CLOUD NATIVE
COMPUTING FOUNDATION

Jaeger
Project Deep Dive

Pavol Loffay (Red Hat), Yuri Shkuro (Uber)

CloudNativeCon NA, San Diego, Nov-20-2019

Agenda

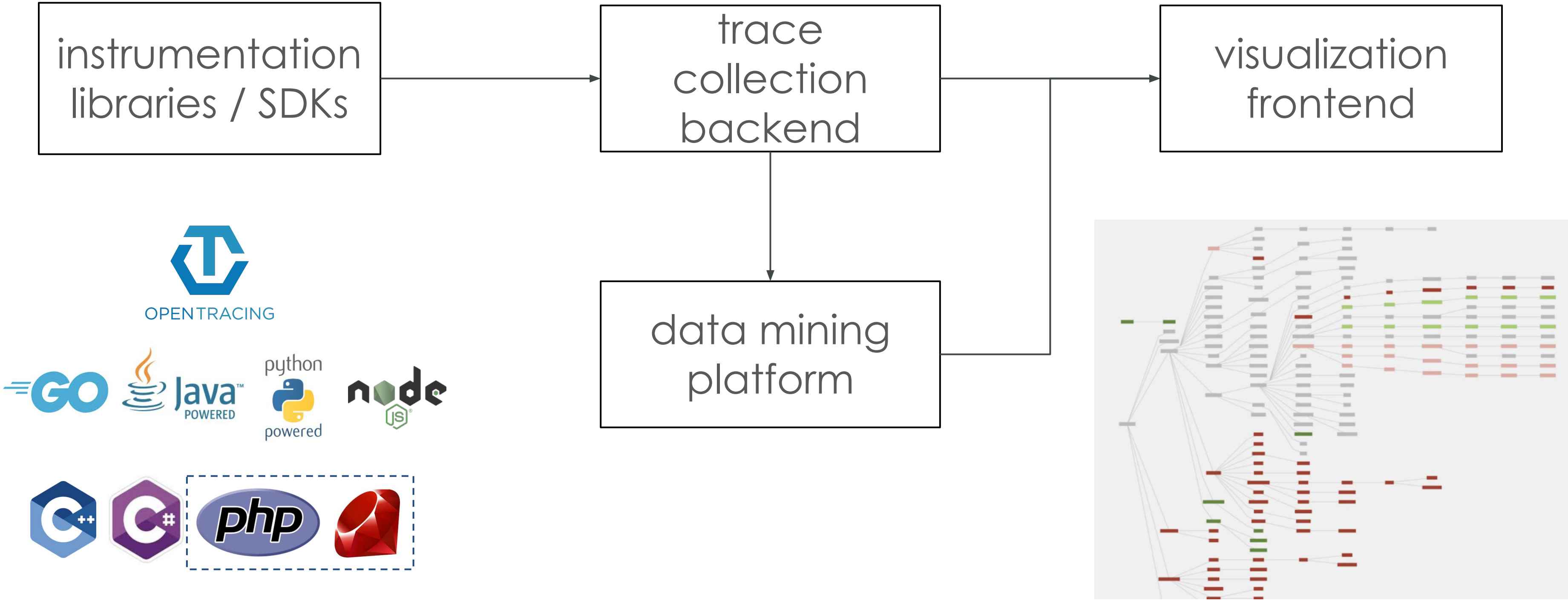
- About the project
- How tracing *really* works
- New features
- Roadmap
- Q & A

About

- Pavol Loffay (https://github.com/pavolloffay)
  - Software engineer at Red Hat
  - Maintainer of Jaeger, OpenTracing, OpenTelemetry

- Yuri Shkuro (https://github.com/yurishkuro)
  - Software engineer at Uber Technologies
  - Maintainer of Jaeger, OpenTracing, OpenTelemetry
  - Author of “Mastering Distributed Tracing” book

Jaeger, a Distributed Tracing Platform



instrumentation libraries / SDKs

trace collection backend

visualization frontend

data mining platform

Jaeger - /ˈyāgər/, *noun*: hunter

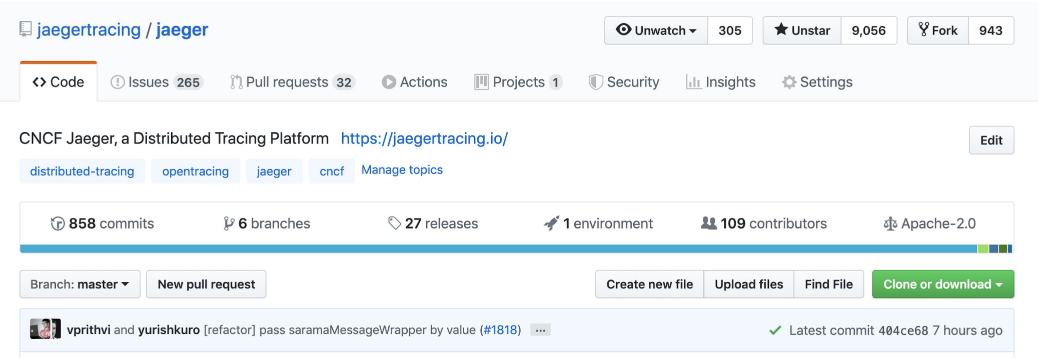
- Inspired by Google's Dapper and OpenZipkin
- Created at Uber in August 2015
- Open sourced in April 2017
- Joined CNCF in Sep 2017 (incubating)
- Graduated to top-level CNCF project Oct 31, 2019 ([CNCF announcement]())

Technology Stack

- Go backend
- Pluggable storage
  - Cassandra, Elasticsearch, badger, memory
- React/Javascript frontend
- OpenTracing Instrumentation libraries
- Integration with Kafka, Apache Flink

Project & Community

- **9,000+ GH stars**
- **1200+ contributors**
- **375 authors of commits and pull requests**
- **15 maintainers** across all components from 5+ companies (backend: 7 and 3 respectively)
- **815** Gitter channel members
- **2,800+** Twitter followers
- **15 releases** since incubation
- **10M+** Docker pulls

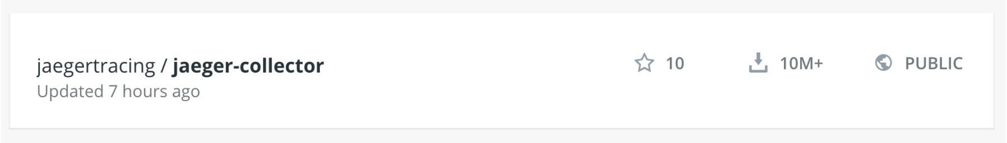


jaegertracing / **jaeger**

CNCF Jaeger, a Distributed Tracing Platform https://jaegertracing.io/

858 commits · **6** branches · **27** releases · **1** environment · **109** contributors · Apache-2.0

vprithvi and **yurishkuro** [refactor] pass saramaMessageWrapper by value (#1818) ··· ✓ Latest commit 404ce68 7 hours ago



jaegertracing / **jaeger-collector**
Updated 7 hours ago

☆ 10 · 10M+ · PUBLIC

CLOUD NATIVE
COMPUTING FOUNDATION

How Jaeger Works

Under the hood

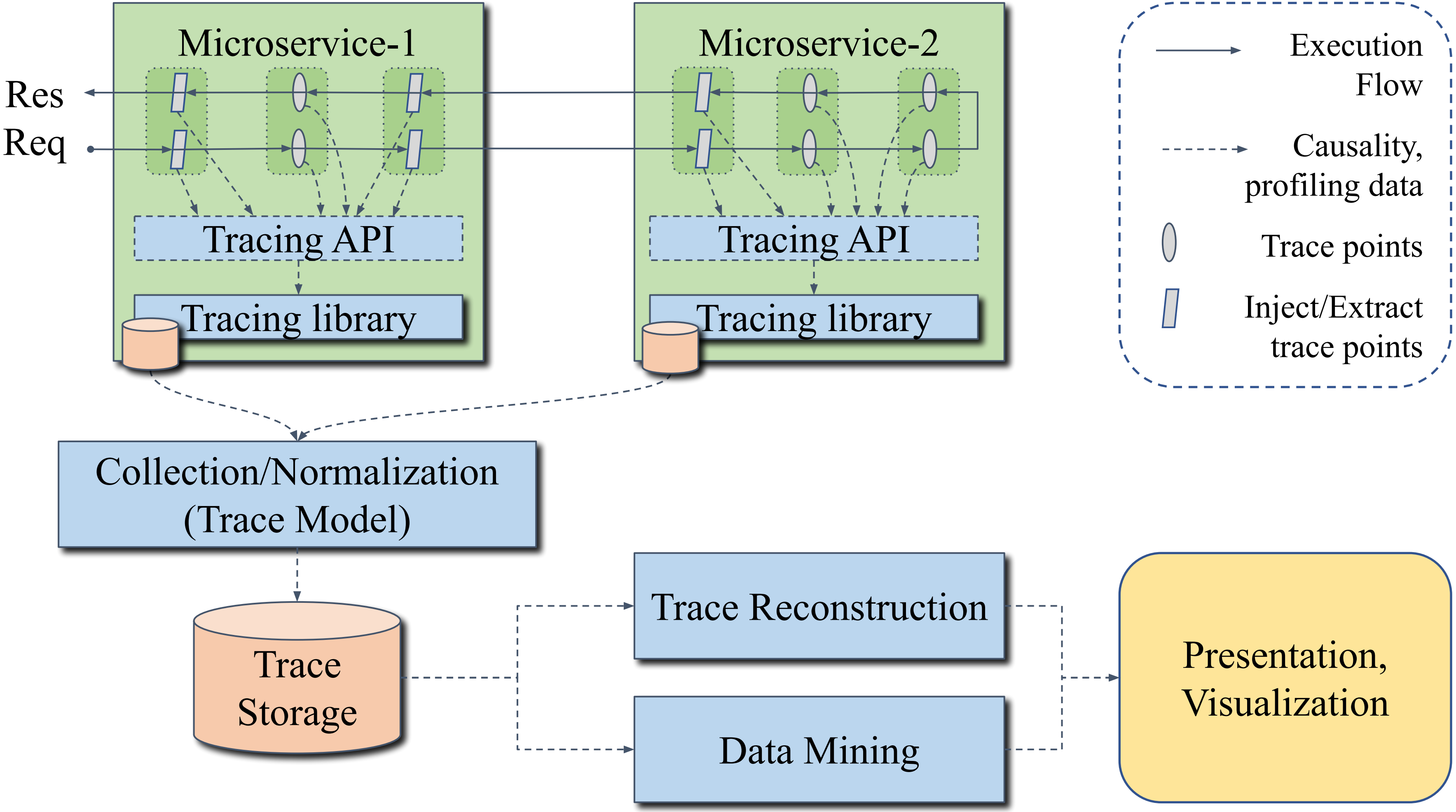
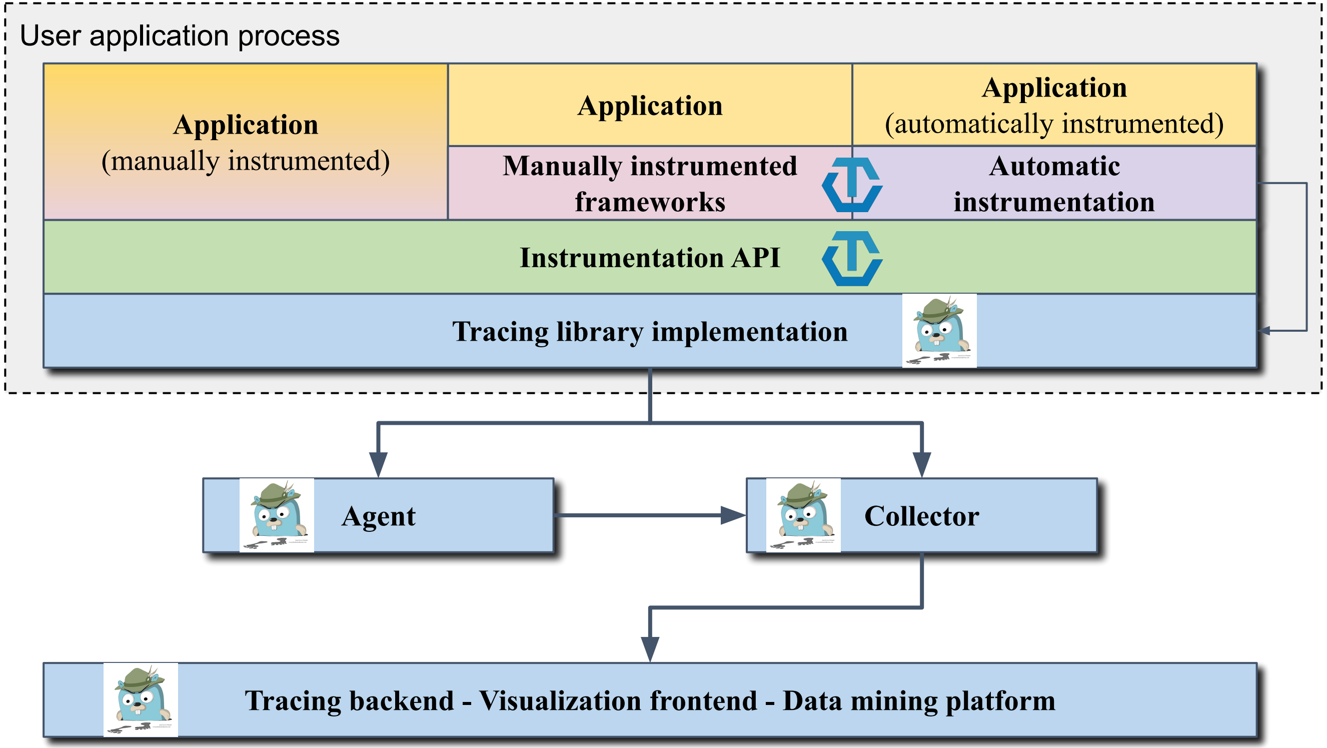


Diagram reproduced from the book Mastering Distributed Tracing, Chapter 3, by Yuri Shkuro

Jaeger vs. OpenTracing



User application process

Application (manually instrumented)

Application

Manually instrumented frameworks

Application (automatically instrumented)

Automatic instrumentation

Instrumentation API

Tracing library implementation

Agent

Collector

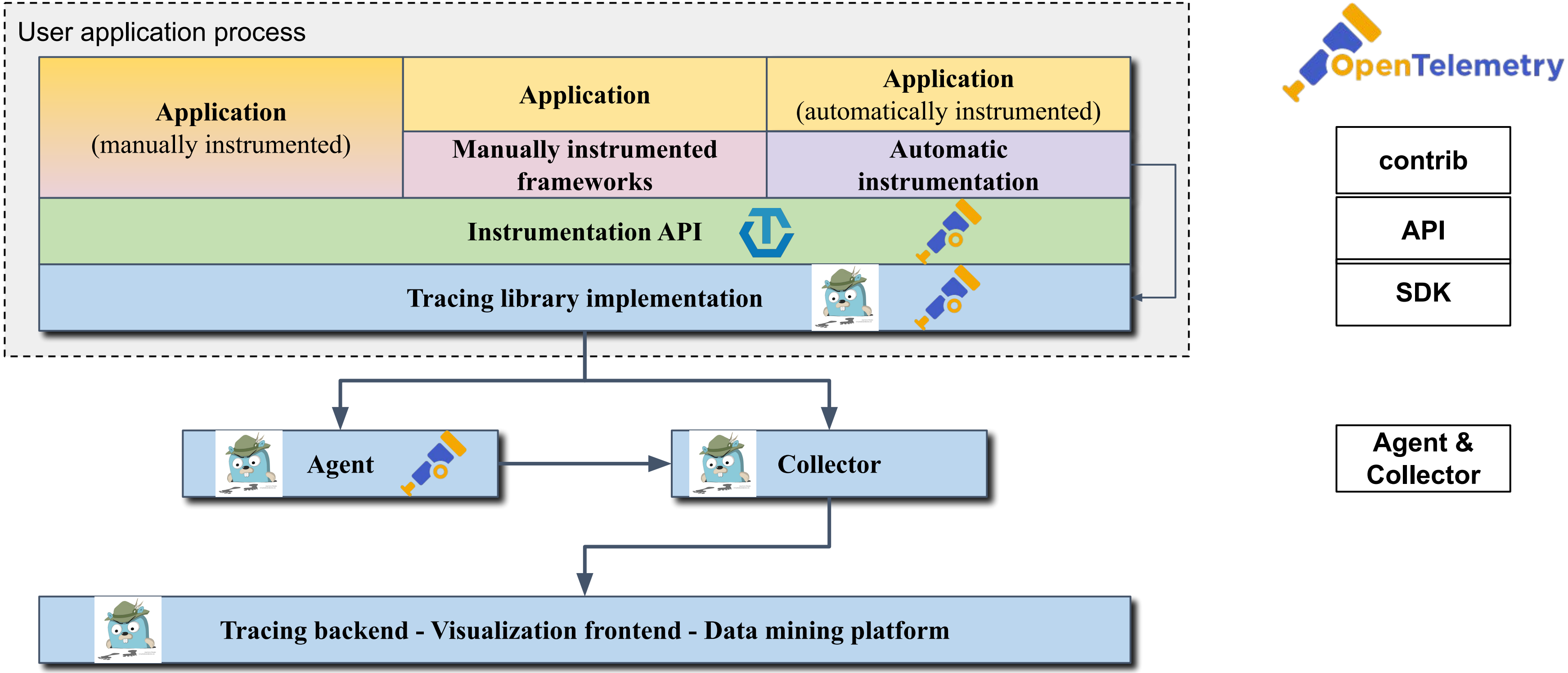
Tracing backend - Visualization frontend - Data mining platform

OPENTRACING

opentracing contrib

API

Jaeger vs. OpenTelemetry



User application process
Application
(manually instrumented)
Application
Manually instrumented frameworks
Application
(automatically instrumented)
Automatic instrumentation
Instrumentation API
Tracing library implementation
Agent
Collector
Tracing backend - Visualization frontend - Data mining platform
OpenTelemetry
contrib
API
SDK
Agent & Collector

CLOUD NATIVE
COMPUTING FOUNDATION

Sampling

(Too much data)

Sampling

- Head-based sampling
  - Probabilistic, rate limiting
- Endpoint-level sampling
- Centrally controlled
- (roadmap) Tag-matching sampling
- (roadmap) Tail-based

CLOUD NATIVE
COMPUTING FOUNDATION

Integrations

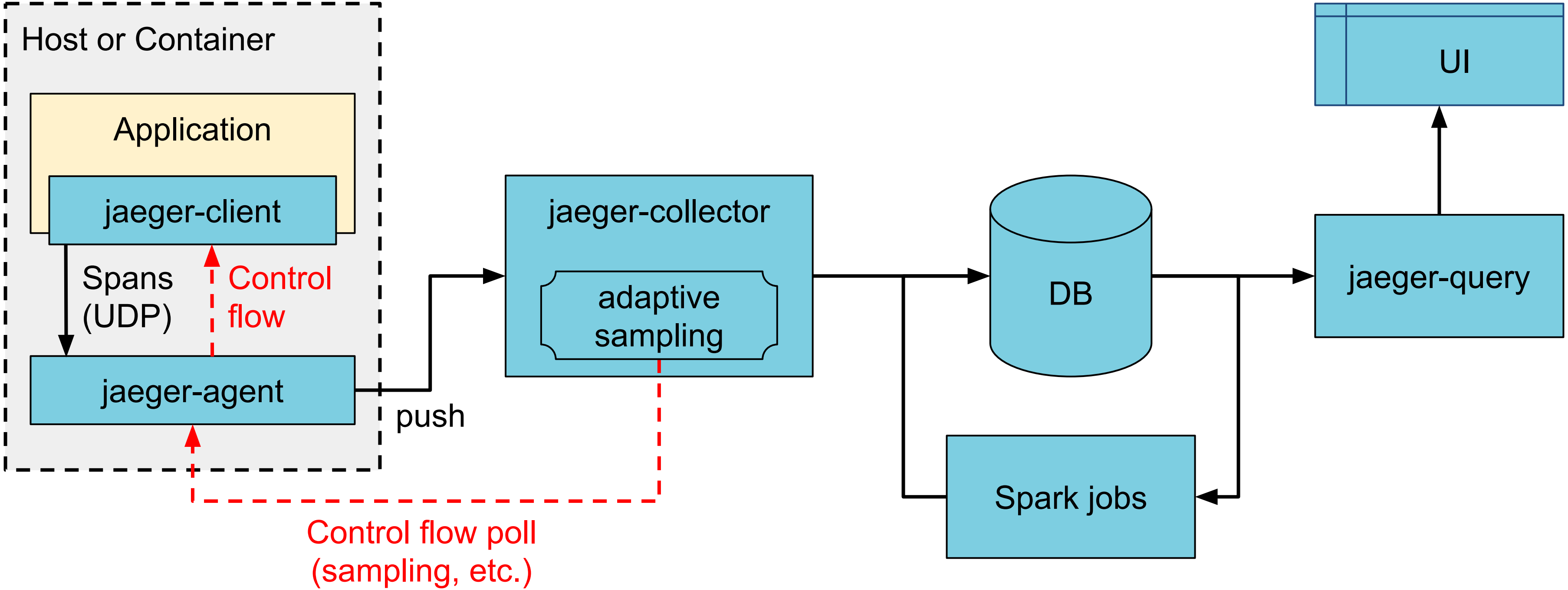
Integrations with other CNCF projects

- Jaeger deployment on Kubernetes
  - Operator, manifests, and Helm Chart
- Jaeger defaults to exposing metrics in Prometheus format
  - https://www.jaegertracing.io/docs/1.15/monitoring/
- OpenTelemetry will ship with exporters for Jaeger
- Linkerd and Istio come with Jaeger included
  - https://istio.io/docs/tasks/telemetry/distributed-tracing/
  - https://linkerd.io/2019/10/07/a-guide-to-distributed-tracing-with-linkerd/
- Envoy works with Jaeger native C++ client
  - https://www.envoyproxy.io/docs/envoy/latest/start/sandboxes/jaeger_native_tracing

CLOUD NATIVE
COMPUTING FOUNDATION

Asynchronous Ingestion

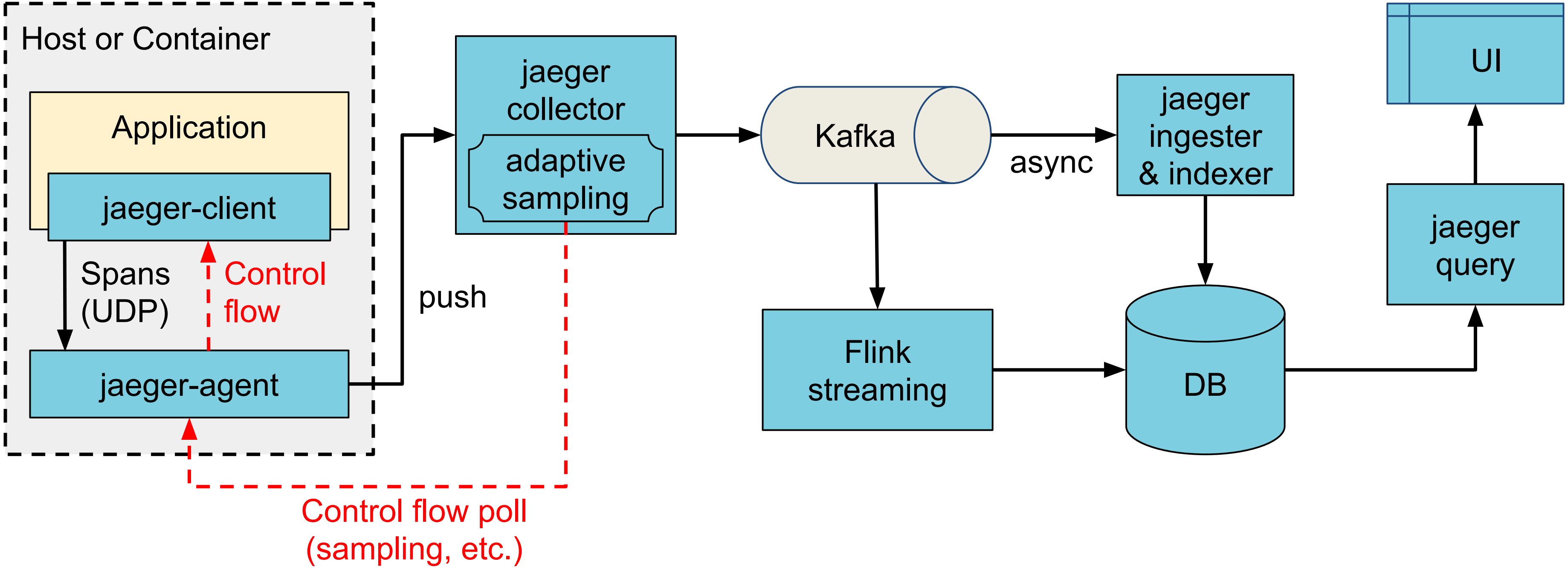
Architecture 2017: Push



Asynchronous span ingestion

- Push model was struggling to keep up with traffic spikes
  - Because of sync storage writes
  - Collectors had to drop data randomly
- Kafka is much more elastic for writes
  - Just raw bytes, no schema, no indexing
  - A lot less overhead on the write path
- Data in Kafka allows for streaming data mining & aggregations
- Two new components: `jaeger-ingester` and `jaeger-indexer`

Architecture now: Push+Async+Streaming



CLOUD NATIVE
COMPUTING FOUNDATION

Protobuf & gRPC

Enabling roadmap

Protobuf & gRPC

- Internal data model generated from Protobuf IDL
- gRPC connection between `jaeger-agent` and `jaeger-collector`

Why

- gRPC plays better with modern routing than TChannel
- Path to official data model and collector/query APIs
- Protobuf-based JSON API
- Unblock development of storage plugins
- (Thrift still supported for backwards compatibility)

CLOUD NATIVE
COMPUTING FOUNDATION

Zipkin Compatibility

Zipkin Compatibility

- Clients
  - Zipkin `B3-***` headers for context propagation
  - Interop between Jaeger-instrumented and Zipkin-instrumented apps
- Collector
  - Zipkin Thrift, Protobuf, and JSON v2 span format
  - Use Zipkin instrumentation (e.g. Brave) to send traces to Jaeger
- Kafka

CLOUD NATIVE
COMPUTING FOUNDATION

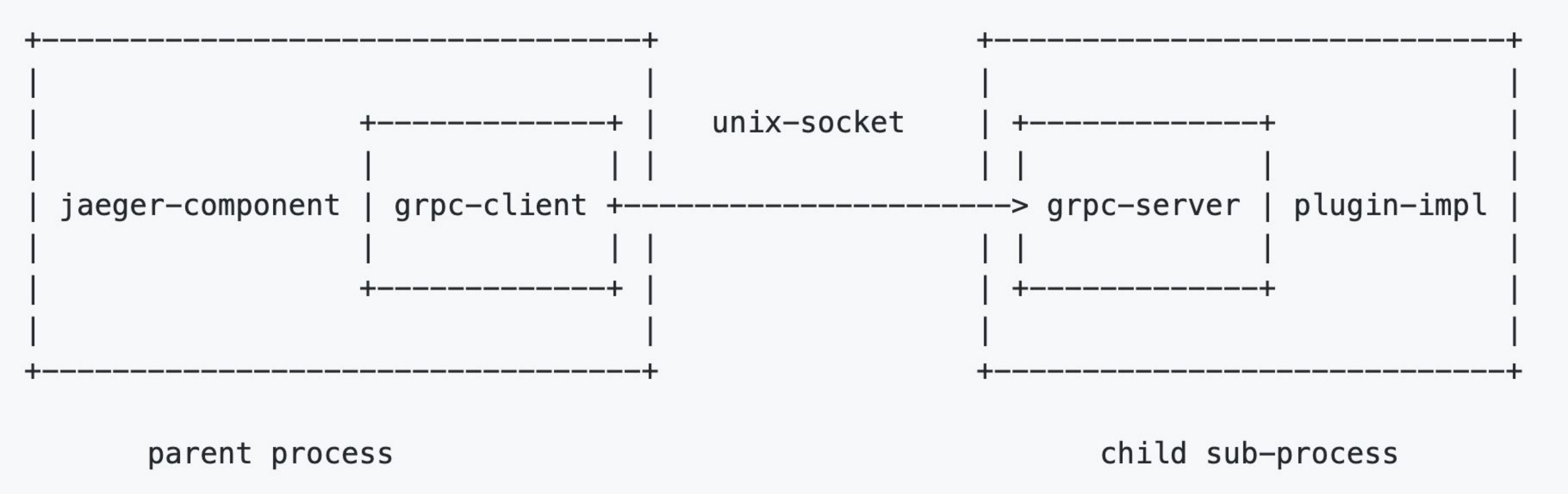
Jaeger 1.9 - 1.15

New Features

New Features

- Kubernetes Operator
- Badger embedded storage
- External storage plugins: Couchbase, InfluxDB
- Graph visualizations
- Ad-hoc & delayed sampling
- Security improvements
    - TLS with gRPC, Kafka, Elasticsearch

External storage plugins

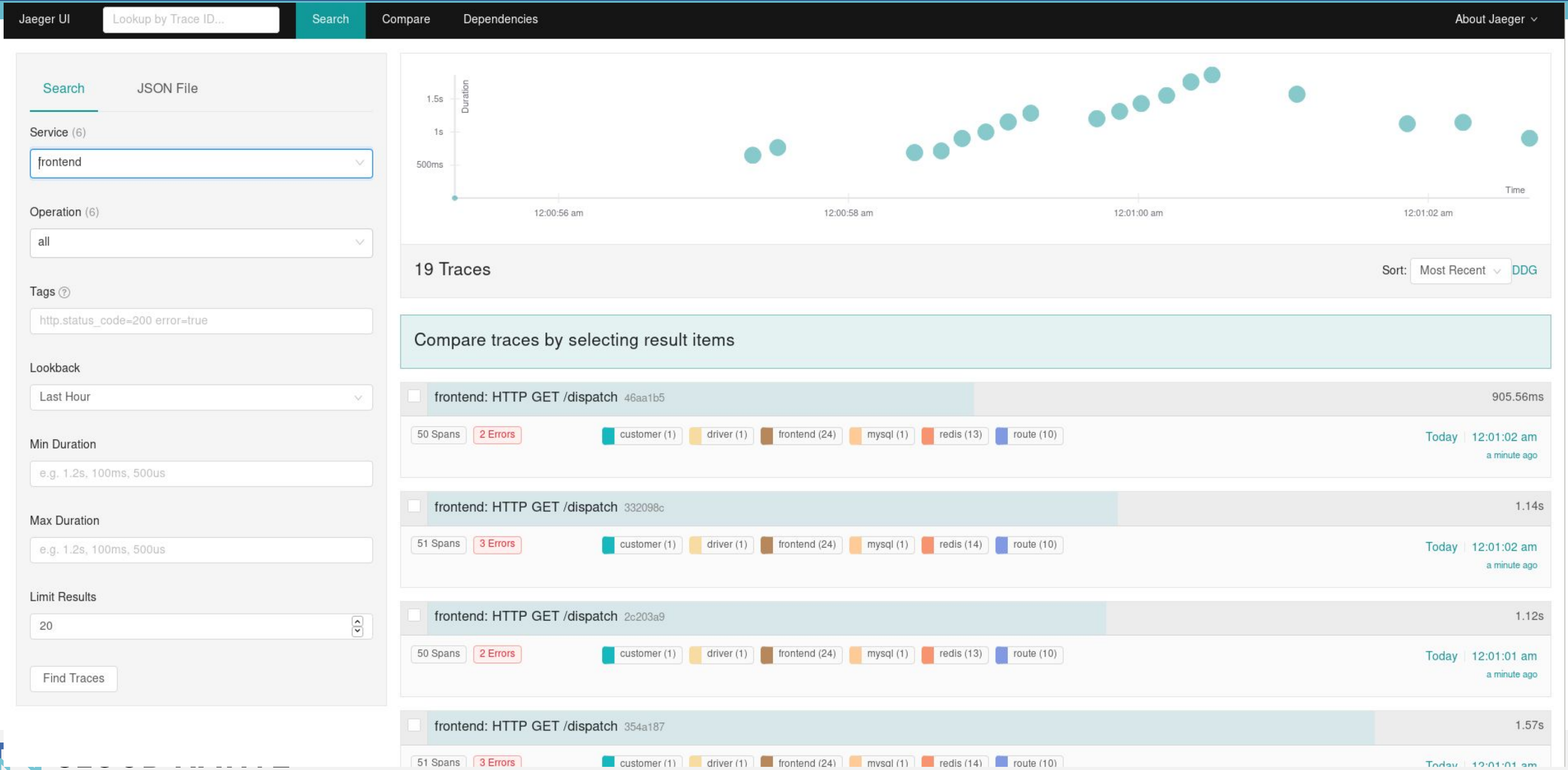


```
+----------------------------------+                  +-----------------------------+
|                                  |                  |                             |
|                  +-------------+ |   unix-socket    | +-------------+             |
|                  |             | |                  | |             |             |
| jaeger-component | grpc-client +----------------------> grpc-server | plugin-impl |
|                  |             | |                  | |             |             |
|                  +-------------+ |                  | +-------------+             |
|                                  |                  |                             |
+----------------------------------+                  +-----------------------------+

       parent process                                        child sub-process
```

https://github.com/jaegertracing/jaeger/tree/master/plugin/storage/grpc

Graph Visualization of Multiple Traces



Search | JSON File

Service (6): frontend

Operation (6): all

Tags: http.status_code=200 error=true

Lookback: Last Hour

Min Duration: e.g. 1.2s, 100ms, 500us

Max Duration: e.g. 1.2s, 100ms, 500us

Limit Results: 20

Find Traces

19 Traces

Sort: Most Recent DDG

Compare traces by selecting result items

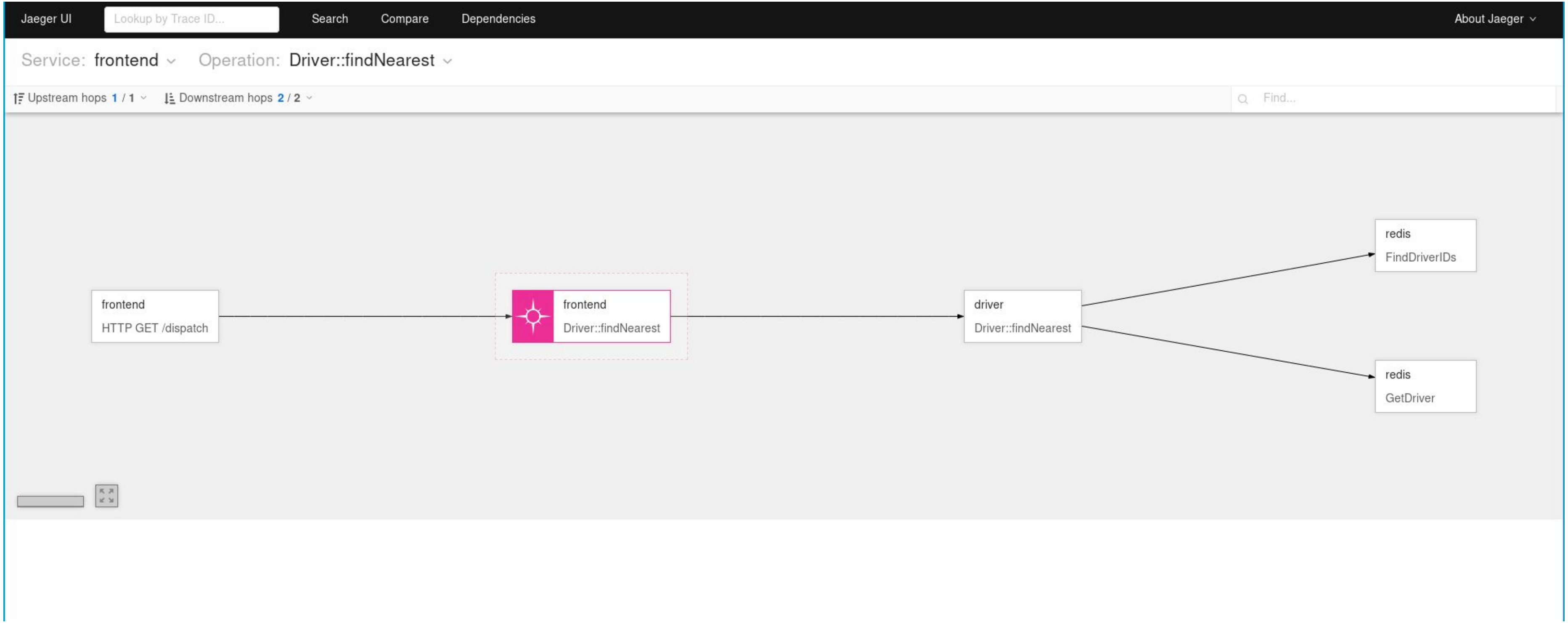
□ frontend: HTTP GET /dispatch 46aa1b5 — 905.56ms
50 Spans | 2 Errors | customer (1) | driver (1) | frontend (24) | mysql (1) | redis (13) | route (10) — Today | 12:01:02 am, a minute ago

□ frontend: HTTP GET /dispatch 332098c — 1.14s
51 Spans | 3 Errors | customer (1) | driver (1) | frontend (24) | mysql (1) | redis (14) | route (10) — Today | 12:01:02 am, a minute ago

□ frontend: HTTP GET /dispatch 2c203a9 — 1.12s
50 Spans | 2 Errors | customer (1) | driver (1) | frontend (24) | mysql (1) | redis (13) | route (10) — Today | 12:01:01 am, a minute ago

□ frontend: HTTP GET /dispatch 354a187 — 1.57s
51 Spans | 3 Errors | customer (1) | driver (1) | frontend (24) | mysql (1) | redis (14) | route (10) — Today | 12:01:01 am

Graph Visualization of Multiple Traces



CLOUD NATIVE
COMPUTING FOUNDATION

Roadmap

http://bit.do/jaeger-roadmap

Roadmap

- Trace DSL, jupyter notebooks and where we are heading
- Delayed & ad-hoc sampling
- Tail-based sampling
- OpenTelemetry

Adaptive Sampling

Problem

- APIs have endpoints with different QPS
- Service owners do not know the full impact of sampling probability

Adaptive Sampling is per service + endpoint, decided by Jaeger backend based on traffic

Adaptive Sampling Status

- Jaeger clients support per service/endpoint sampling strategies
- Can be statically configured in collector
- Pull requests for dynamic recalculations

Data Pipeline

- Based on Kafka and Apache Flink
- Support aggregations and data mining
- Examples:
  - Pairwise dependencies diagram
  - Path-based dependencies diagram
  - Latency histograms

Partial Spans (community driven)

- Add ability to store/retrieve partial spans
- Use case:
  - Certain workflows are hours long. Unfortunately spans are only emitted once after it’s Finished(). “Root span” is missing until the complete workflow is finished.

CLOUD NATIVE
COMPUTING FOUNDATION

Learn More

Website: jaegertracing.io/

Blog: medium.com/jaegertracing

Getting in Touch

- GitHub: https://github.com/jaegertracing
- Chat: https://gitter.im/jaegertracing/
- Mailing List - jaeger-tracing@googlegroups.com
- Blog: https://medium.com/jaegertracing
- Twitter: https://twitter.com/JaegerTracing
- Bi-Weekly Community Meetings

CLOUD NATIVE
COMPUTING FOUNDATION

Q & A

Open Discussion