freshworks

How does Freshworks leverage Distributed Tracing?

Rashmi R
Senior Staff Engineer
Freshworks

Agenda

- Freshworks Distributed Tracing Architecture
- Challenges
- Adopting OpenTelemetry(OTEL)
- Cost effective, rule aware sampling strategies
- Features
- What next?
- Q & A

Freshworks at a glance

2010

Founded

FRSH

IPO September 2021

$580M+

2023 Annual Revenue Guidance

5,000+

Employees

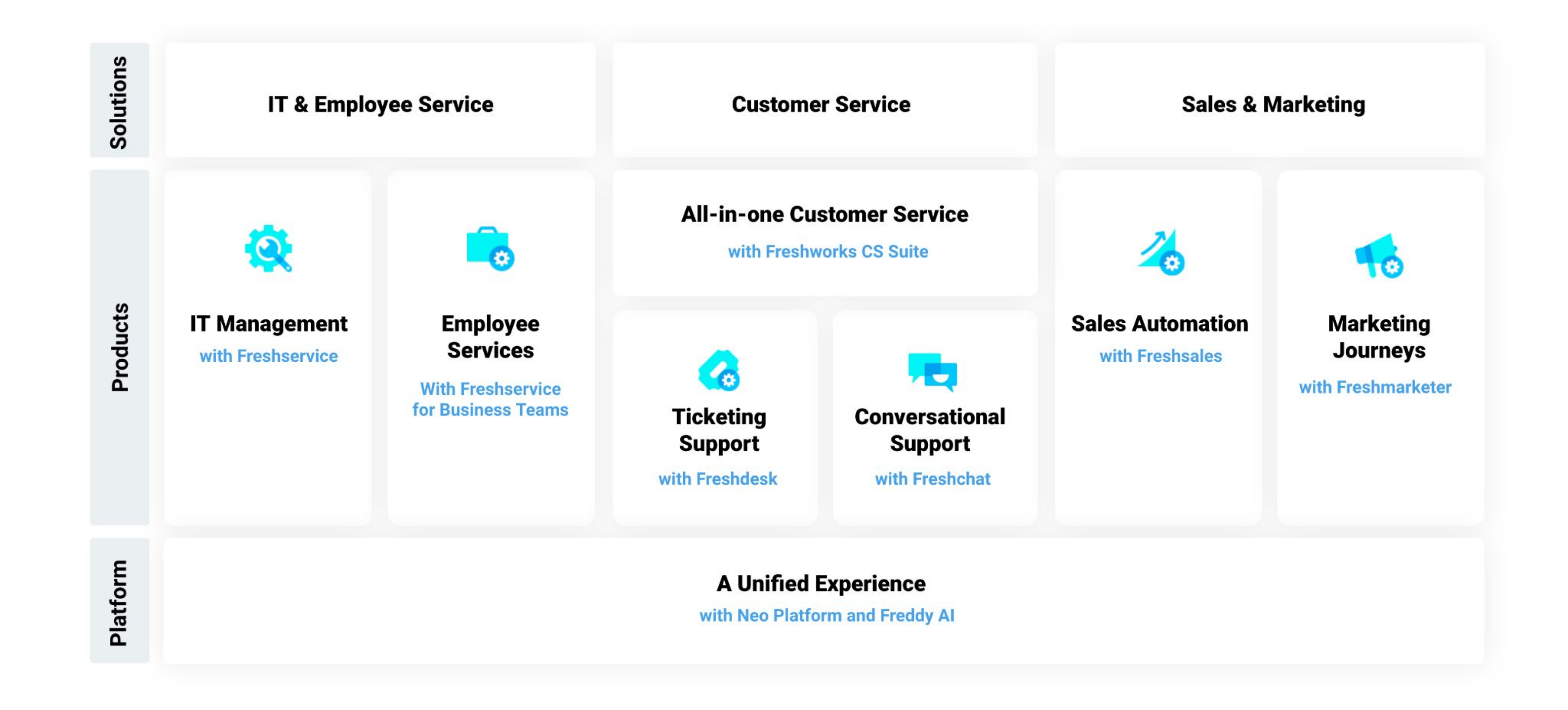
65,000+

Total Customers

Recognition

3 Gartner Magic Quadrants
Leader in 3 Major Peer Reviews

Freshworks product portfolio



Solutions

IT & Employee Service

Customer Service

Sales & Marketing

Products

IT Management
with Freshservice

Employee Services
With Freshservice for Business Teams

All-in-one Customer Service
with Freshworks CS Suite

Ticketing Support
with Freshdesk

Conversational Support
with Freshchat

Sales Automation
with Freshsales

Marketing Journeys
with Freshmarketer

Platform

A Unified Experience
with Neo Platform and Freddy AI

freshworks

Current Scale

~4Mil.
events /sec

Logs

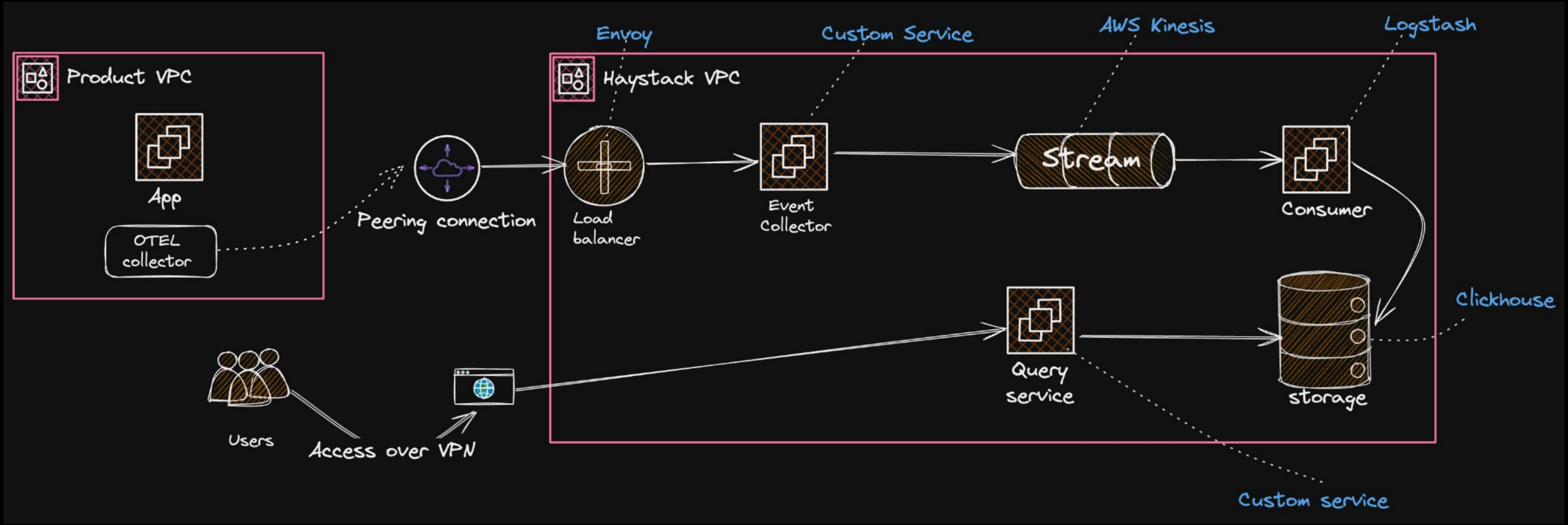
~850K
events /sec

Traces

~9 Mil.
samples/sec

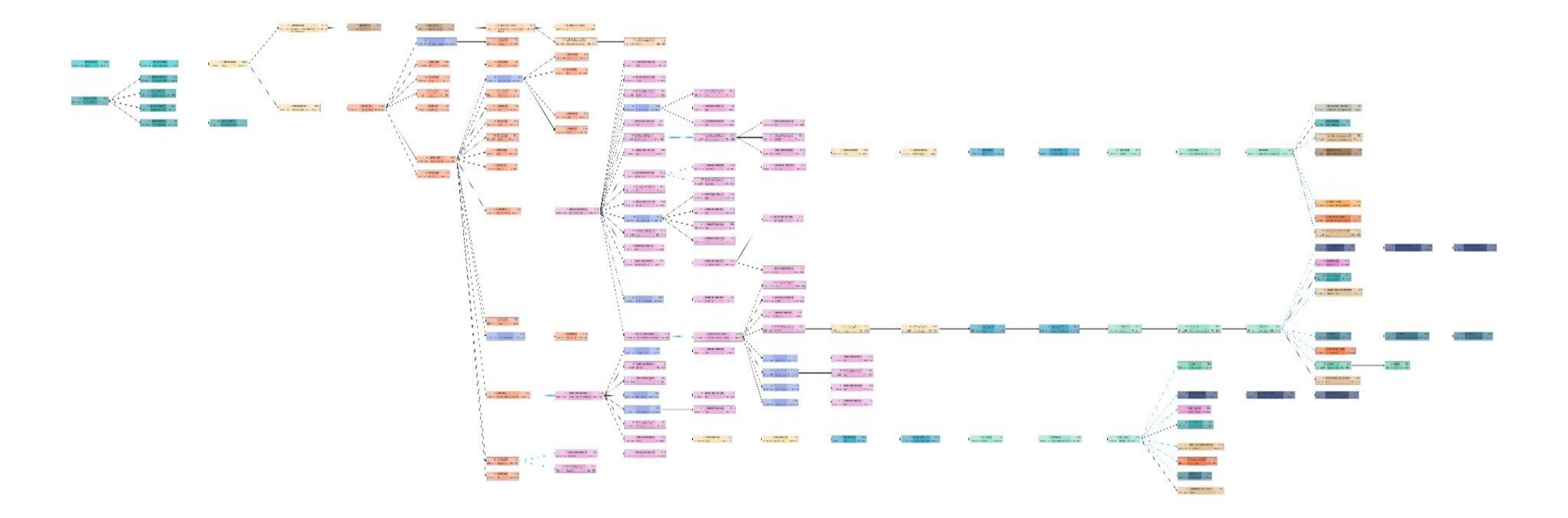
Metrics

Freshworks Distributed Tracing Architecture



freshworks

Sample Trace



- Ex., Ticket create request in Freshdesk pass through 10+ services including the bg jobs.
- Ingress for all product requests will be through edge proxy(envoy)

freshworks

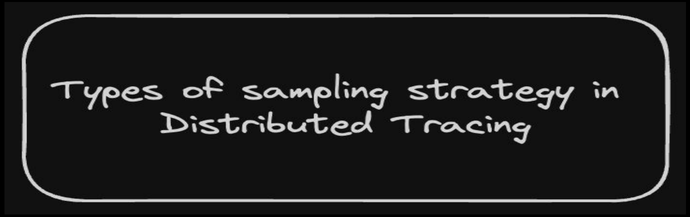
Challenges

- Multiple services
- Multi tenancy at different layers
- Multiple languages & frameworks
- High write Throughput

Adopting OpenTelemetry

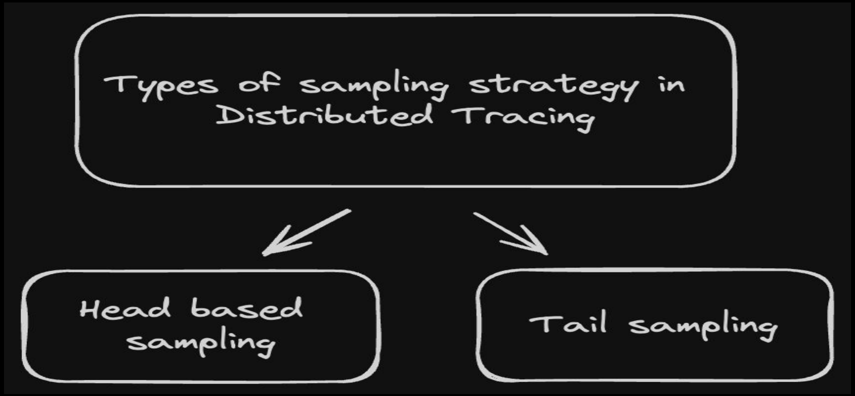
- Apache Licensed
- De-facto standard
- Language agnostic
- Backward compatibility
- Support for APM centric dashboards
- Active community
- 10+ OSS contributions from Freshworks

Sampling Strategies

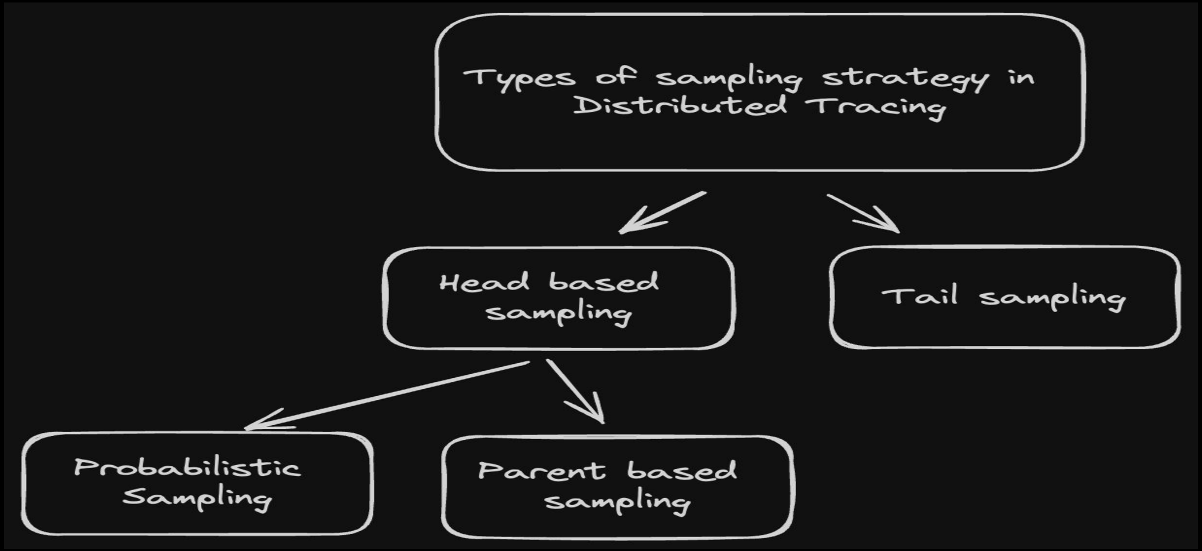


Types of sampling strategy in Distributed Tracing

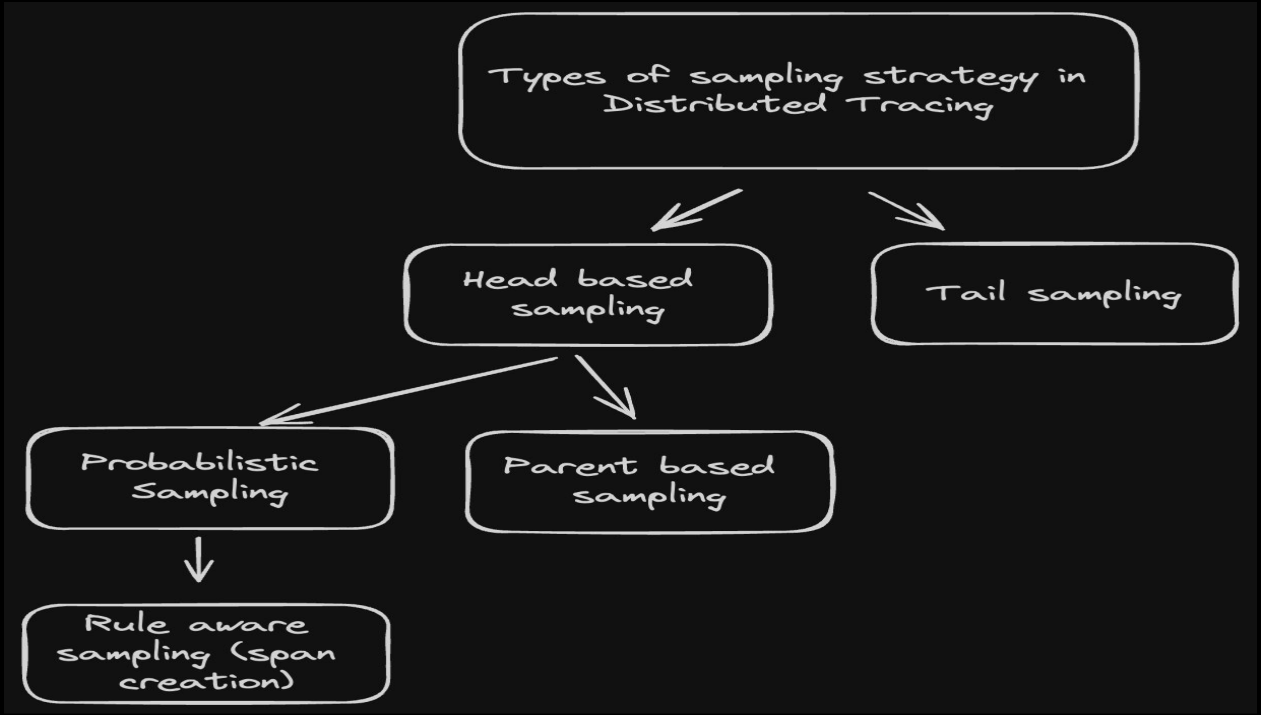
Sampling Strategies



Sampling Strategies



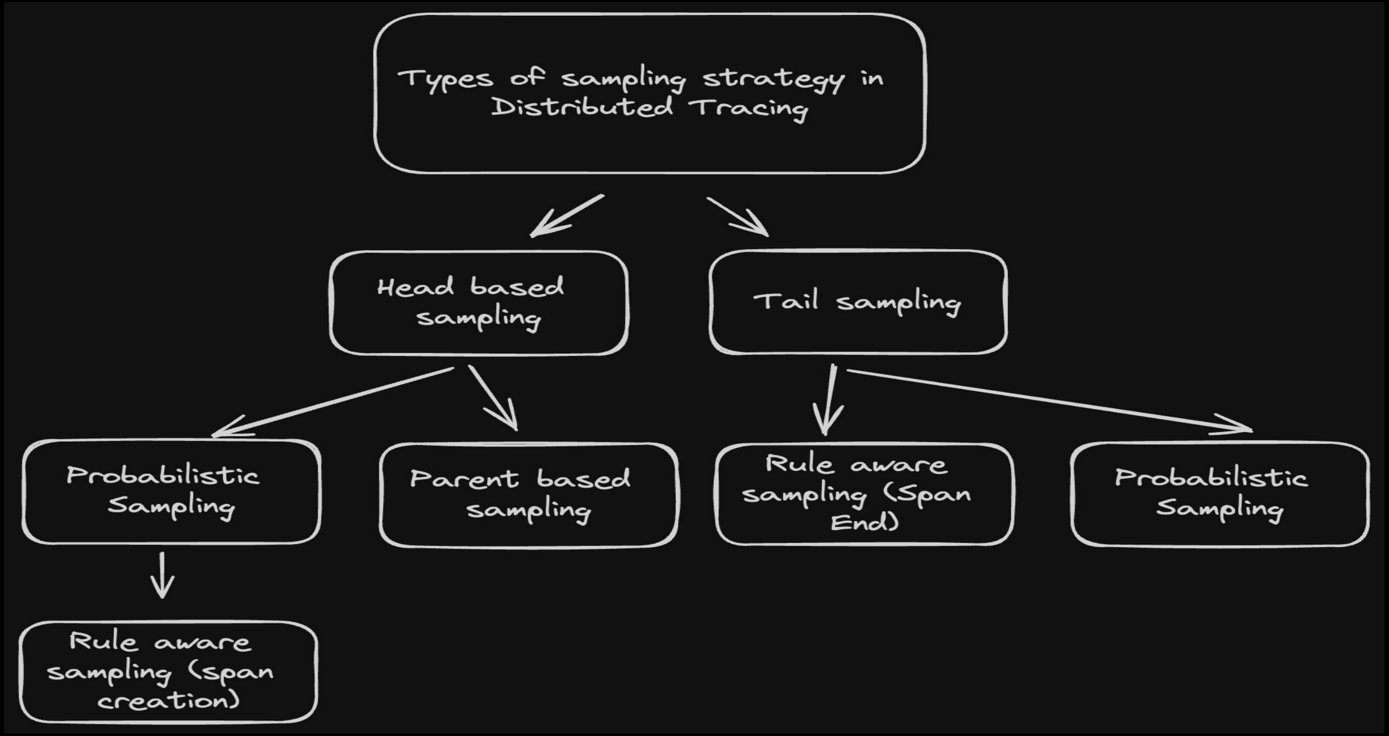
Sampling Strategies



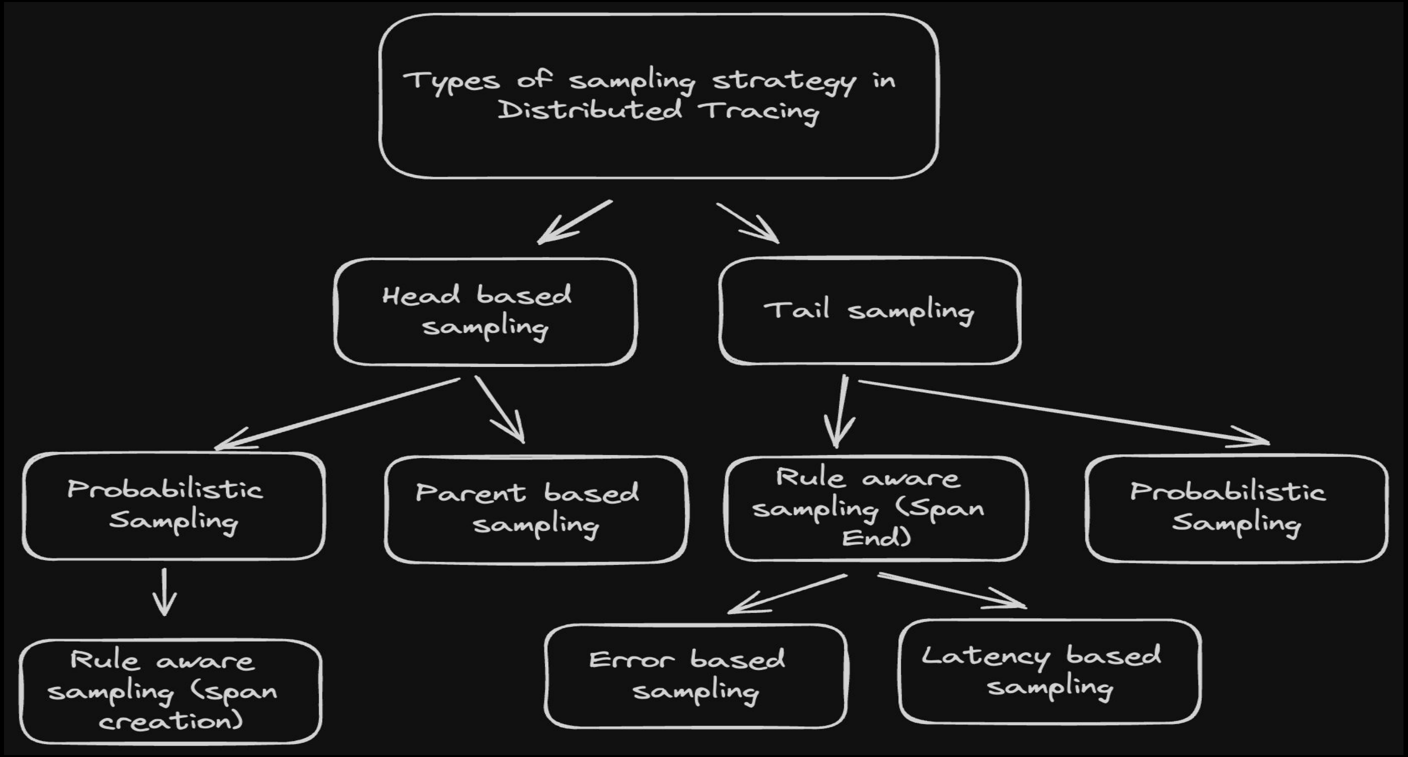
Types of sampling strategy in Distributed Tracing
Head based sampling
Tail sampling
Probabilistic Sampling
Parent based sampling
Rule aware sampling (span creation)

freshworks

Sampling Strategies


Types of sampling strategy in Distributed Tracing
Head based sampling
Tail sampling
Probabilistic Sampling
Parent based sampling
Rule aware sampling (Span End)
Probabilistic Sampling
Rule aware sampling (span creation)

Sampling Strategies



Our Sampling Strategy

- Head based sampling to reduce n/w cost & performance concern at instrumented apps.
- Hybrid sampling approach
  - Probabilistic sampling at ingress edge proxy (Envoy)
  - Parent based sampling for upstream services

Our Sampling Strategy

- Rule based sampling at Envoy
  - Helps to highlight the critical low traffic requests
  - Helps to mitigate the noisy neighbour
- Custom rule based sampling in Java and Ruby SDK
  - Based on span attributes present at the time of span creation.
  - Includes ratelimiter to prevent abuse.
  - Capture all errors in Ruby and Java SDK to cater to APM requirements.

Features

1

End to End Traceability

2

Trace Analytics

3

Link to logs & metrics

4

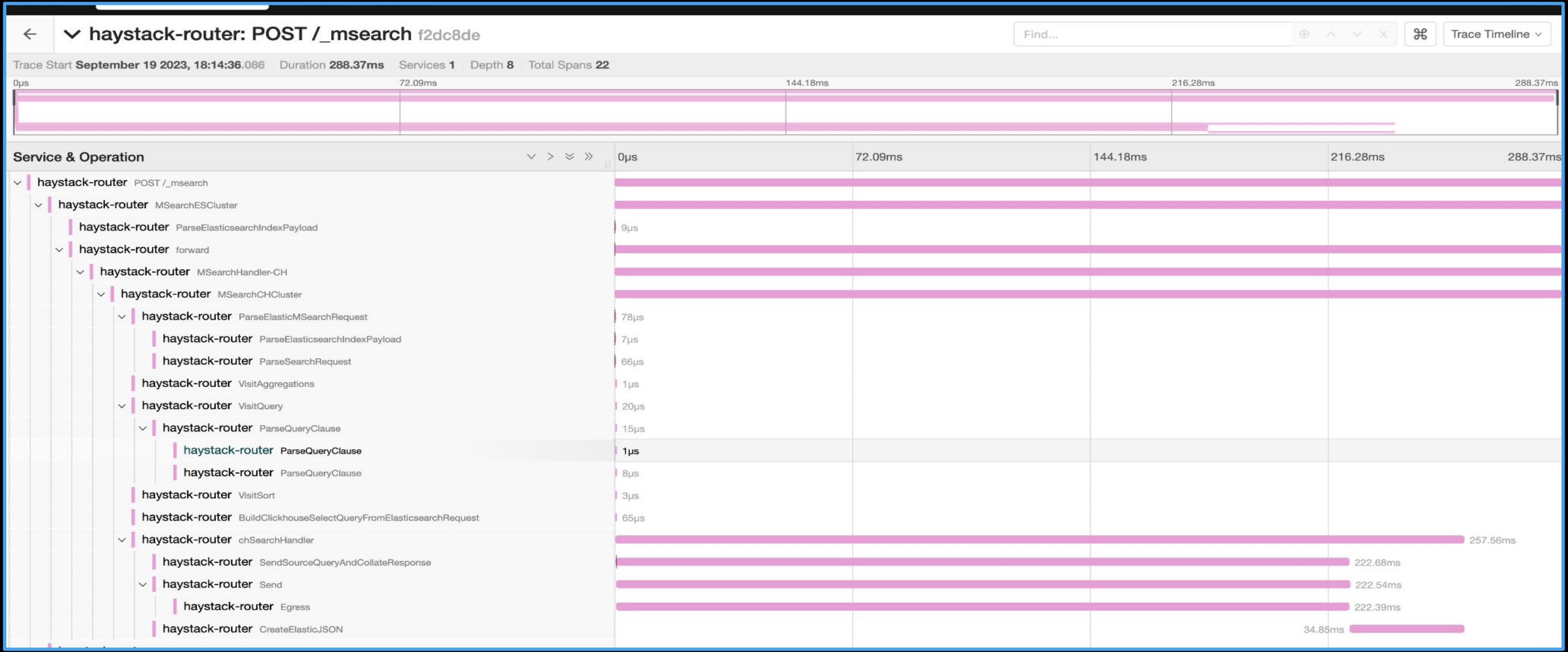
API & Alerts

5

Service Dependency Graph

freshworks

End to End Traceability



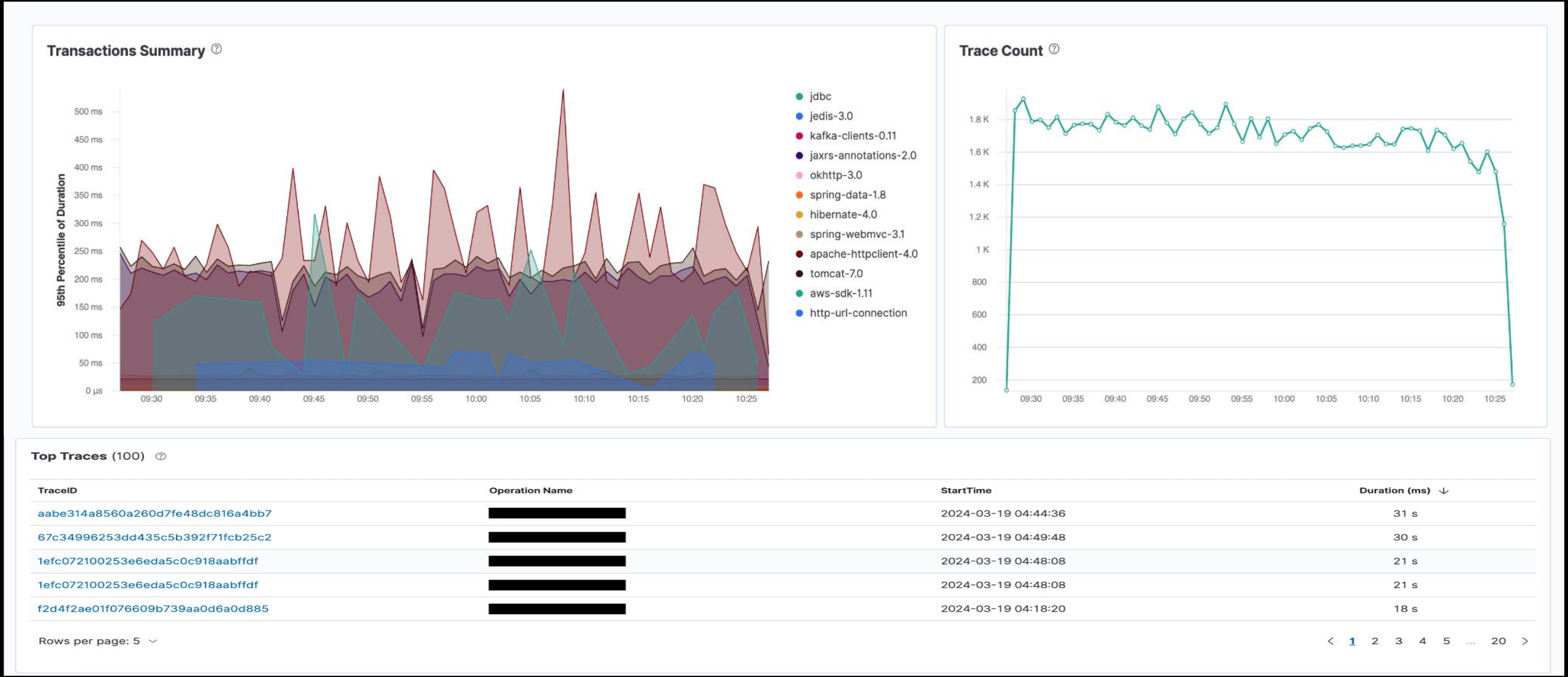
haystack-router: POST /_msearch f2dc8de

Trace Start September 19 2023, 18:14:36.086 Duration 288.37ms Services 1 Depth 8 Total Spans 22

```
haystack-router POST /_msearch
  haystack-router MSearchESCluster
    haystack-router ParseElasticsearchIndexPayload                      9μs
    haystack-router forward
      haystack-router MSearchHandler-CH
        haystack-router MSearchCHCluster
          haystack-router ParseElasticMSearchRequest                     78μs
            haystack-router ParseElasticsearchIndexPayload               7μs
            haystack-router ParseSearchRequest                           66μs
          haystack-router VisitAggregations                              1μs
          haystack-router VisitQuery                                     20μs
            haystack-router ParseQueryClause                             15μs
              haystack-router ParseQueryClause                           1μs
              haystack-router ParseQueryClause                           8μs
          haystack-router VisitSort                                      3μs
          haystack-router BuildClickhouseSelectQueryFromElasticsearchRequest 65μs
          haystack-router chSearchHandler                                257.56ms
            haystack-router SendSourceQueryAndCollateResponse            222.68ms
            haystack-router Send                                         222.54ms
              haystack-router Egress                                     222.39ms
            haystack-router CreateElasticJSON                            34.85ms
```

freshworks

Trace Analytics



Top Traces (100) ⓘ

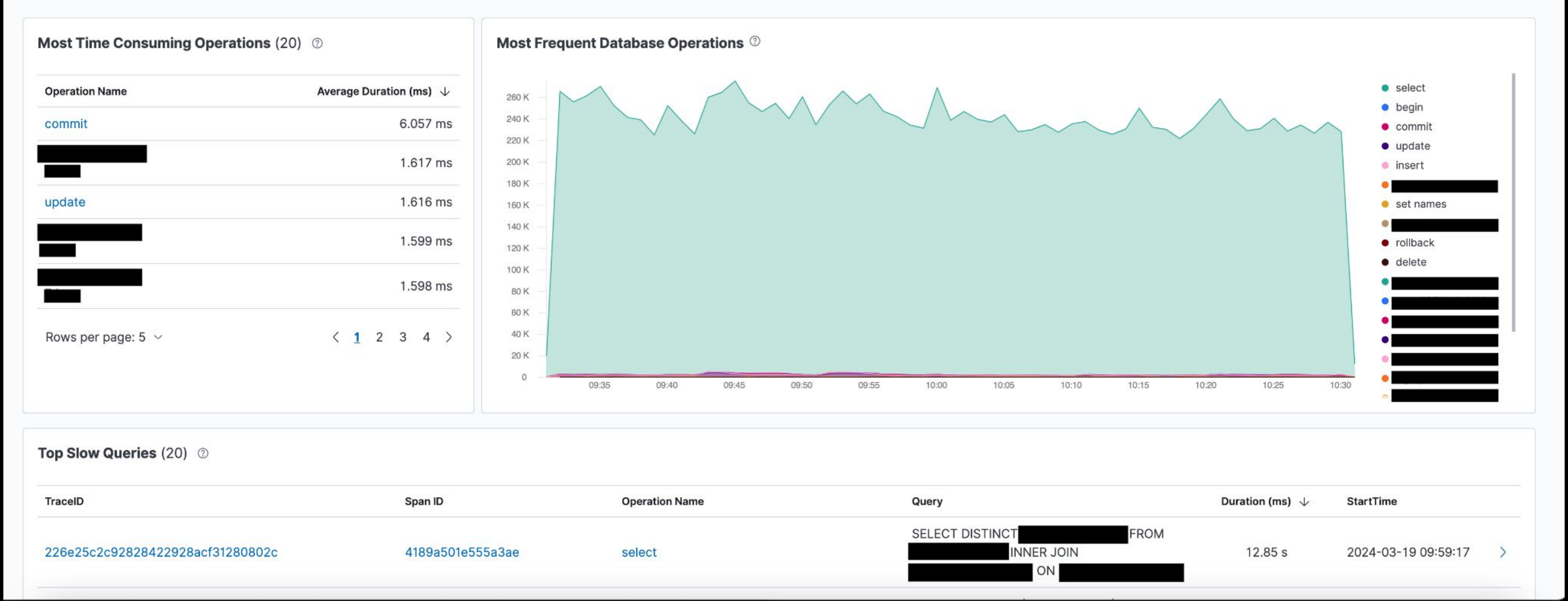
TraceID	Operation Name	StartTime	Duration (ms) ↓
aabe314a8560a260d7fe48dc816a4bb7	[redacted]	2024-03-19 04:44:36	31 s
67c34996253dd435c5b392f71fcb25c2	[redacted]	2024-03-19 04:49:48	30 s
1efc072100253e6eda5c0c918aabffdf	[redacted]	2024-03-19 04:48:08	21 s
1efc072100253e6eda5c0c918aabffdf	[redacted]	2024-03-19 04:48:08	21 s
f2d4f2ae01f076609b739aa0d6a0d885	[redacted]	2024-03-19 04:18:20	18 s

Rows per page: 5 ⌄

‹ 1 2 3 4 5 ... 20 ›

freshworks

Trace Analytics



Most Time Consuming Operations (20) ⓘ

Operation Name	Average Duration (ms) ↓
commit	6.057 ms
[redacted]	1.617 ms
update	1.616 ms
[redacted]	1.599 ms
[redacted]	1.598 ms

Rows per page: 5 ⌄ ‹ 1 2 3 4 ›

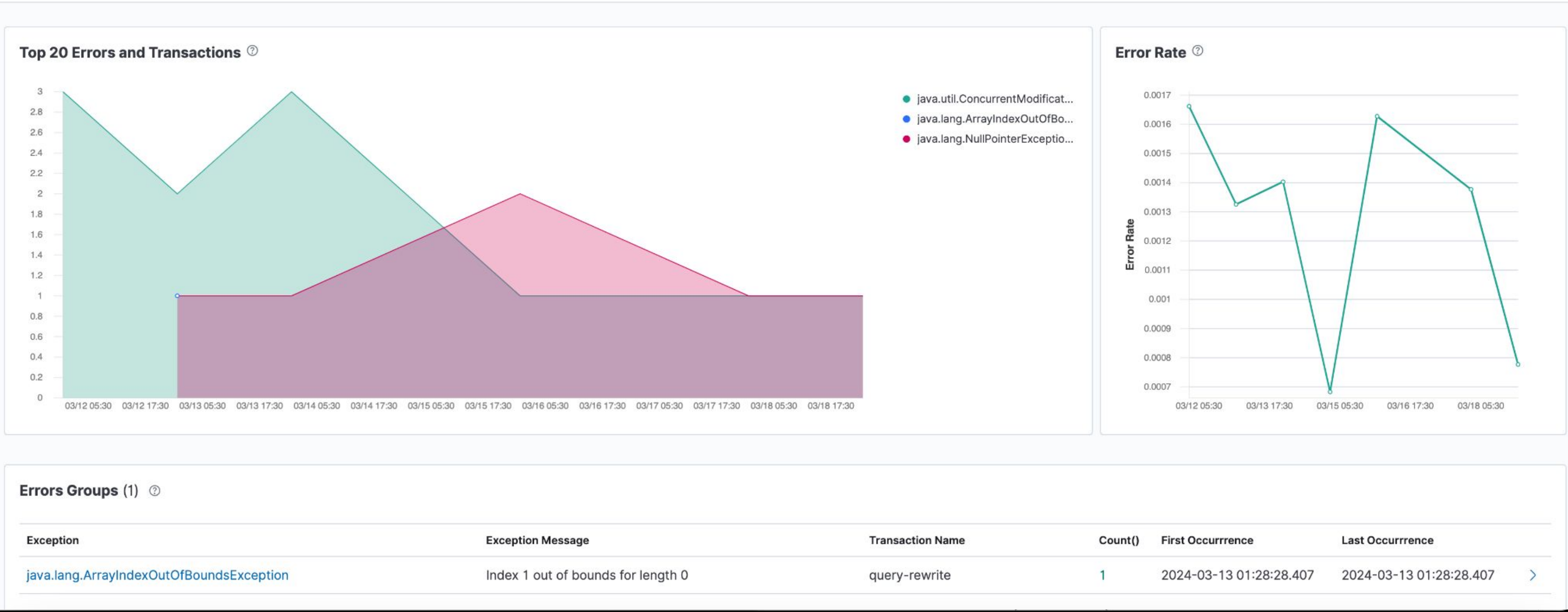
Most Frequent Database Operations ⓘ

Top Slow Queries (20) ⓘ

TraceID	Span ID	Operation Name	Query	Duration (ms) ↓	StartTime	
226e25c2c92828422928acf31280802c	4189a501e555a3ae	select	SELECT DISTINCT [redacted] FROM [redacted] INNER JOIN [redacted] ON [redacted]	12.85 s	2024-03-19 09:59:17	›

freshworks

Trace Analytics



Top 20 Errors and Transactions

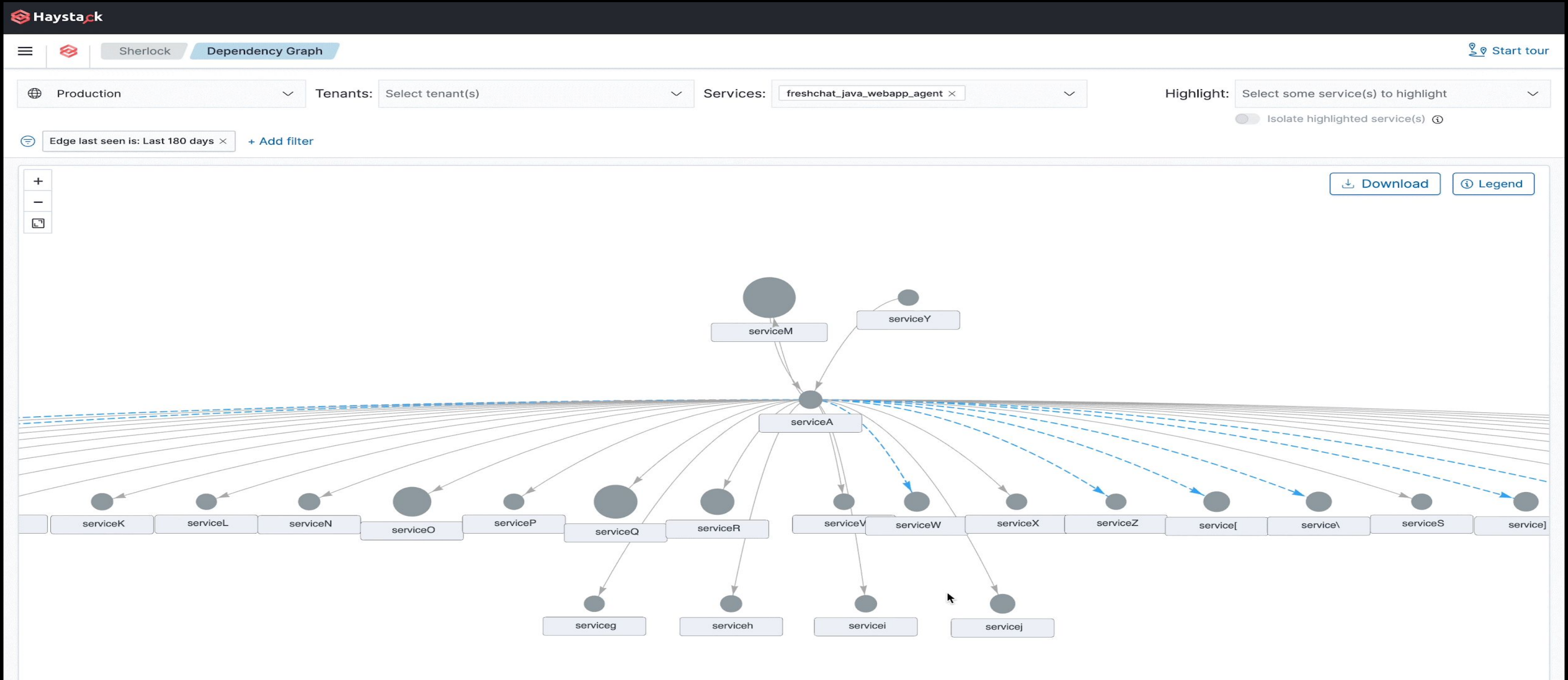
Error Rate

Errors Groups (1)

Exception	Exception Message	Transaction Name	Count()	First Occurrrence	Last Occurrrence
java.lang.ArrayIndexOutOfBoundsException	Index 1 out of bounds for length 0	query-rewrite	1	2024-03-13 01:28:28.407	2024-03-13 01:28:28.407

freshworks

Dependency Graph



Tenants: Select tenant(s)

Services: freshchat_java_webapp_agent ×

Highlight: Select some service(s) to highlight

Isolate highlighted service(s)

Edge last seen is: Last 180 days ×

freshworks

What next?

- Unified UI to view both logs and traces
- Tail sampling to increase visibility into problematic requests
- Derive metrics from Trace data to power the monitoring of Golden Signals (RED)
- Autocorrelation with other telemetry signals to reduce MTTR

Thank you!

freshworks

Q&A

freshworks