CLOUD NATIVE
COMPUTING FOUNDATION

Introducing Jaeger 1.0

Yuri Shkuro (Uber Technologies)

CNCF Webinar Series, Jan-16-2018

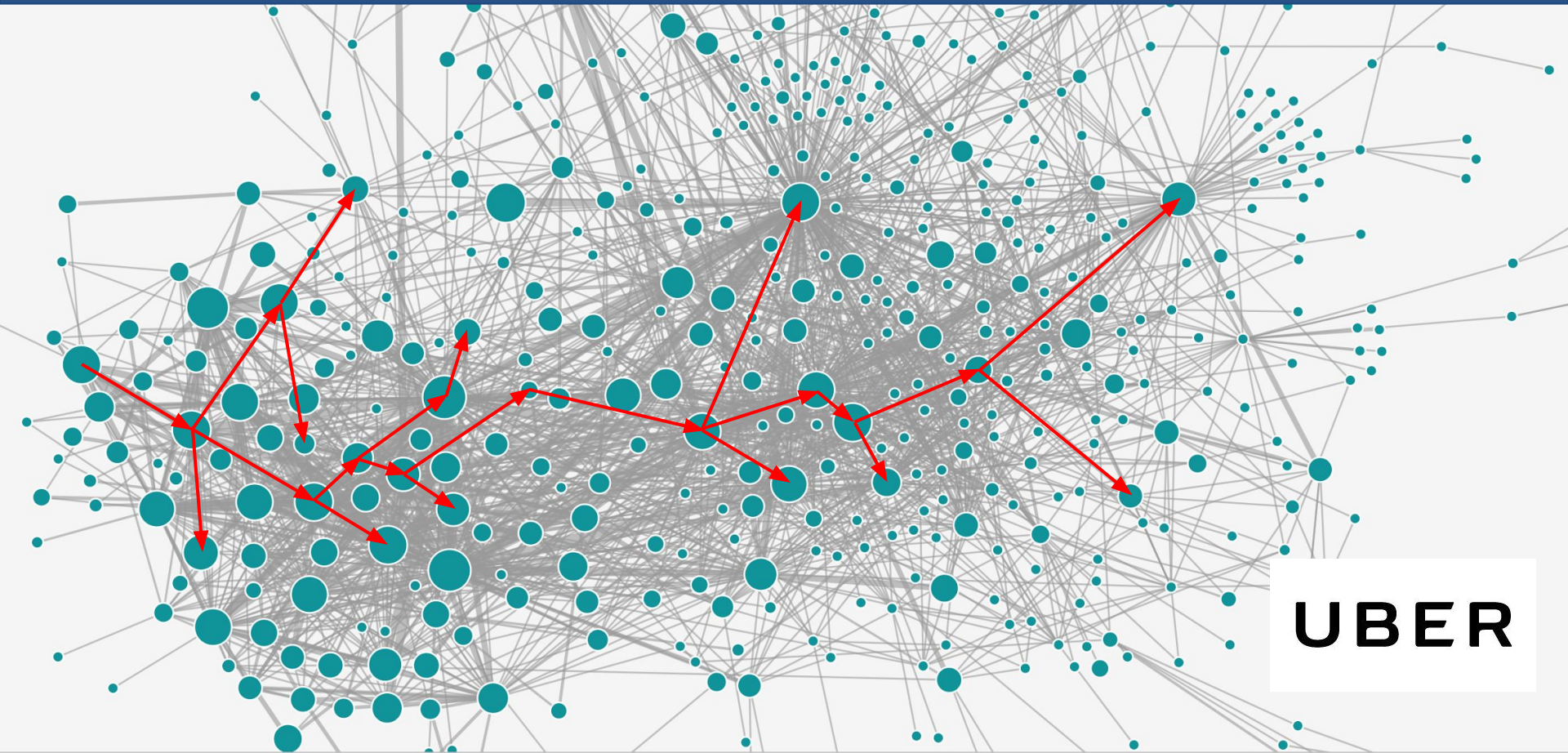
Agenda

- What is distributed tracing
- Jaeger in a HotROD
- Jaeger under the hood
- Jaeger v1.0
- Roadmap
- Project governance, public meetings, contributions
- Q & A

About

- Software engineer at Uber
  - NYC Observability team
- Founder of Jaeger
- Co-author of OpenTracing Specification

BILLIONS times a day!



How do we know what's going on?

We use MONITORING tools

Metrics / Stats

- Counters, timers, gauges, histograms
- Four golden signals
  - utilization
  - saturation
  - throughput
  - errors
- Prometheus, Grafana

Logging

- Application events
- Errors, stack traces
- ELK, Splunk, Sentry

Monitoring tools must “tell stories” about your system

How do you debug this?

```
2017/12/04 21:30:37 scanning error: bufio.Scanner: token too long
```

WHAT IS THE CONTEXT?

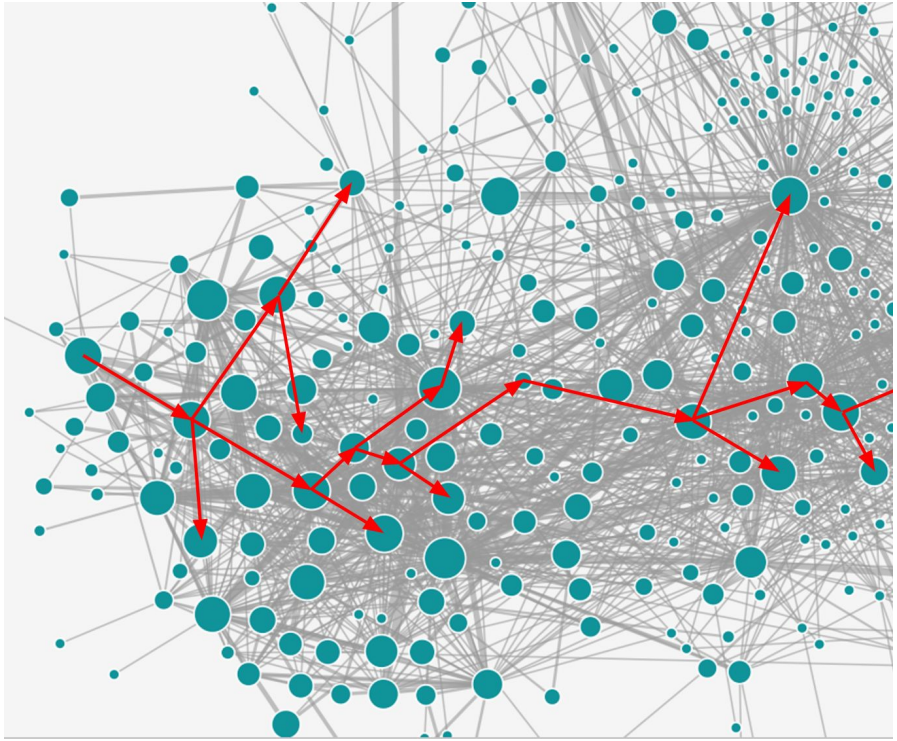
Metrics and logs don't cut it anymore!

Metrics and logs are

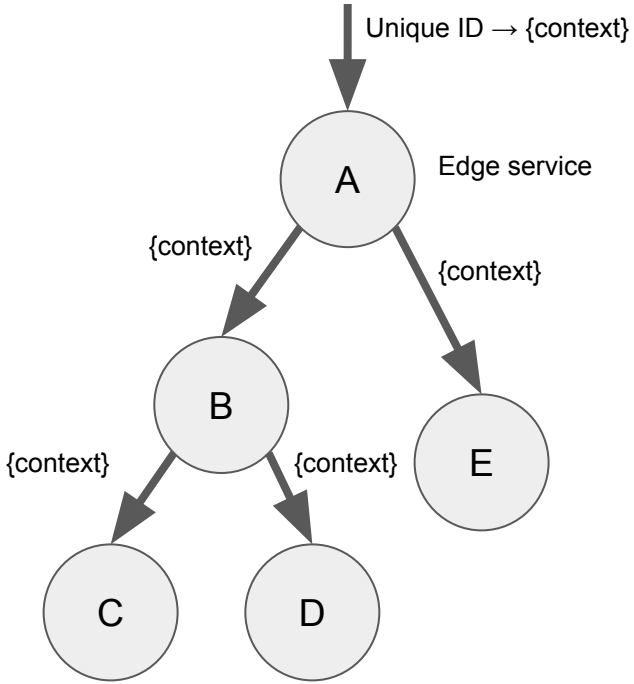
- per-instance
- missing the context

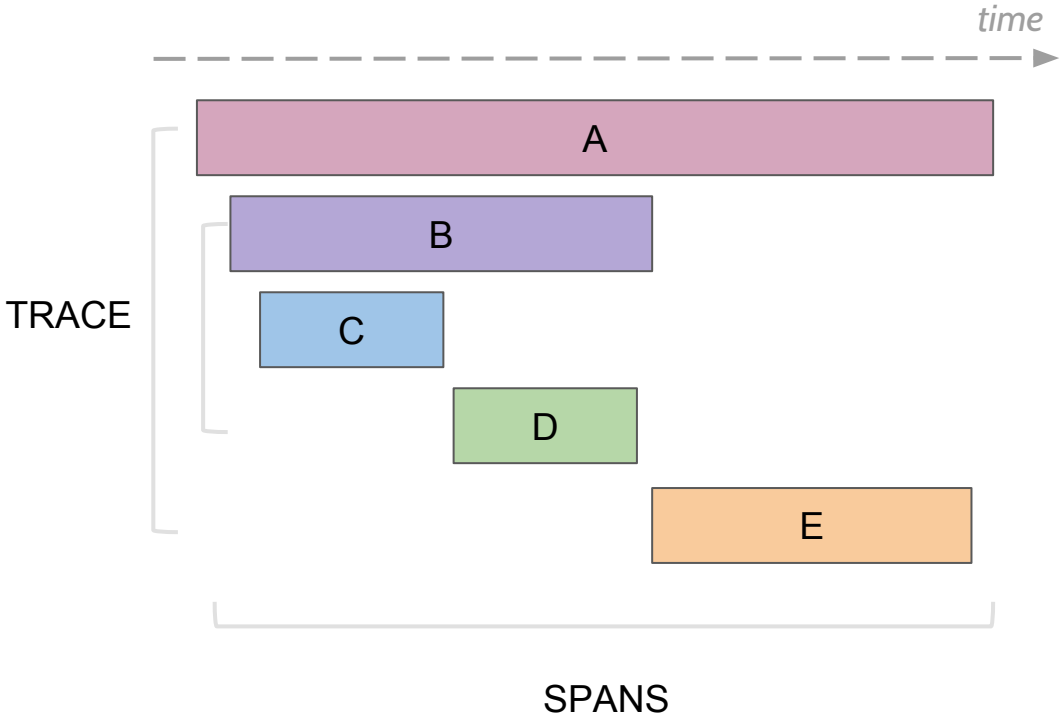
It's like debugging without a stack trace

We need to monitor **distributed transactions**



Distributed Tracing In A Nutshell





Let's look at some traces

demo time: http://bit.do/jaeger-hotrod

Distributed Tracing Systems

distributed transaction monitoring

performance and latency optimization

root cause analysis

service dependency analysis

distributed context propagation

Jaeger under the hood

Architecture, etc.

Jaeger - /ˈyāgər/, *noun*: hunter

- Inspired by Google's Dapper and OpenZipkin
- Started at Uber in August 2015
- Open sourced in April 2017
- Official CNCF project since Sep 2017
- Built-in OpenTracing support
- http://jaegertracing.io

UBER

Community

- 10 full time engineers at Uber and Red Hat
- 80+ contributors on GitHub
- Already used by many organizations
  - including Uber, Symantec, Red Hat, Base CRM, Massachusetts Open Cloud, Nets, FarmersEdge, GrafanaLabs, Northwestern Mutual, Zenly

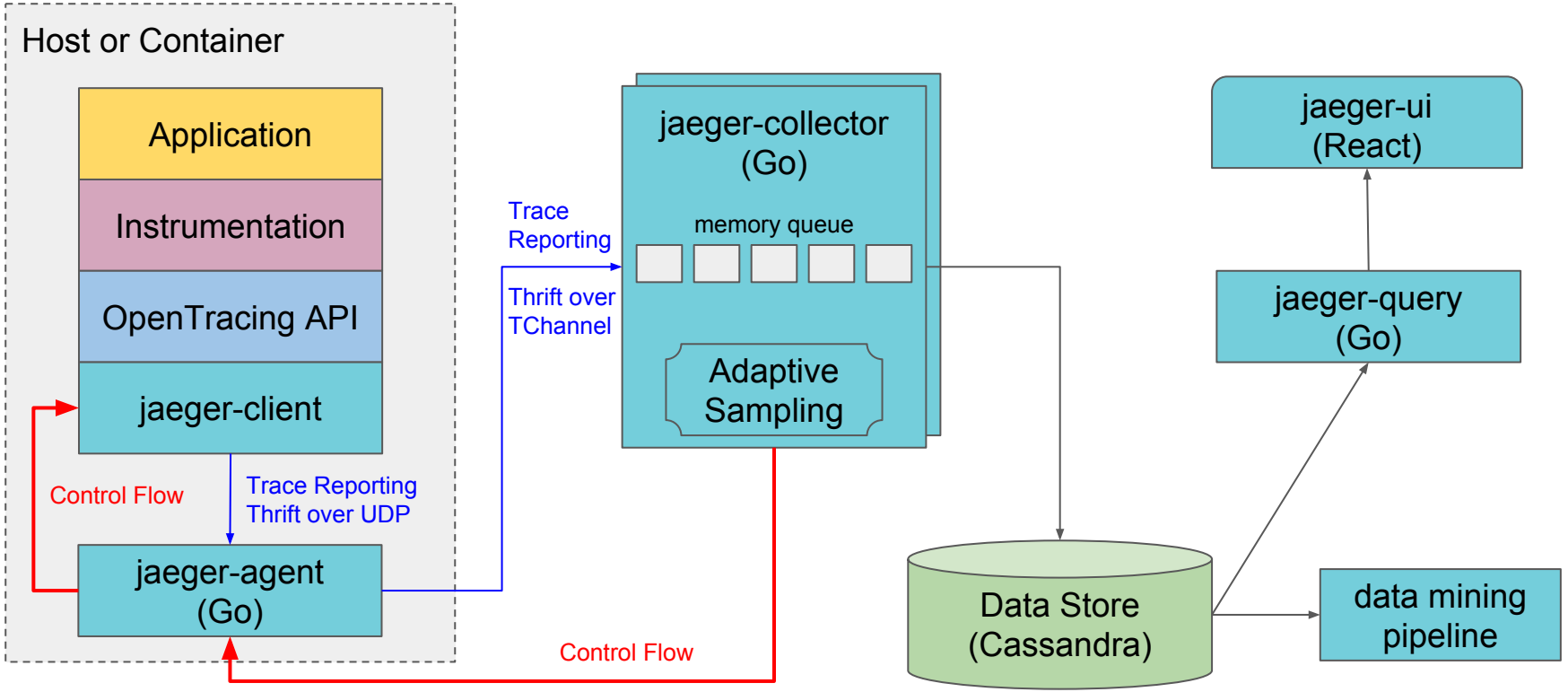
Technology Stack

- Backend components in Go
- Pluggable storage
  - Cassandra, Elasticsearch, memory, ...
- Web UI in React/Javascript
- OpenTracing instrumentation libraries

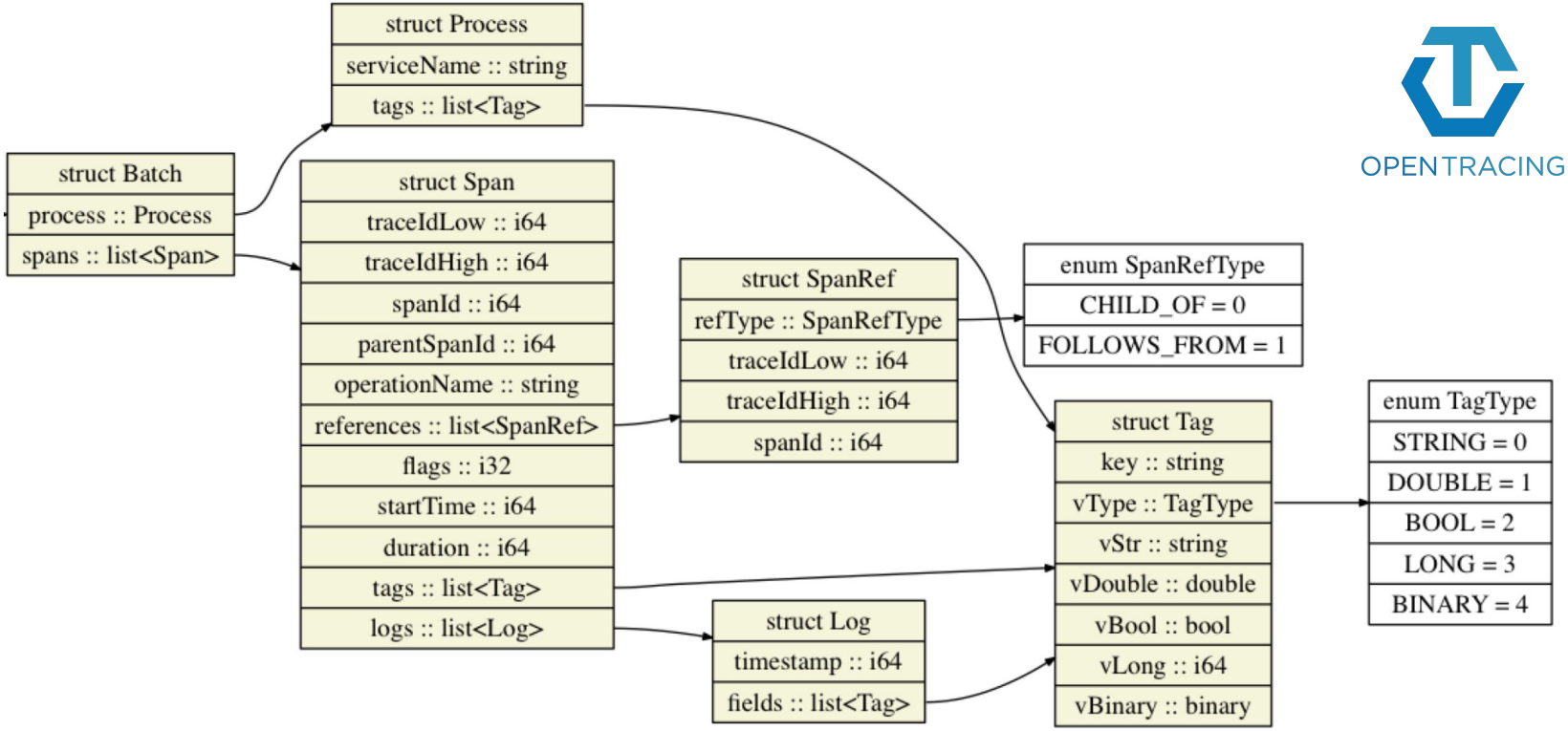
Go

python™

Architecture



Data model


struct Process
serviceName :: string
tags :: list<Tag>
struct Batch
process :: Process
spans :: list<Span>
struct Span
traceIdLow :: i64
traceIdHigh :: i64
spanId :: i64
parentSpanId :: i64
operationName :: string
references :: list<SpanRef>
flags :: i32
startTime :: i64
duration :: i64
tags :: list<Tag>
logs :: list<Log>
struct SpanRef
refType :: SpanRefType
traceIdLow :: i64
traceIdHigh :: i64
spanId :: i64
enum SpanRefType
CHILD_OF = 0
FOLLOWS_FROM = 1
struct Log
timestamp :: i64
fields :: list<Tag>
struct Tag
key :: string
vType :: TagType
vStr :: string
vDouble :: double
vBool :: bool
vLong :: i64
vBinary :: binary
enum TagType
STRING = 0
DOUBLE = 1
BOOL = 2
LONG = 3
BINARY = 4

Understanding Sampling

Tracing data can exceed business traffic.

Most tracing systems sample transactions:

- **Head-based sampling**: the sampling decision is made just before the trace is started, and it is respected by all nodes in the graph
- **Tail-based sampling**: the sampling decision is made after the trace is completed / collected

Jaeger 1.0

Released 06-Dec-2017

Jaeger 1.0 Highlights

Announcement: http://bit.do/jaeger-v1

- Multiple storage backends
- Various UI improvements
- Prometheus metrics by default
- Templates for Kubernetes deployment
  - Also a Helm chart
- Instrumentation libraries
- Backwards compatibility with Zipkin

Multiple storage backends

Official

- Cassandra 3.4+
- Elasticsearch 5.x, 6.x
- Memory storage

Experimental (by community)

- InfluxDB, ScyllaDB, AWS DynamoDB, …
- https://github.com/jaegertracing/jaeger/issues/638

Jaeger UI

- Improved performance in all screens
- Viewing large traces (e.g. 80,000 spans)
- Keyboard navigation
- Minimap navigation, zooming in & out
- Top menu customization

Zipkin drop-in replacement

Collector can accept Zipkin spans:

- JSON v1/v2 and Thrift over HTTP
- Kafka transport not supported yet

Clients:

- B3 propagation
- Jaeger clients in Zipkin environment

Monitoring

- Metrics
  - `--metrics-backend`
    - `prometheus (default), expvar`
  - `--metrics-http-route`
    - `/metrics (default)`
- Scraping Endpoints
  - Query service - API port 16686
  - Collector - HTTP API port 14268
  - Agent - sampler port 5778

Roadmap

Things we are working on

Adaptive Sampling

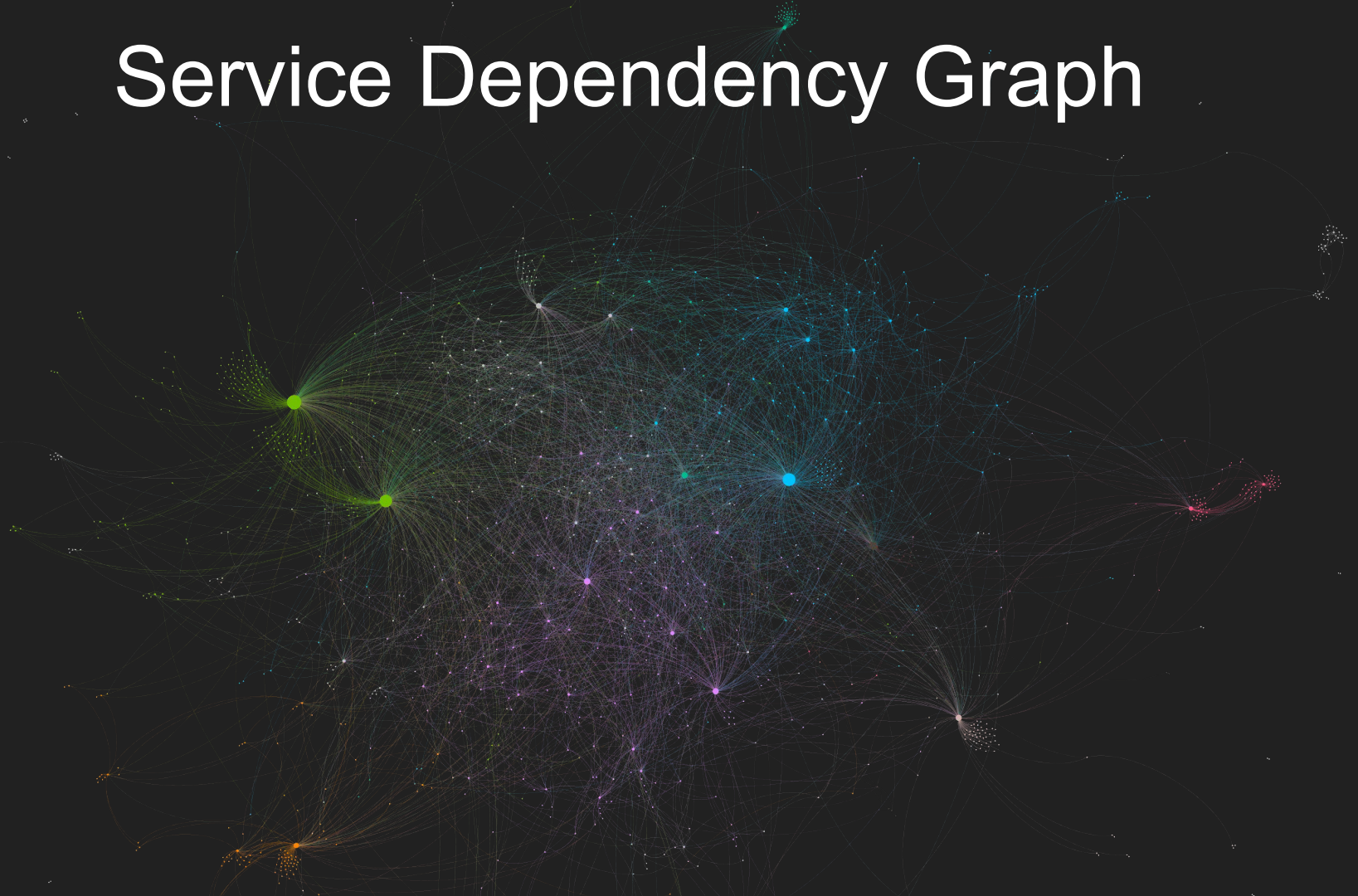
- APIs have endpoints with different QPS
- Service owners do not know the full impact of sampling probability

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

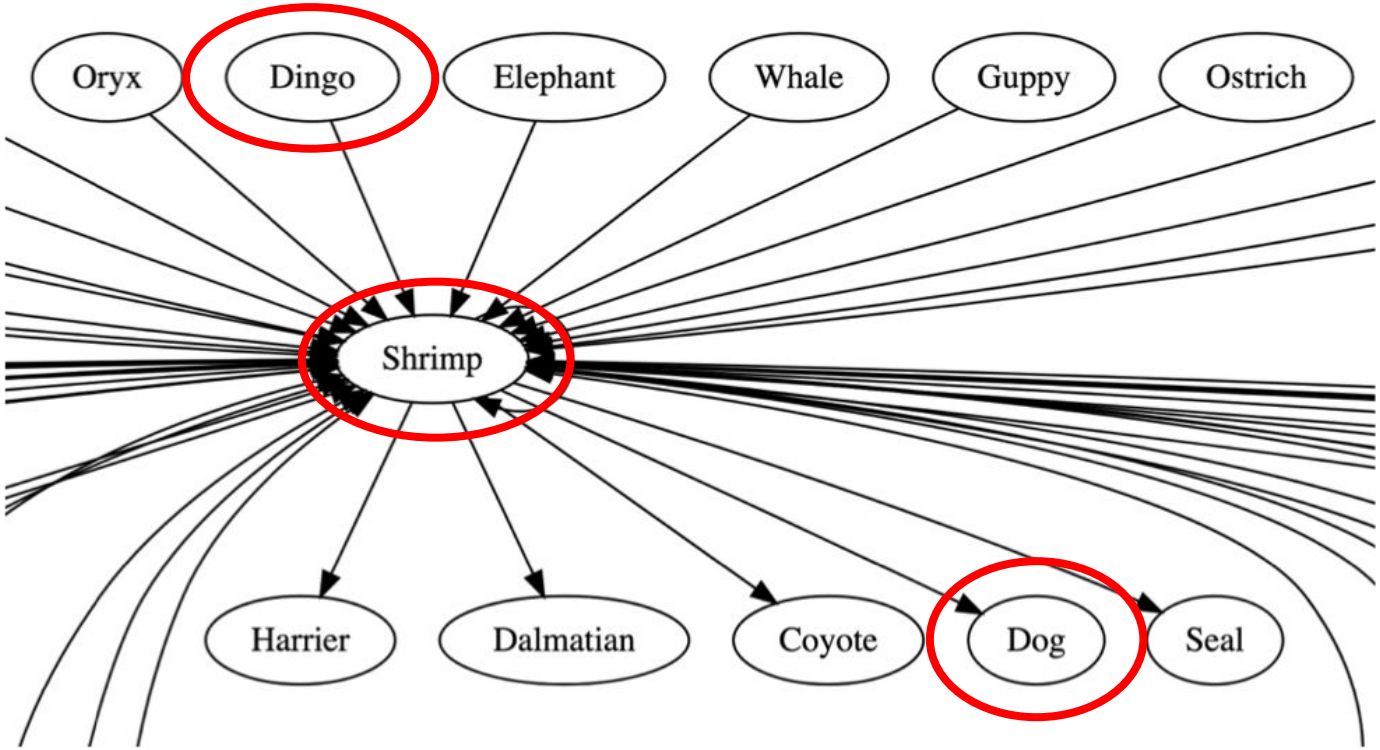
Data Pipeline

- Based on Kafka and Apache Flink
- Support aggregations and data mining
- Examples:
  - Pairwise dependency graph
  - Path-based, per endpoint dependency graph
  - Latency histograms by upstream caller

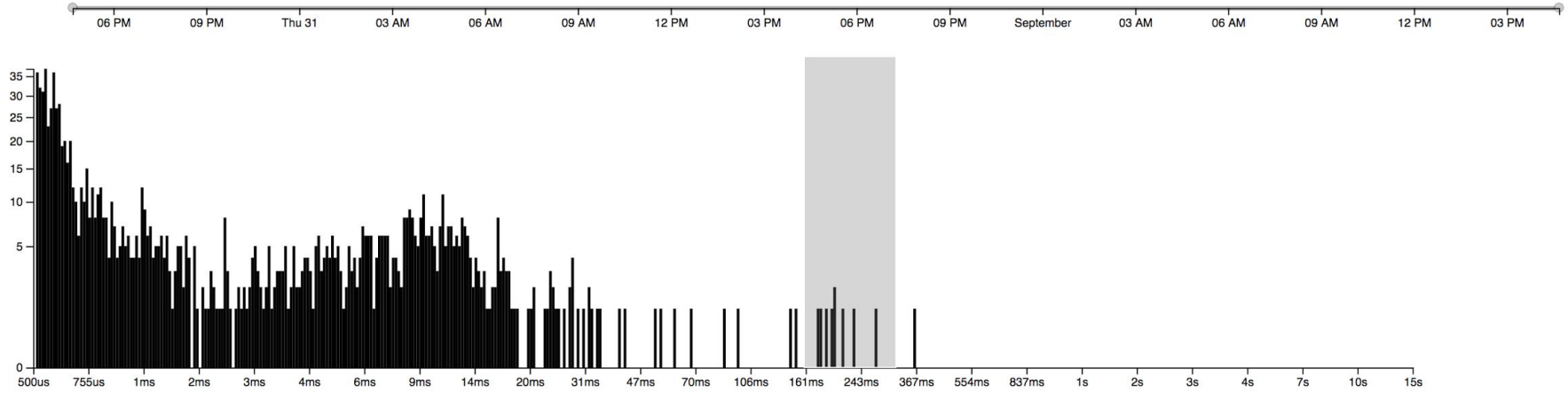
Service Dependency Graph



Does Dingo Depend on Dog?



Latency Histogram



Representative Jaeger Traces: 6 Traces Found!

Estimated Call Count by Endpoint

4	Postmaster::testEmail
4	GET
1	POST
0	Meta::health

Estimated Call Count by Upstream

5	[postmaster]Postmaster::testEmail
3	[cli-user1]
1	[cli-user2]
0	[cfgtools]
0	[tcheck]
0	[health-agent]

Project & Community

Contributors are welcome

Contributing

- **iOS client library** help wanted outreachy
  #576 opened on Dec 2, 2017 by yurishkuro
- **Flaky test TestPeerListManager_updatePeers** bug help wanted
  #574 opened on Nov 30, 2017 by yurishkuro
- **Unify metric name and format in client libraries** help wanted
  #572 opened on Nov 30, 2017 by NeoCN
- **Document external interfaces** documentation help wanted outreachy
  #456 opened on Oct 6, 2017 by jpkrohling Core Infra Best ...

Contributing

- Agree to the Certificate of Origin
- Sign all commits (`git commit -s`)
- Test coverage cannot go ↓ (backend - 100%)
- Plenty of work to go around
  - Backend
  - Client libraries
  - Kubernetes templates
  - Documentation

References

- 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 Online Community Meetings

Q & A

Open Discussion