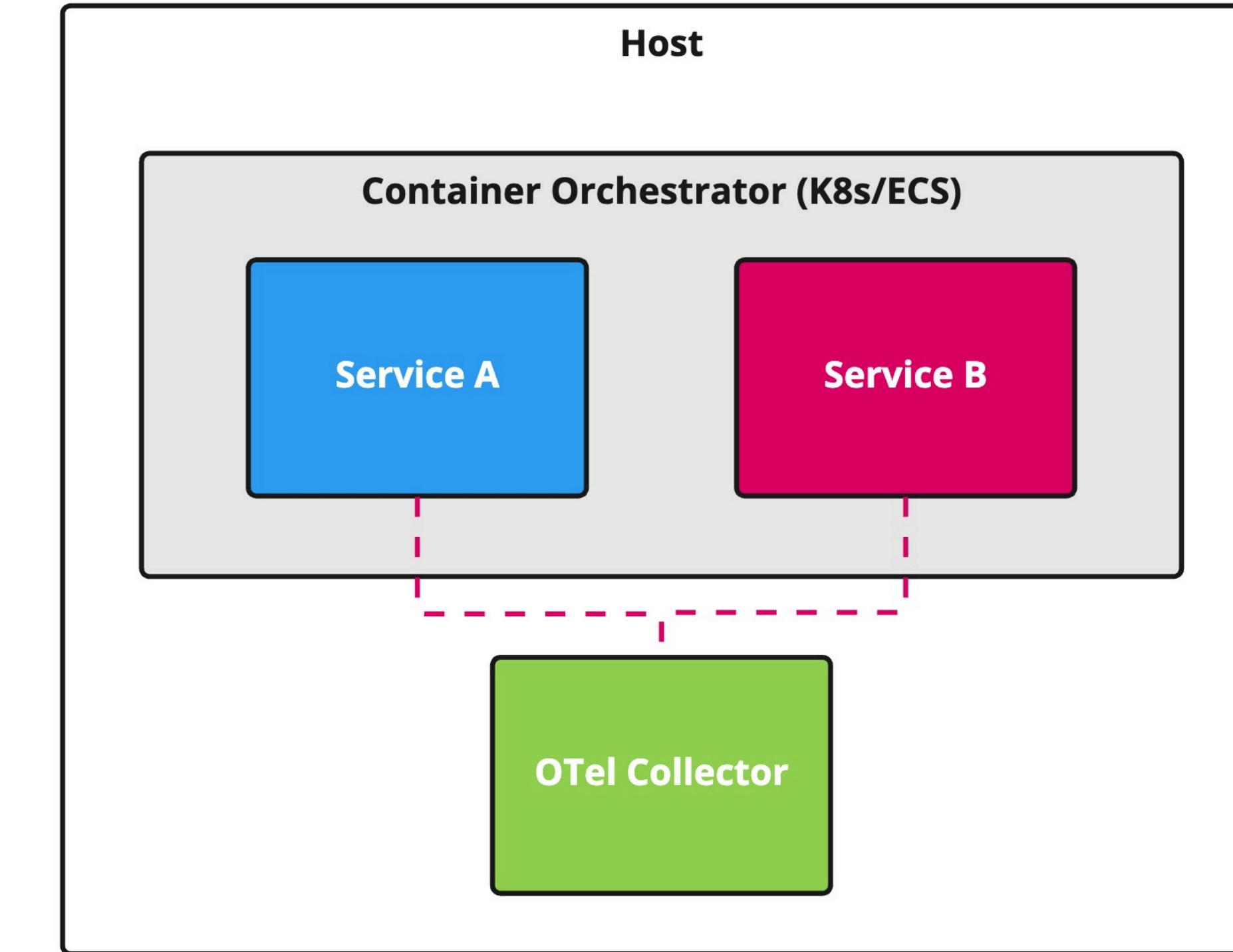
- Offload telemetry data quickly.
- Delegate retry logic, batching and encryption.
- Avoid managing multiple vendor specific agents/collectors.
- Easily add new data backends without service changes.
- Flexible and extensible data processing pipelines.



Deployment Strategies (Agent)

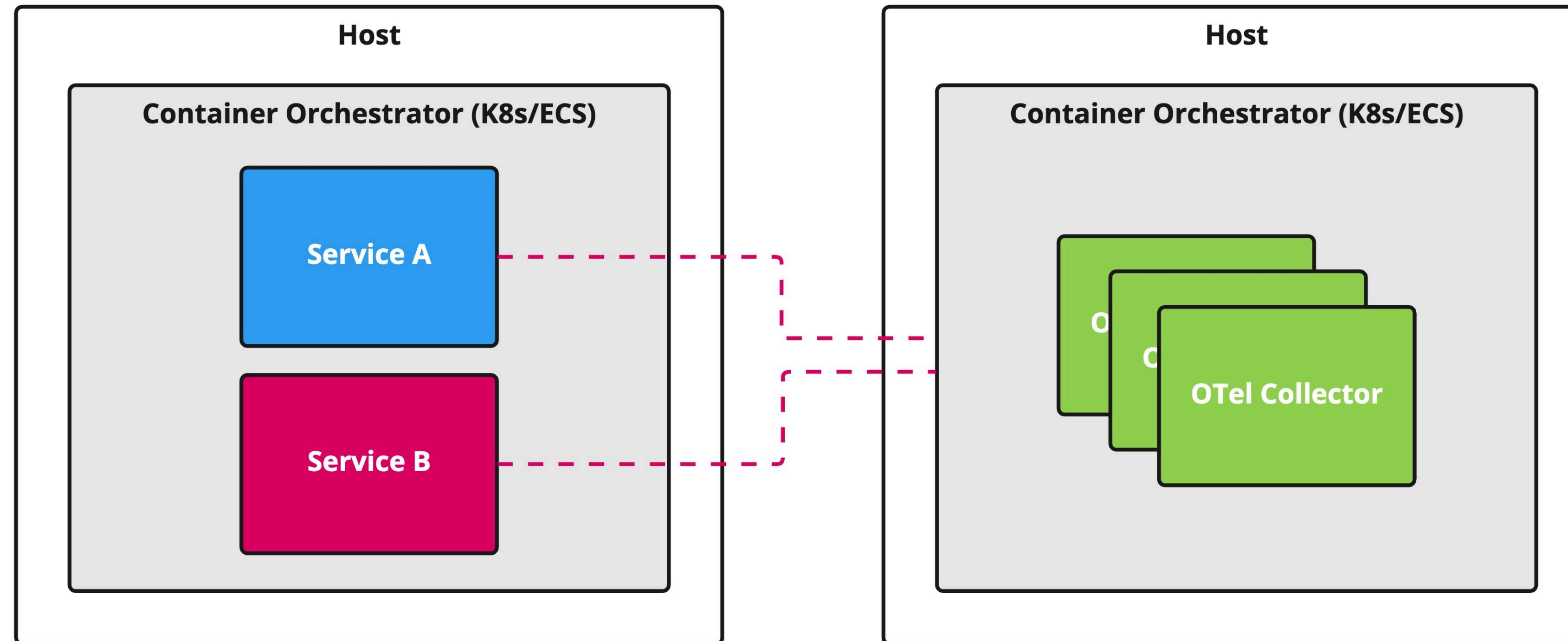


Sidecar



Daemon

Deployment Strategies (Gateway)

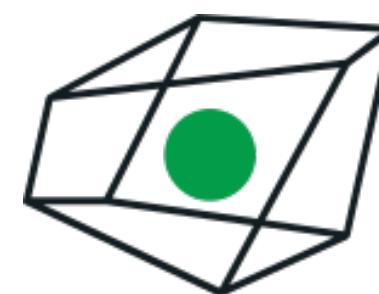




Distributions

splunk®>

 Uptrace



Lightstep

aws

 Aspecto

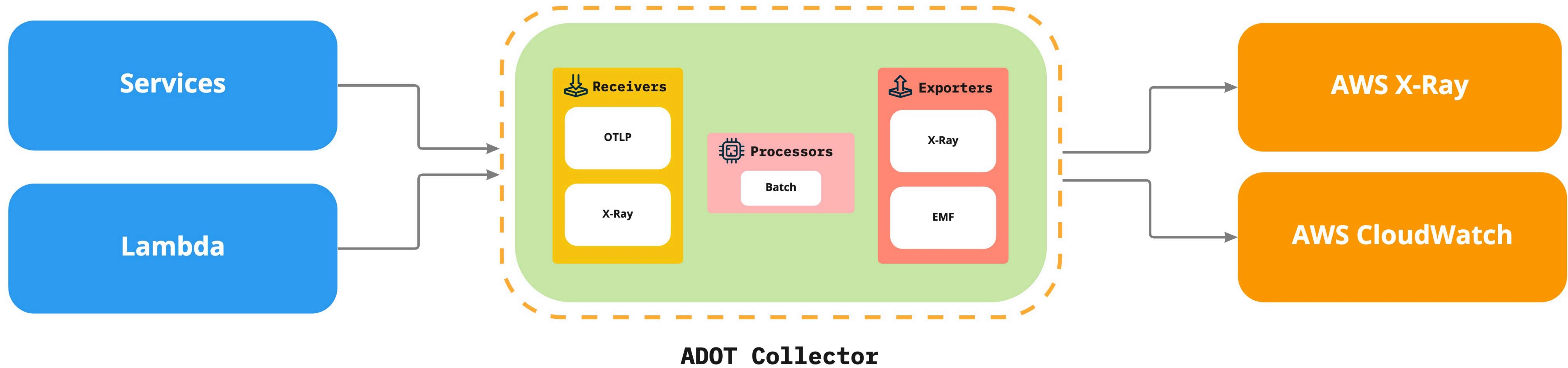


sumo logic

 DATADOG



AWS Distro for OpenTelemetry



github.com/aws-observability/aws-otel-collector





AWS X-Ray





Live Demo

What could possibly go wrong... 😬





Resources

- **Slides/Code:** github.com/lukestoward/otel-with-aws-presentation-go
- **OTel Docs:** opentelemetry.io/docs
- **OTel Contrib (Go):** github.com/open-telemetry/opentelemetry-go-contrib
- **AWS ADOT:** aws-otel.github.io/
- **The Future of Observability with OpenTelemetry (Ted Young):**
go.lightstep.com/register-oreilly-future-of-observability.html





Want to get involved?



youview

We're Hiring!

