

Kubernetes Observability Scorecard

Author: Nho Luong

Skill: DevOps Engineer Lead



Nho Luong
has successfully passed all requirements for
Microsoft Certified: Azure Solutions Architect Expert

Credential ID: 3C3D14D053F5E51
Certification number: 46F9FB-703F58
Earned on: June 14, 2024
Expires on: June 15, 2025

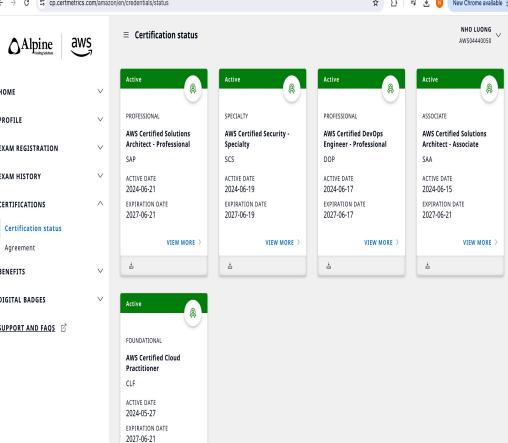
 Online Verifiable



nho luong
AWS Certified Solutions Architect - Professional

VALIDATION NUMBER: 119ee670510e4bb99fe2956152effb7e
VALIDATE AT: <https://aws.amazon.com/verification>

Issue Date: June 21, 2024
Expiration Date: June 21, 2027



HOME | EXAM REGISTRATION | CERTIFICATIONS | SUPPORT AND FAQS

PROFILE | EXAM HISTORY | BENEFITS | DIGITAL BADGES

Agreement | ACTIVE DATE: 2024-05-19 | EXPIRATION DATE: 2027-05-21 | VIEW MORE >

PROFESSIONAL | SPECIALTY | PROFESSIONAL | ASSOCIATE

ARN Certified Solutions Architect - Professional | AWS Certified Security Specialty | AWS Certified DevOps Engineer - Professional | AWS Certified Solutions Architect - Associate

SAP | SCS | DOP | SAA

ACTIVE DATE: 2024-05-19 | EXPIRATION DATE: 2027-05-19 | VIEW MORE >

ACTIVE DATE: 2024-05-17 | EXPIRATION DATE: 2027-05-17 | VIEW MORE >

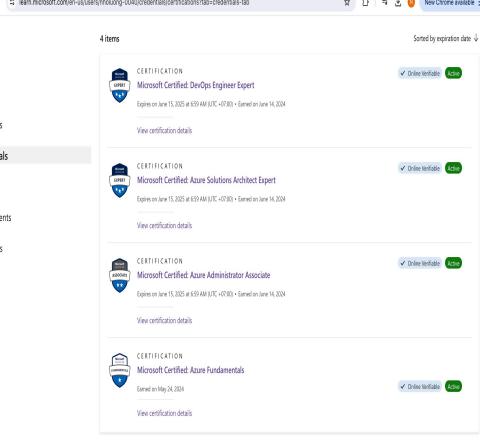
ACTIVE DATE: 2024-05-15 | EXPIRATION DATE: 2027-05-17 | VIEW MORE >

ACTIVE DATE: 2024-05-15 | EXPIRATION DATE: 2027-05-21 | VIEW MORE >

FOUNDATIONAL | ACTIVE DATE: 2024-05-27 | EXPIRATION DATE: 2027-05-21 | VIEW MORE >

ARN Certified Cloud Practitioner | CLF

ACTIVE DATE: 2024-05-27 | EXPIRATION DATE: 2027-05-21



HOME | EXAM REGISTRATION | CERTIFICATIONS | SUPPORT AND FAQS

PROFILE | EXAM HISTORY | BENEFITS | DIGITAL BADGES

Agreement | ACTIVE DATE: 2024-05-19 | EXPIRATION DATE: 2027-05-21 | VIEW MORE >

Activity | 4 items

Training | Plans | Challenges | Credentials | Q&A | Achievements | Collections | Transcript

CERTIFICATION | Microsoft Certified: DevOps Engineer Expert | Expire on June 15, 2025 at 03:59 AM (UTC+07:00) • Earned on June 14, 2024

CERTIFICATION | Microsoft Certified: Azure Solutions Architect Expert | Expire on June 15, 2025 at 03:59 AM (UTC+07:00) • Earned on June 14, 2024

CERTIFICATION | Microsoft Certified: Azure Administrator Associate | Expire on June 15, 2025 at 03:59 AM (UTC+07:00) • Earned on June 14, 2024

CERTIFICATION | Microsoft Certified: Azure Fundamentals | Expire on May 24, 2024



First, a Critique

Author: Nho Luong

Skill: DevOps Engineer Lead

The Conventional Wisdom

Observing microservices is hard

Google and Facebook solved this (right???)

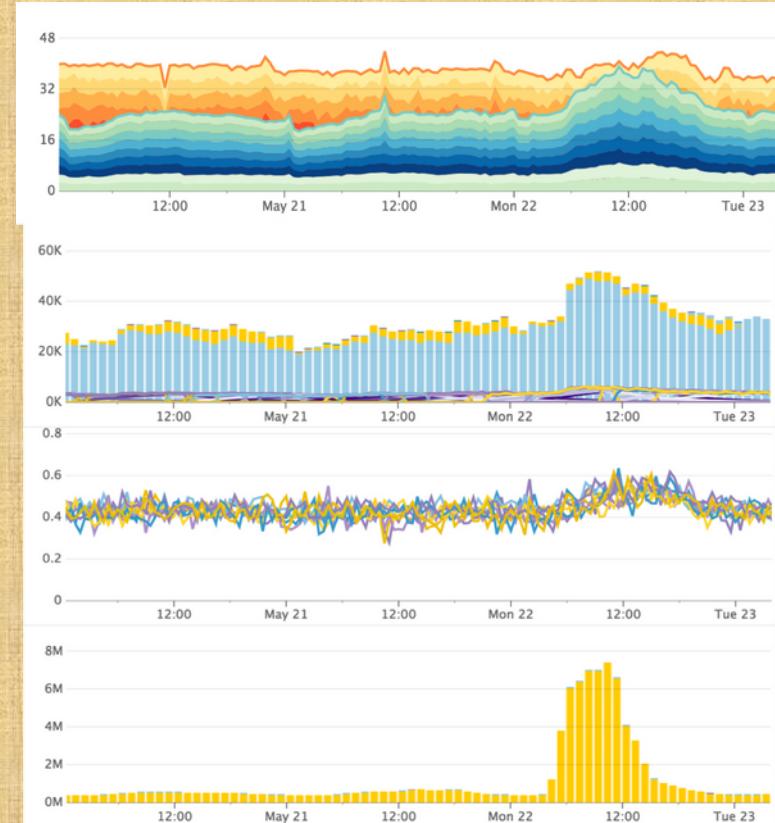
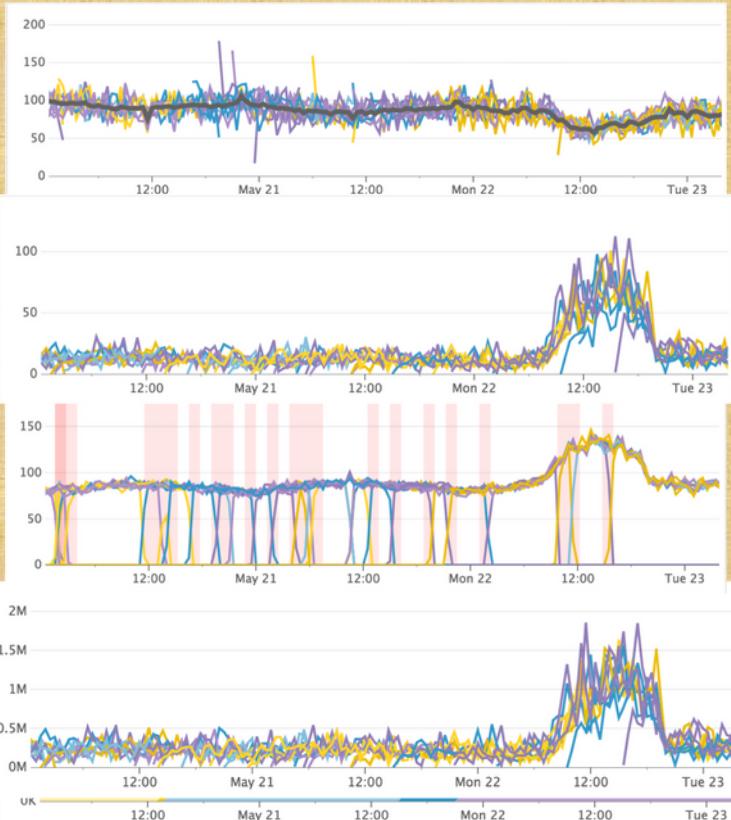
They used Metrics, Logging, and Distributed Tracing ...

So we should, too.

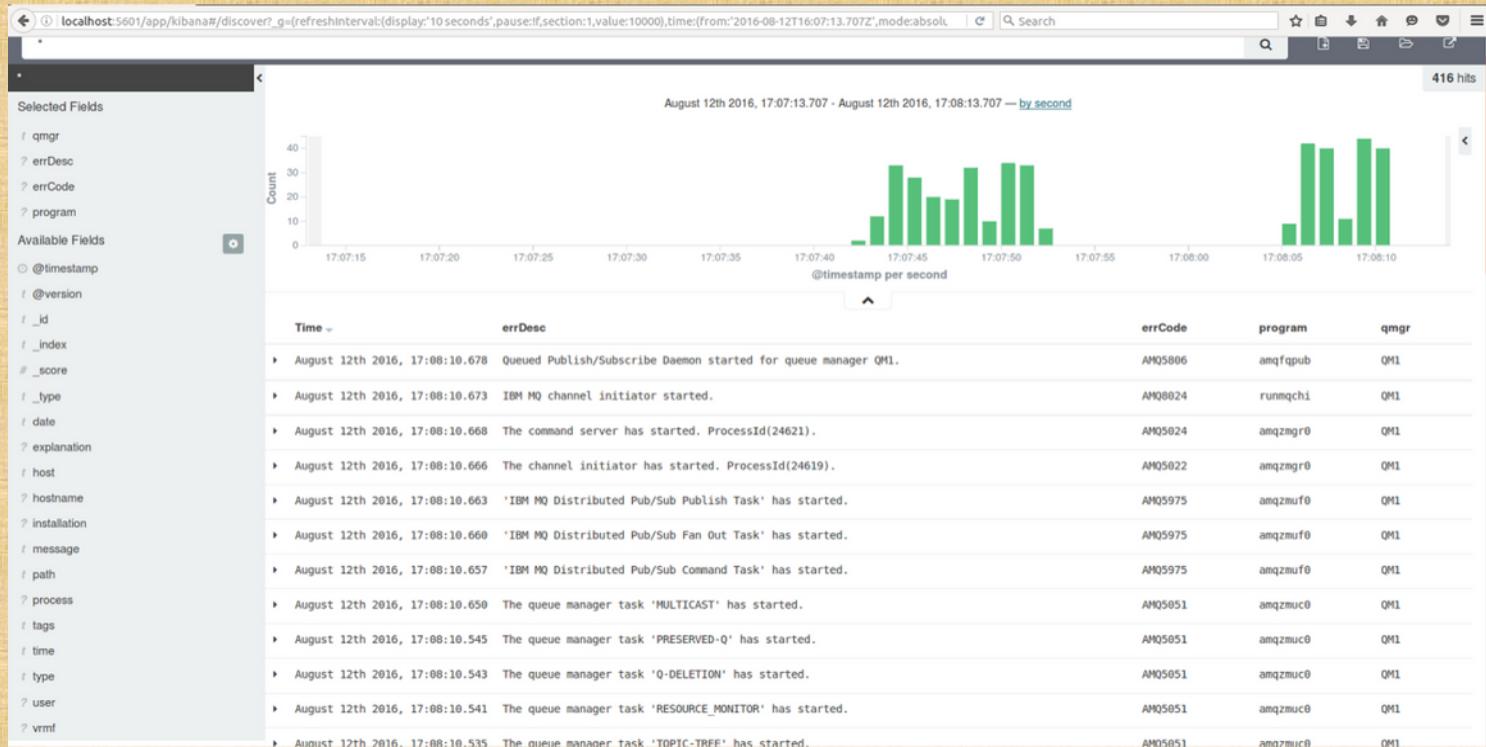
The Three Pillars of Observability

- Metrics
- Logging
- Distributed Tracing

Metrics!



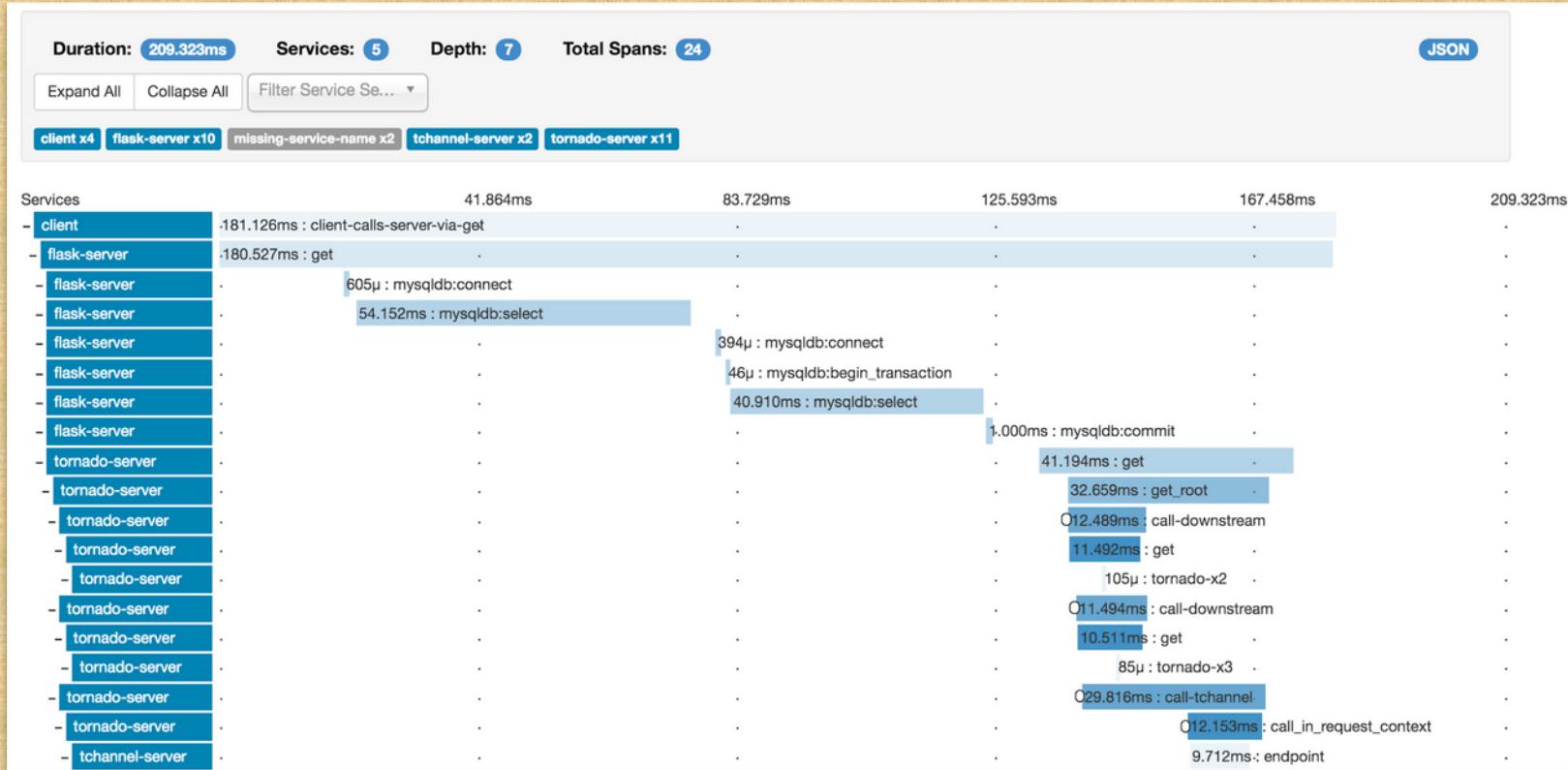
Logging!



Author: Nho Luong

Skill: DevOps Engineer Lead

Tracing!



Author: Nho Luong

Skill: DevOps Engineer Lead



Author: Nho Luong

Skill: DevOps Engineer Lead

Fatal Flaws



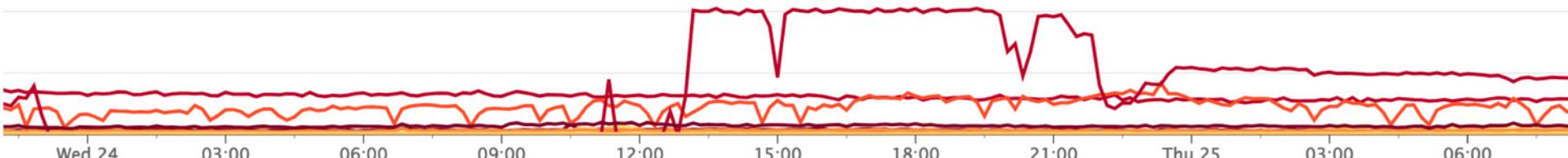
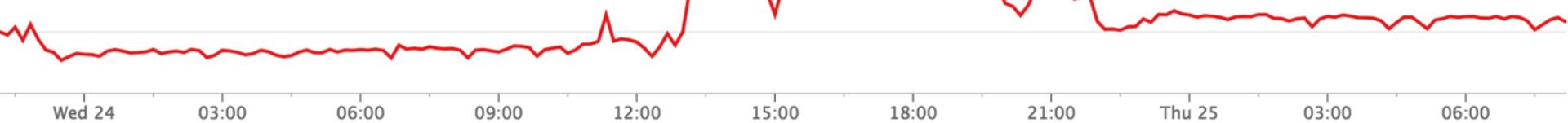
Author: Nho Luong

Skill: DevOps Engineer Lead

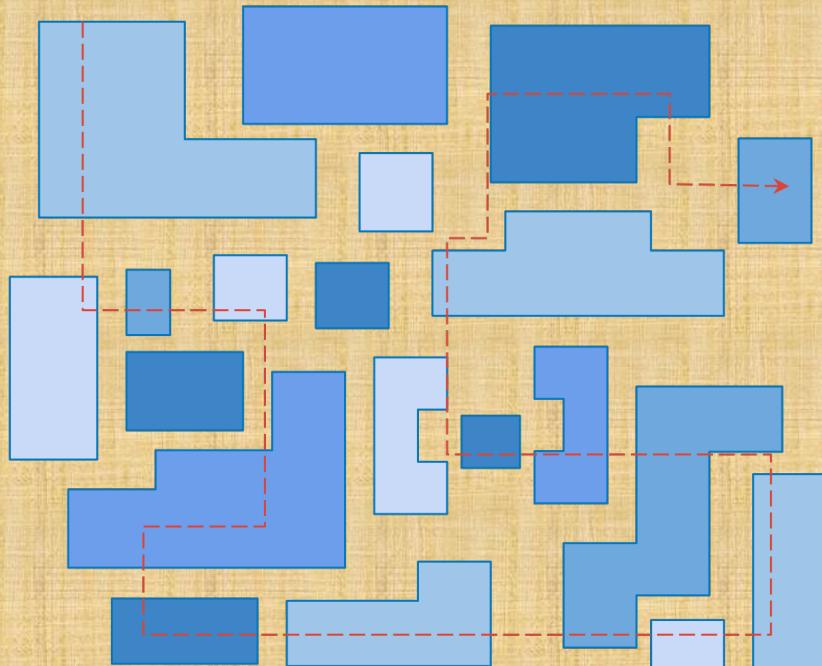
A word nobody knew in 2015...

Dimensions (aka “tags”) can explain variance in timeseries data (aka “metrics”) ...

... but cardinality



Logging Data Volume: a reality check



transaction rate all
x microservices cost of
x net+storage weeks of
retention -----

way too much \$\$\$

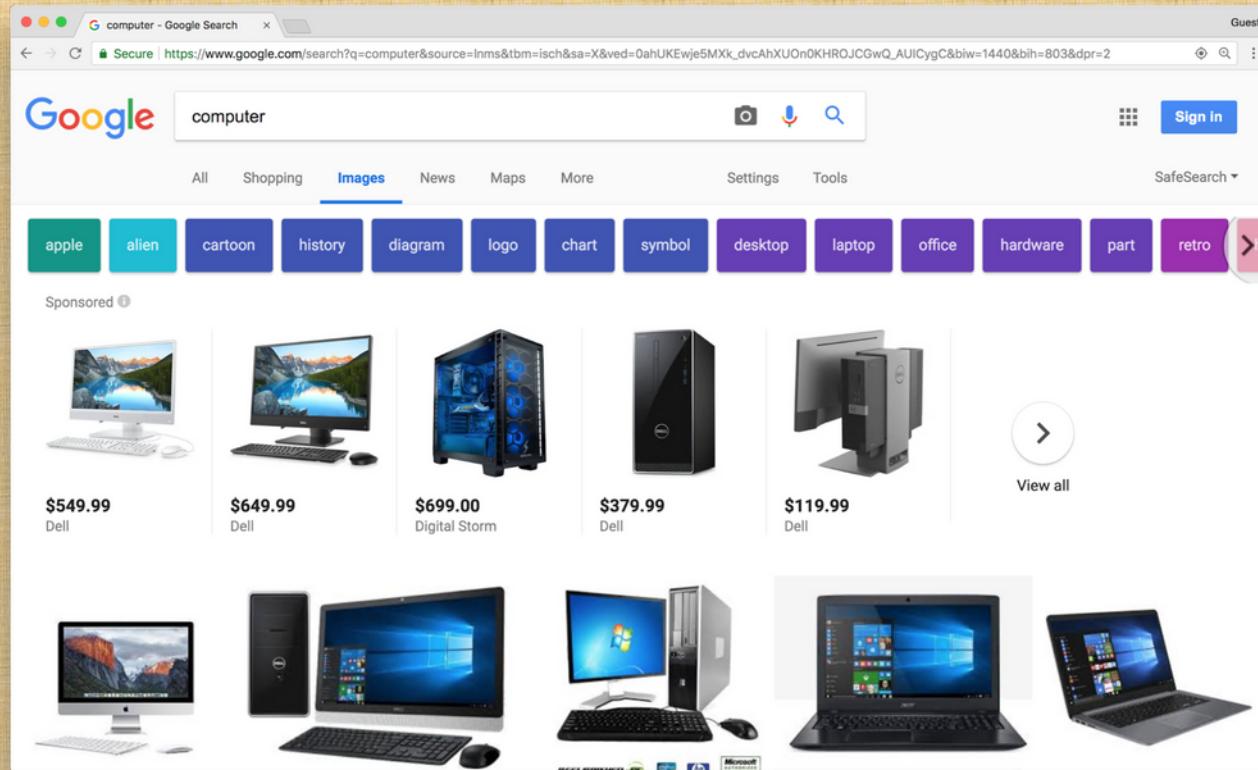
The Life of Transaction Data: Dapper

Stage	Overhead affects...	Retained
Instrumentation Executed	App App App Regional	100.00%
Buffered within app	network + storage WAN +	000.10%
process Flushed out of	storage	000.10%
process Centralized		000.10%
regionally Centralized		000.01%
globally		

Fatal Flaws: A Review

	Logs	Metrics	Dist. Traces
TCO scales gracefully	—	✓	✓
Accounts for all data (i.e., unsampled)	✓	✓	—
Immune to cardinality	✓	—	✓

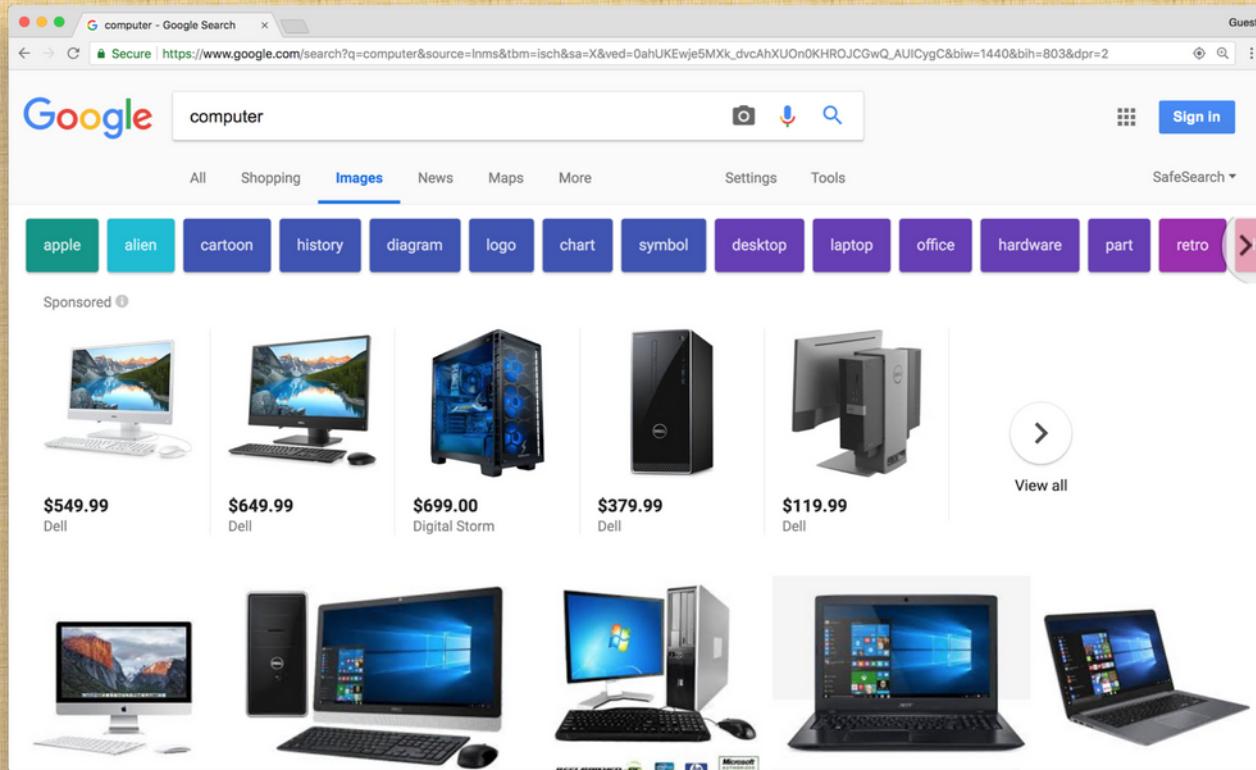
Data vs UI



Author: Nho Luong

Skill: DevOps Engineer Lead

Data vs UI



Author: Nho Luong

Skill: DevOps Engineer Lead

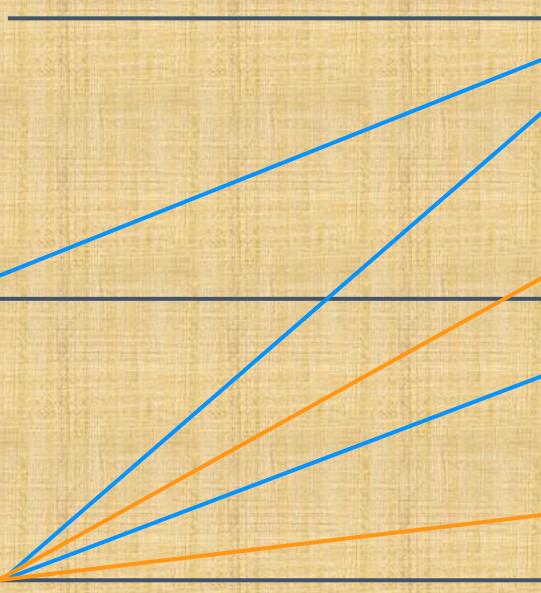
Data vs UI

Metric

s

Logs

Traces



Metrics, Logs, and Traces are
Just Data,

... not a feature or use case.

A New Scorecard for Observability

Author: Nho Luong

Skill: DevOps Engineer Lead

Observability: Quick Vocab Refresher

“SLI” = “Service Level Indicator”

TL;DR: An SLI is **an indicator of health** that a service’s **consumers** would care about.

... not an indicator of its inner workings

Observability: Two Fundamental **Goals**

- Gradually improving an SLI
- Rapidly restoring an SLI

NOW!!!!

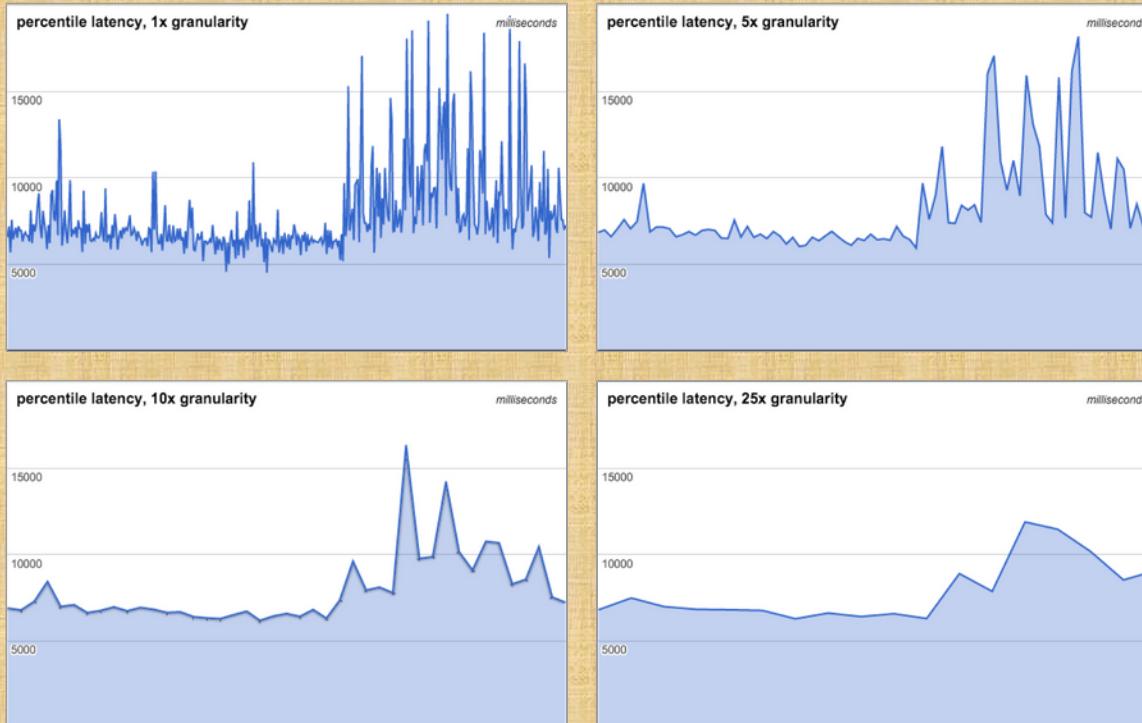
days, weeks, months...

Reminder: “SLI” = “Service Level Indicator”

Observability: Two Fundamental Activities

1. Detection: perfect SLI capture
2. Refinement: reduce the search space

An interlude about stats frequency



Author: Nho Luong

Skill: DevOps Engineer Lead

Scorecard >> **Detection**

Specificity:

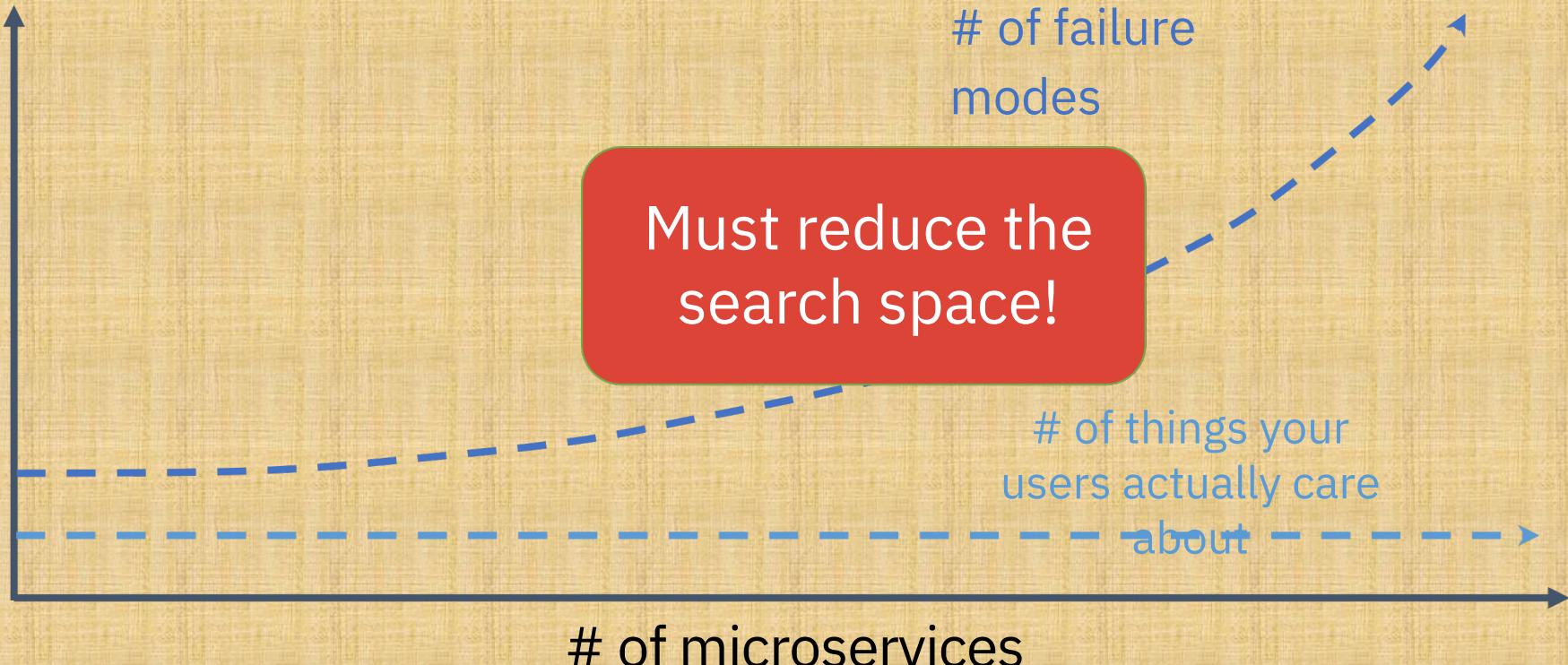
- Arbitrary dimensionality and cardinality
- Any layer of the stack, including mobile+web!

Fidelity:

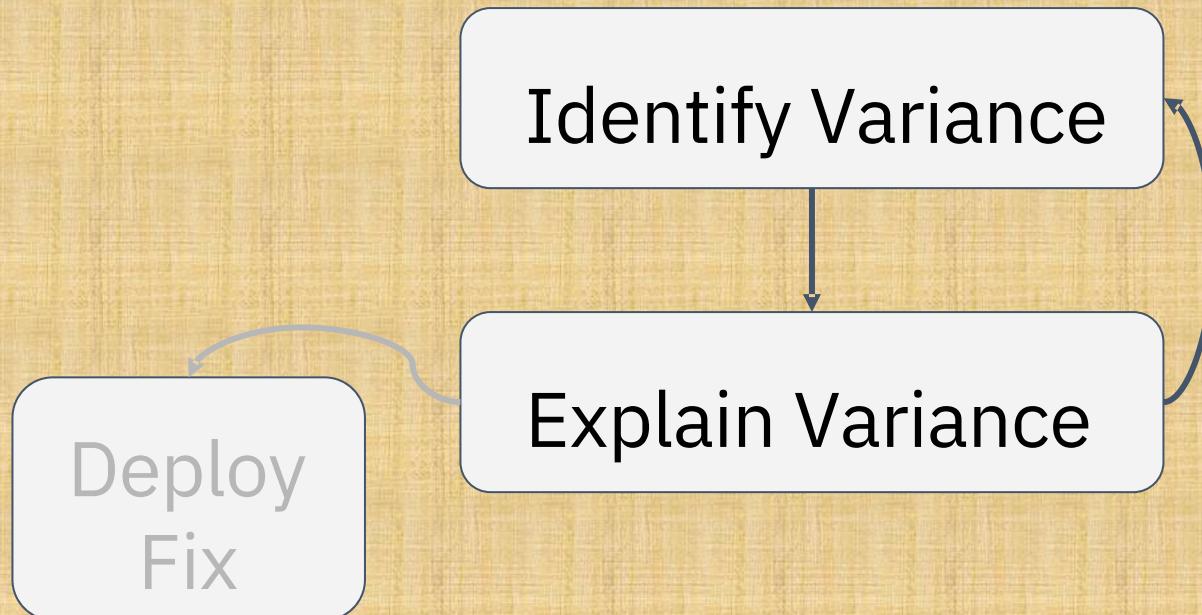
- Correct stats!!!
- High stats frequency (i.e., “beware smoothing”!)

Freshness: \leq 5 second lag

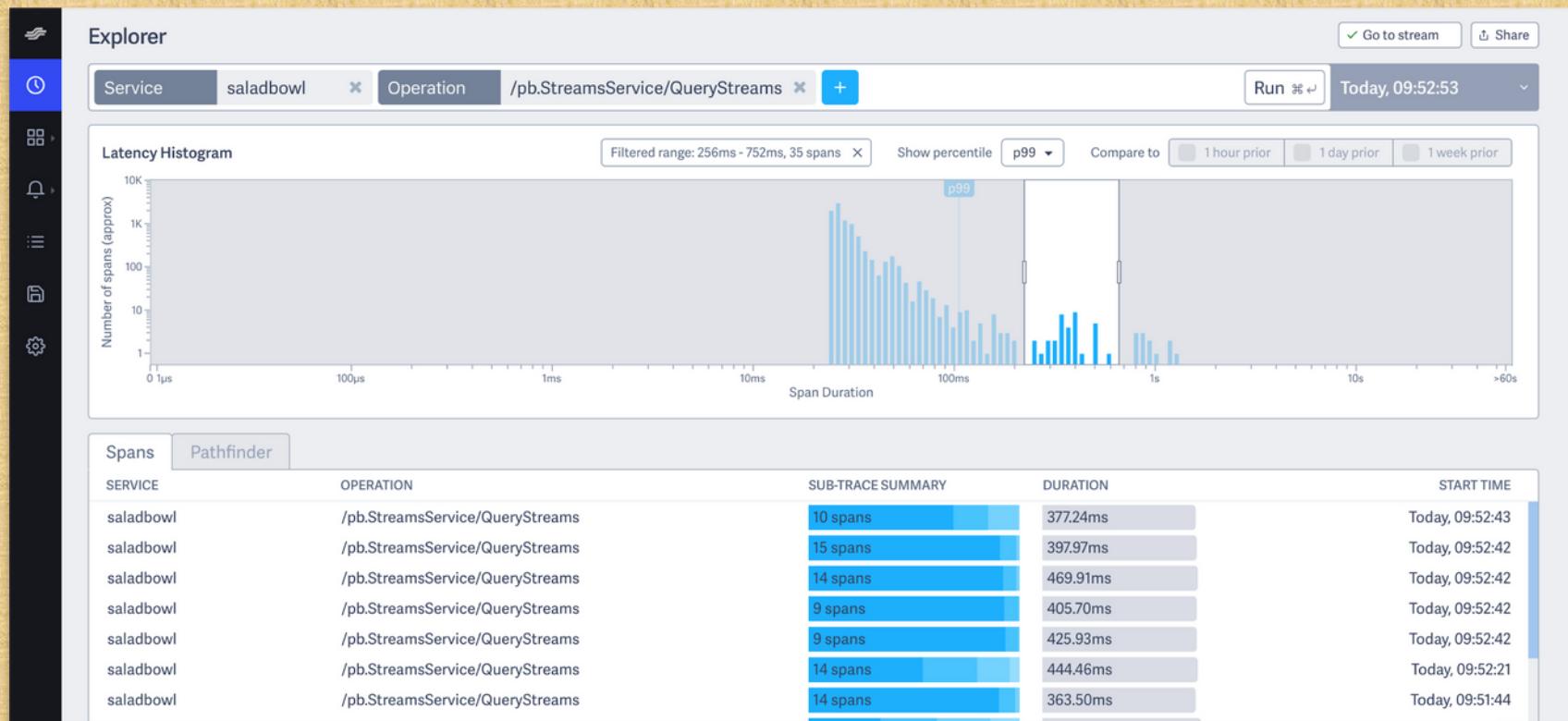
Scorecard >> Refinement



Scorecard >> Refinement



An interlude about variance and “p99”



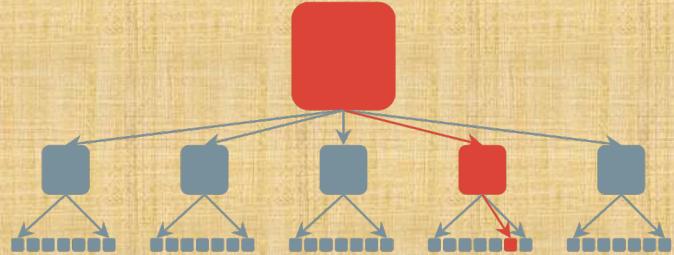
Scorecard >> Refinement

Identifying Variance:

- Cardinality: understand which tag changed
 - Robust stats: *histograms*(see prev slide)
- Data retention: always “Know What’s Normal”

Explaining variance:

- Correct stats!!!
- “Suppress the messengers” of microservice failures



Wrapping up...

(first, a hint at my
perspective)

The Life of Transaction Data: Dapper

Stage	Overhead affects...	Retained
Instrumentation Executed	App App App Regional	100.00%
Buffered within app process	network + storage WAN +	000.10%
Flushed out of process	storage	000.10%
Centralized regionally		000.10%
Centralized globally		000.01%

The Life of Transaction Data: Dapper ~~Dapper~~ LightStep

Stage	Overhead affects...	Retained
Instrumentation Executed	App App App Regional	100.00%
Buffered within app process	network + storage WAN +	100.00%
Flushed out of process	storage	100.00%
Centralized regionally		100.00%
Centralized globally		on-demand

An Observability Scorecard

Detection

- Specificity: unlimited cardinality, across the entire stack
- Fidelity:
 - correct stats, high stats frequency
- Freshness: ≤5 seconds

Refinement

- Identifying variance: unlimited cardinality, hi-fi histograms, data retention
- “Suppress the messengers”

Extra slides

Ideal Measurement: Robust



Author: Nho Luong

Skill: DevOps Engineer Lead

Ideal Measurement: High-Dimensional



Author: Nho Luong

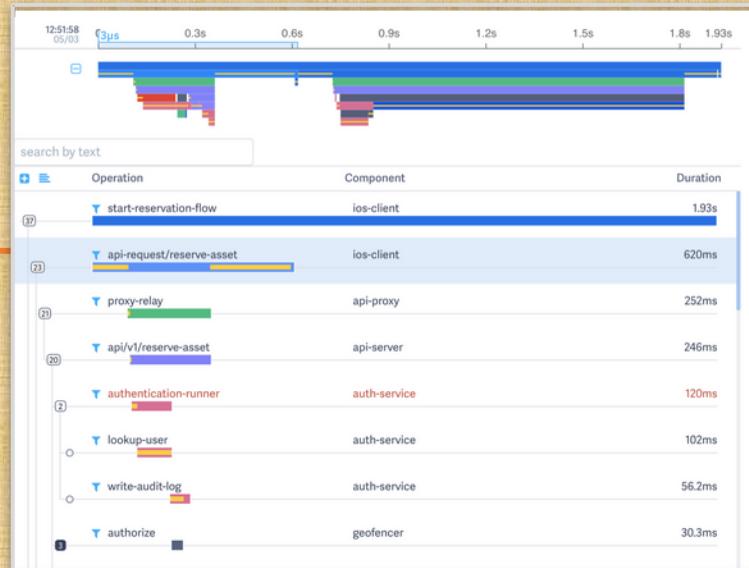
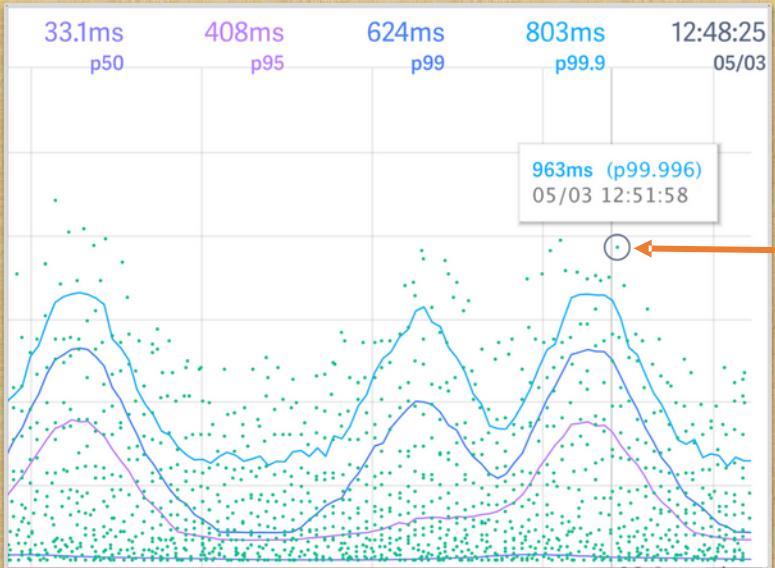
Skill: DevOps Engineer Lead

Ideal Refinement: Real-time

Must be able to test and eliminate hypotheses quickly

- Actual data must be $\leq 10\text{s}$ fresh UI /
- API latency must be very low

Ideal Refinement: Global

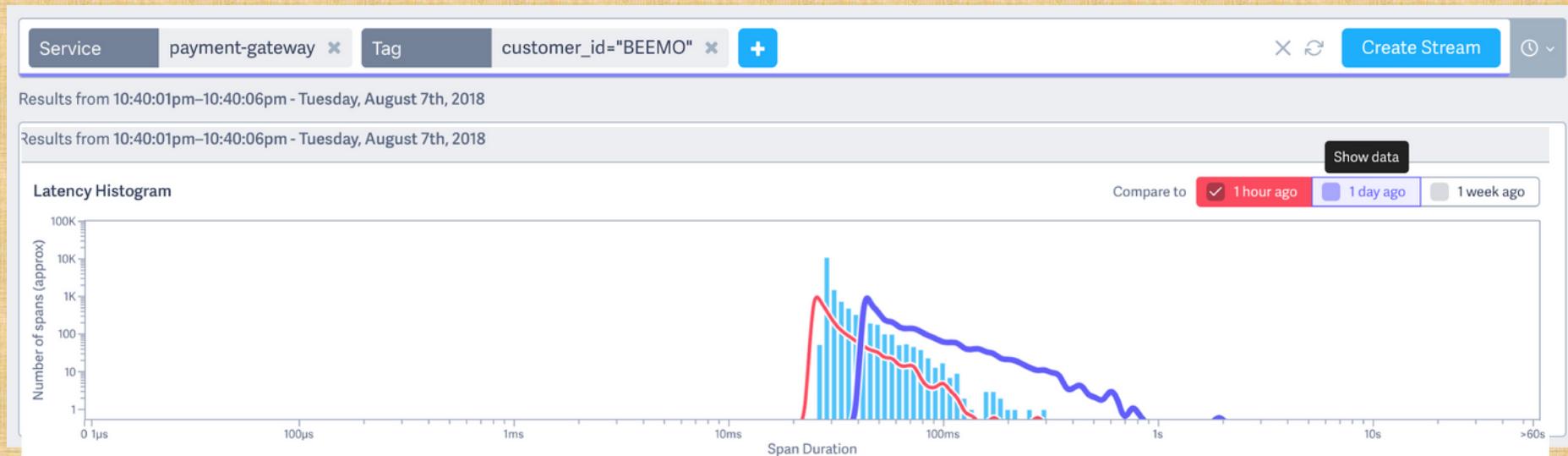


Author: Nho Luong

Skill: DevOps Engineer Lead

Ideal Refinement: Context-Rich

We can't expect humans to **know what's normal**



Author: Nho Luong

Skill: DevOps Engineer Lead



Thank You