Distributed Tracing

With Jaeger

A presentation by Amirfazel koozegar

CONTENTS

- The bigger Picture
- What Does Observability Mean?
- Observability in MicroServices
- Three pillars of Observability
  - Metrics
  - Traces
  - Logs
- Open source tools
    - Jaeger
    - Jaeger interior
- Demo

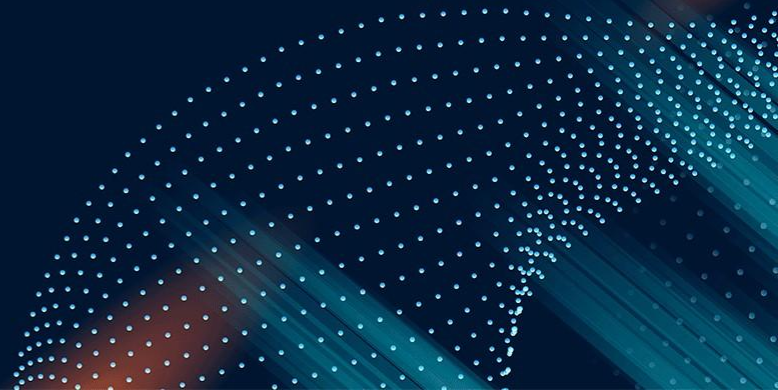
The bigger Picture

What Does Observability Mean?

What Does Observability Mean?

Observability is a measure of how well ***internal states*** of a system can be inferred from knowledge of its ***external outputs.***

Observability in MicroServices



Micro Services
What we see:

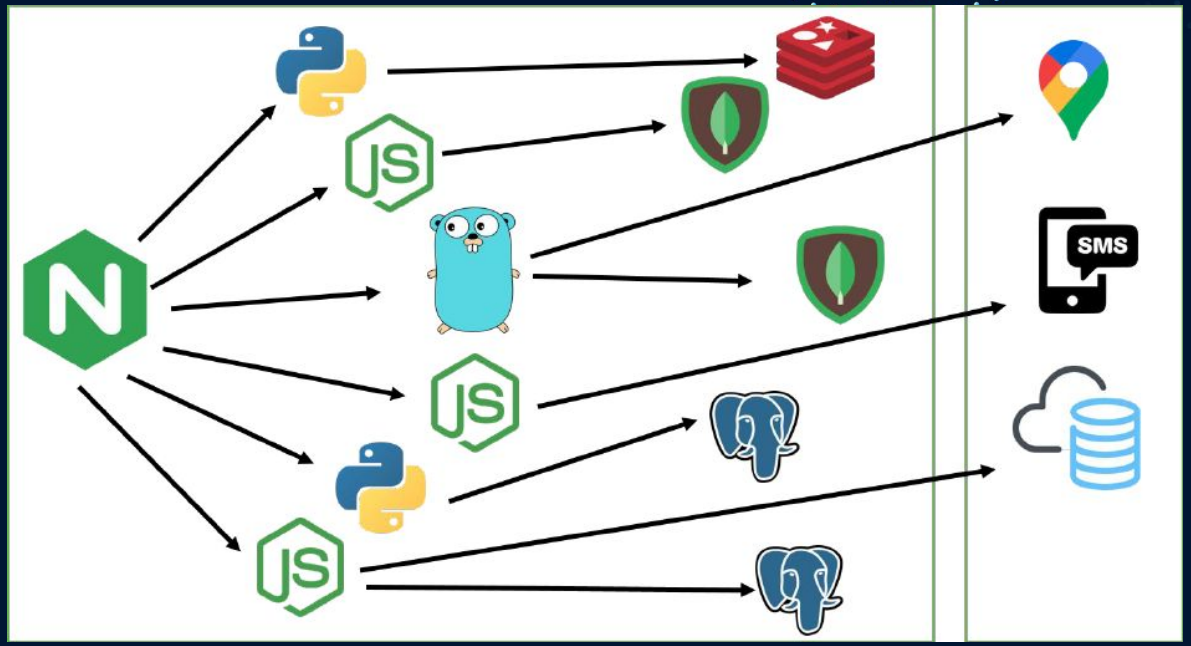
GET api/v1/basket

500 internal server error

Micro Services
What is really happening:

GET api/v1/basket

500 internal server error



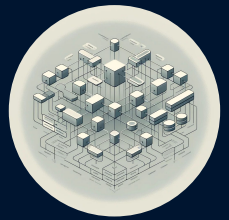
Three pillars of Observability

01

Metrics

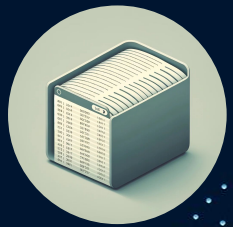
02

Traces



03

Logs

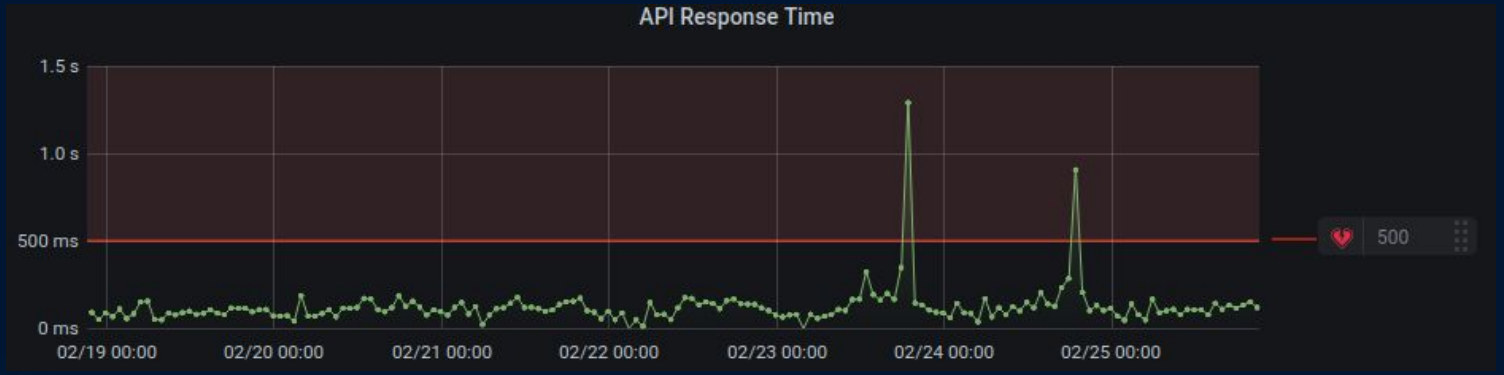


Metric

- Reflect the state of the system as a time series of numbers

- Can be used to trigger an alert when the metric exceeds specified thresholds for a predefined length of time.

Metric

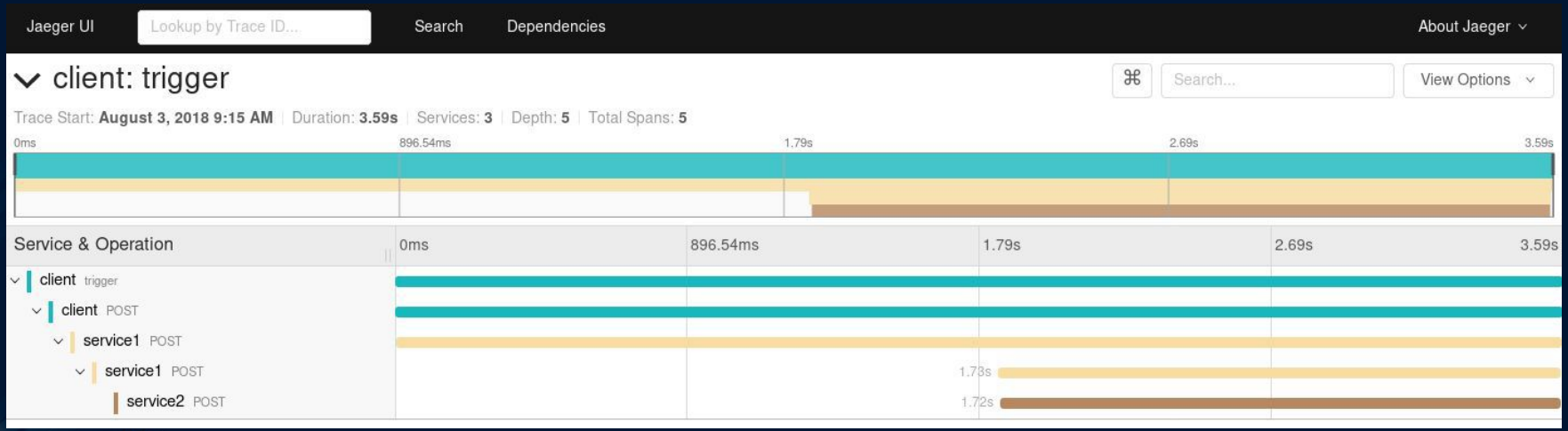

API Response Time
1.5 s
1.0 s
500 ms
0 ms
02/19 00:00
02/20 00:00
02/21 00:00
02/22 00:00
02/23 00:00
02/24 00:00
02/25 00:00
500

Traces

- Show the activity for a transaction within the entire application — all the way from the browser or mobile device down through to the database and back

- Are ***the only way*** to understand the relationships between microservices

Traces


Jaeger UI
Lookup by Trace ID...
Search
Dependencies
About Jaeger
client: trigger
Search...
View Options
Trace Start: August 3, 2018 9:15 AM | Duration: 3.59s | Services: 3 | Depth: 5 | Total Spans: 5
0ms
896.54ms
1.79s
2.69s
3.59s
Service & Operation
client trigger
client POST
service1 POST
service1 POST
service2 POST
1.73s
1.72s

Logs

- Help us examine what really happened at the system or software level

Logs

message

[redacted] mongodb 2021-02-12T00:26:01.949+0330 I COMMAND [conn26675] command [redacted] command: find { find: [redacted] }, projection: {}, returnKey: false, showRecordId: false, lsid: { id: UUID("9cec27de-af2f-4385-a6e5-5b4225cfd3fb") }, $clusterTime: { clusterTime: Timestamp(1613076960, 59), signature: { hash: BinData(0, BBF15FDC70B9233F77C2FDC878314650913995A6), keyId: 6869354269267460097 } }, $db: [redacted] } planSummary: COLLSCAN keysExamined:0 docsExamined:41561 hasSortStage:1 cursorExhausted:1 numYields:324 nreturned:0 reslen:244 locks:{ Global: { acquireCount: { r: 650 } }, Database: { acquireCount: { r: 325 } }, Collection: { acquireCount: { r: 325 } } } protocol:op_msg 115ms

Open source tools

Grafana

Real-time data visualization

Jaeger

distributed tracing system

Prometheus

monitoring system and time-series database

Jaeger

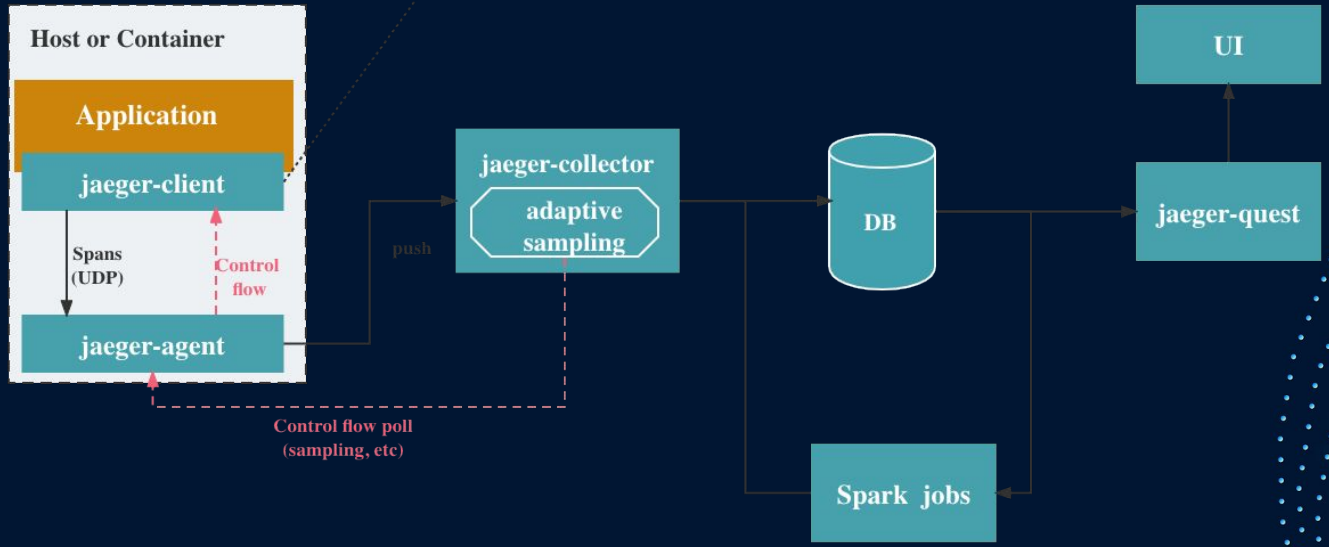
- software that you can use to monitor and troubleshoot problems on interconnected software components called microservices.

- Jaeger collects detailed information about how requests travel through a system (traces), which helps in understanding the system's behavior and identifying bottlenecks or issues.

Jaeger

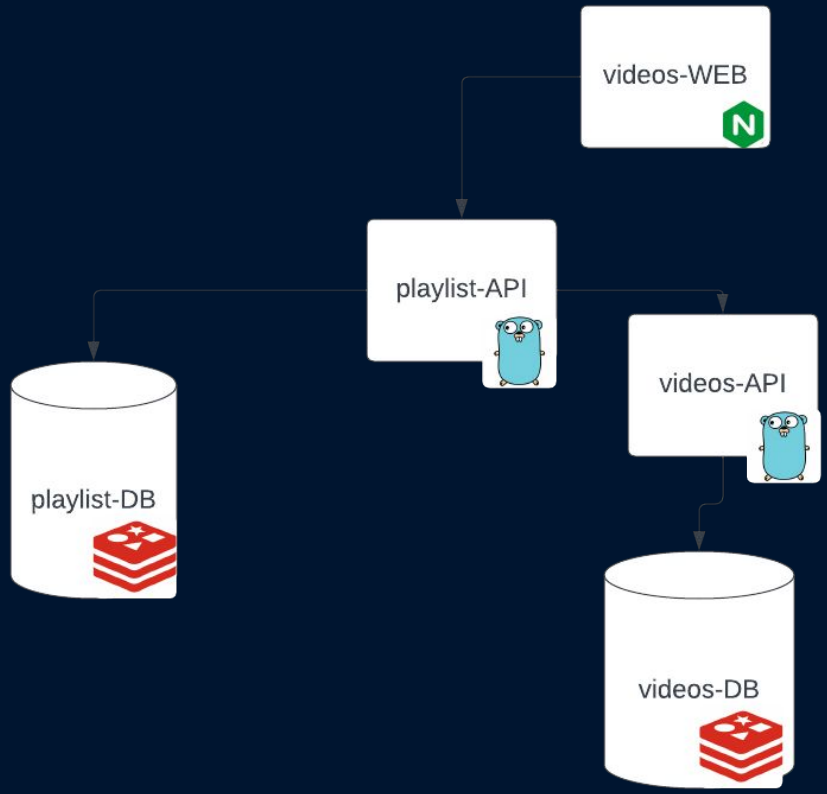
- Uses ElasticSearch and Cassandra as data stores
- Span : Unit of work
- Trace: A collection of spans
- Jaeger Agent: A network daemon that listens for spans sent over UDP. It batches these spans and forwards them to the Jaeger Collector.
- Jaeger Collector: It receives traces from Jaeger Agents and runs them through a processing pipeline. The pipeline validates, indexes, transforms, and stores trace data in a persistent storage backend.

Jaeger Interior



Lets get Hands on

Micro-service Architecture



Thank you for your attention