QCon LONDON by InfoQ

Effective and Efficient Observability with OpenTelemetry

Daniel Gomez Blanco

Principal Engineer

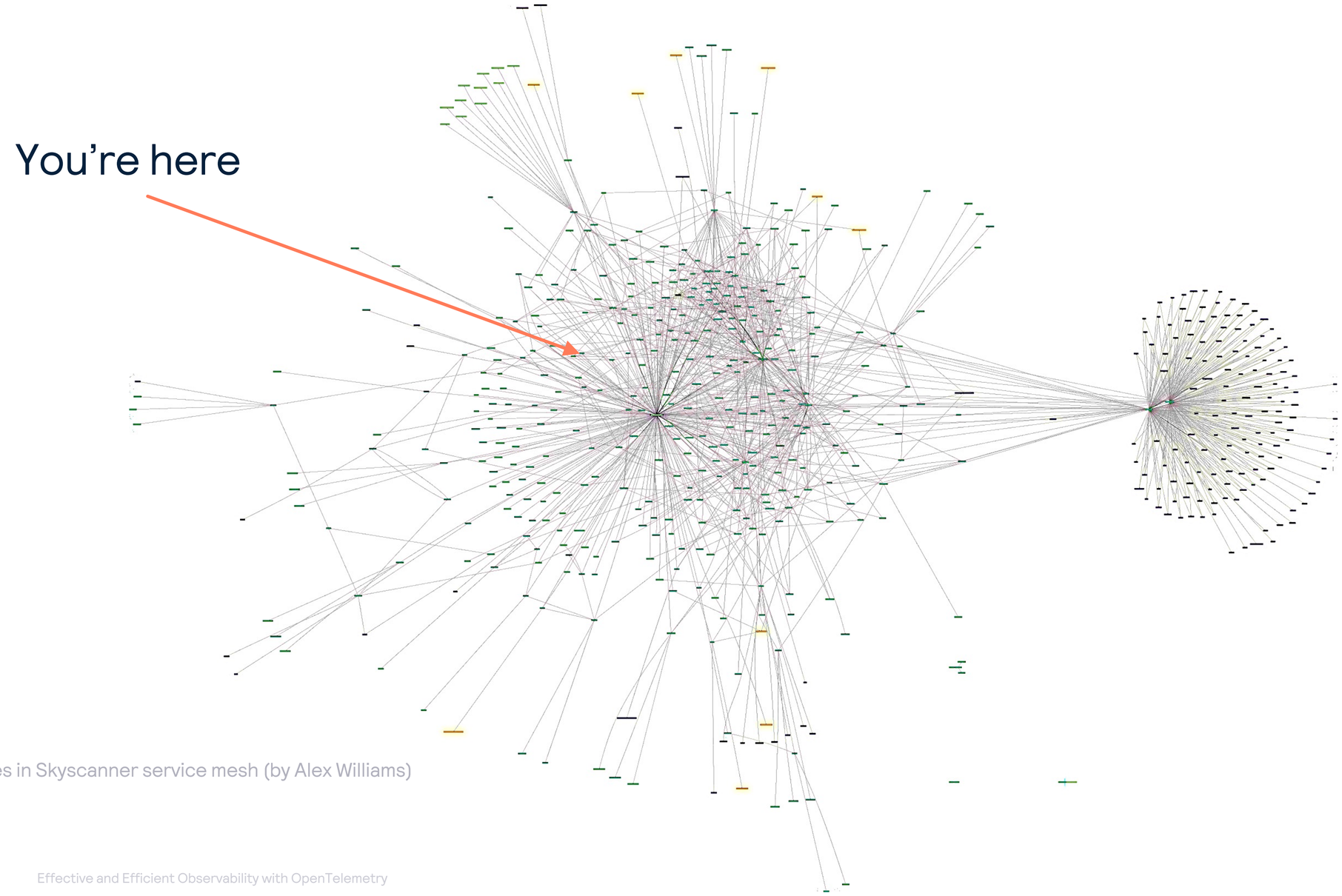
Skyscanner

When our systems **change**, how do we know **what** changed?

Easy when we are the only ones changing it…

```
if my_condition:
    print("I'm here")
    # Do some work
else:
    print("I'm there")
    # Do some other work
```

… not easy in a real distributed system



Source: service dependencies in Skyscanner service mesh (by Alex Williams)

Photo by Ross Cameron

A bit about me

- Joined Skyscanner in 2018 to work on performance and resource optimisation
- Principal Engineer leading observability strategy since 2020
- 12 years as platform engineer in organisations from 5 to 2,500 employees
- Author of Practical OpenTelemetry: Adopting Open Observability Standards Across Your Organization (Apress, 2023)

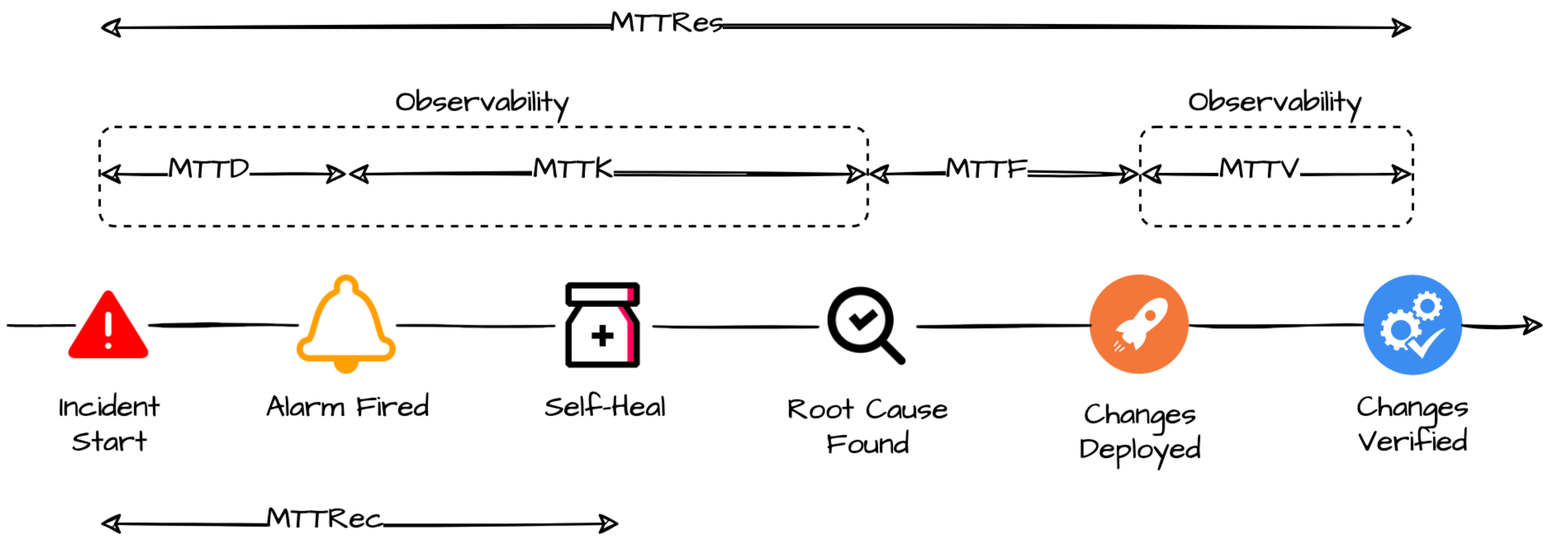
Agenda

1. Why observability matters
2. How open standards help observability
3. Rolling out OpenTelemetry
4. Adopting observability in practice

Agenda

1. **Why observability matters**
2. How open standards help observability
3. Rolling out OpenTelemetry
4. Adopting observability in practice

Observability within incident response



Source: Practical OpenTelemetry: Adopting Open Observability Standards Across Your Organization

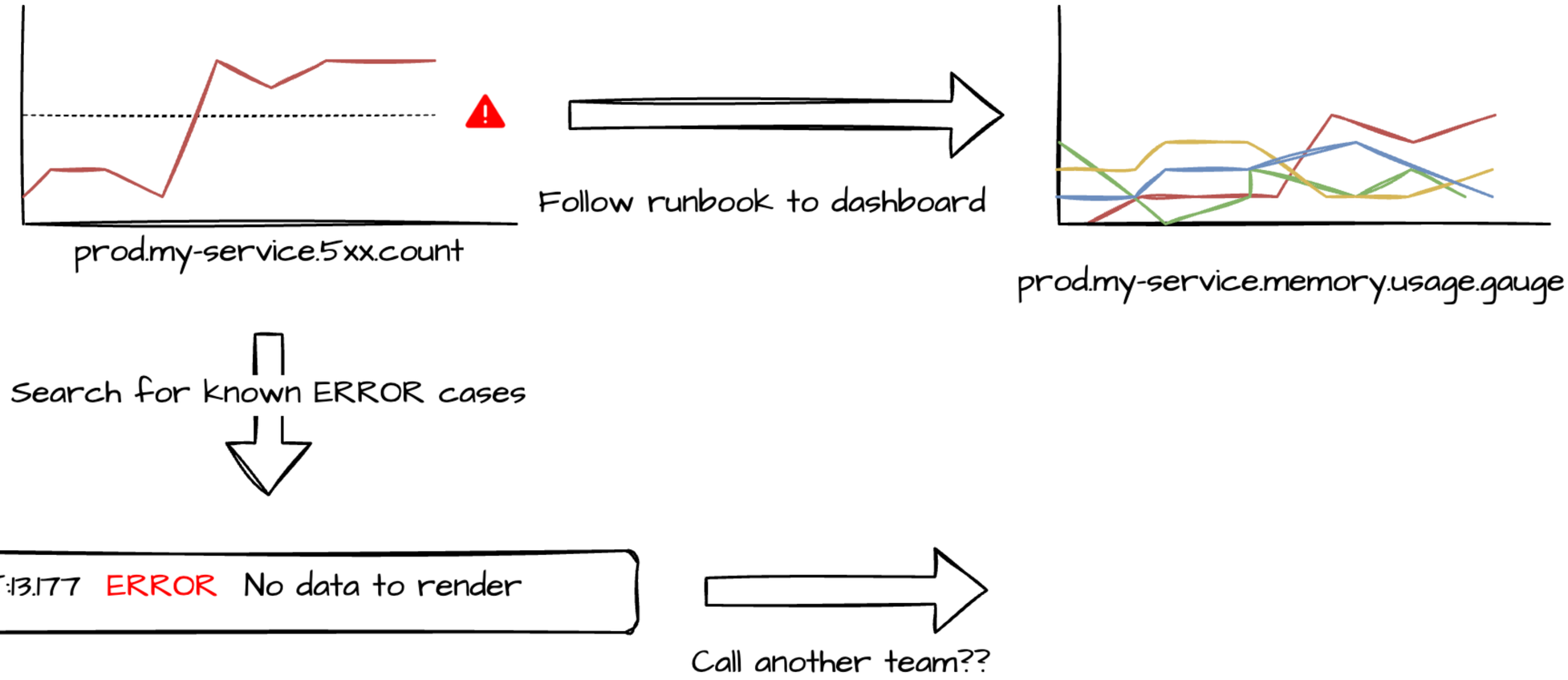
Is my system behaving as expected?

Reducing Mean-Time-to-Detect and Mean-Time-To-Verify

Why is my system not behaving as expected?

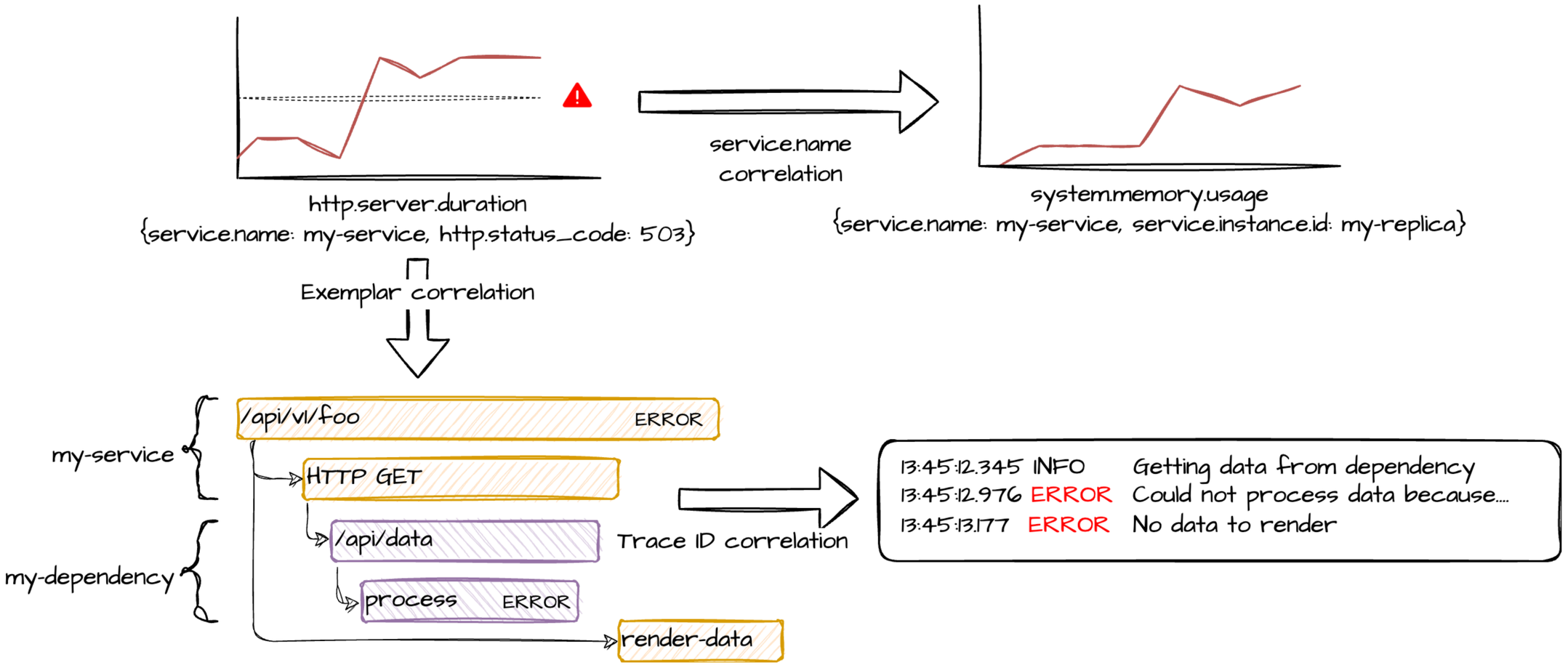
Reducing Mean-Time-to-Know

Debugging that relies on past experience



Now what?

Debugging that relies on context



Effective observability means...

High granularity

Detailed telemetry data corresponding to individual operations within system transactions

Rich context

Considering multiple telemetry signals and dependencies under one single holistic view of the system

Signal correlation

Linking metrics, traces and logs under one single stream of events

Service correlation

Relating telemetry from different services part of the same common operation

Open standards

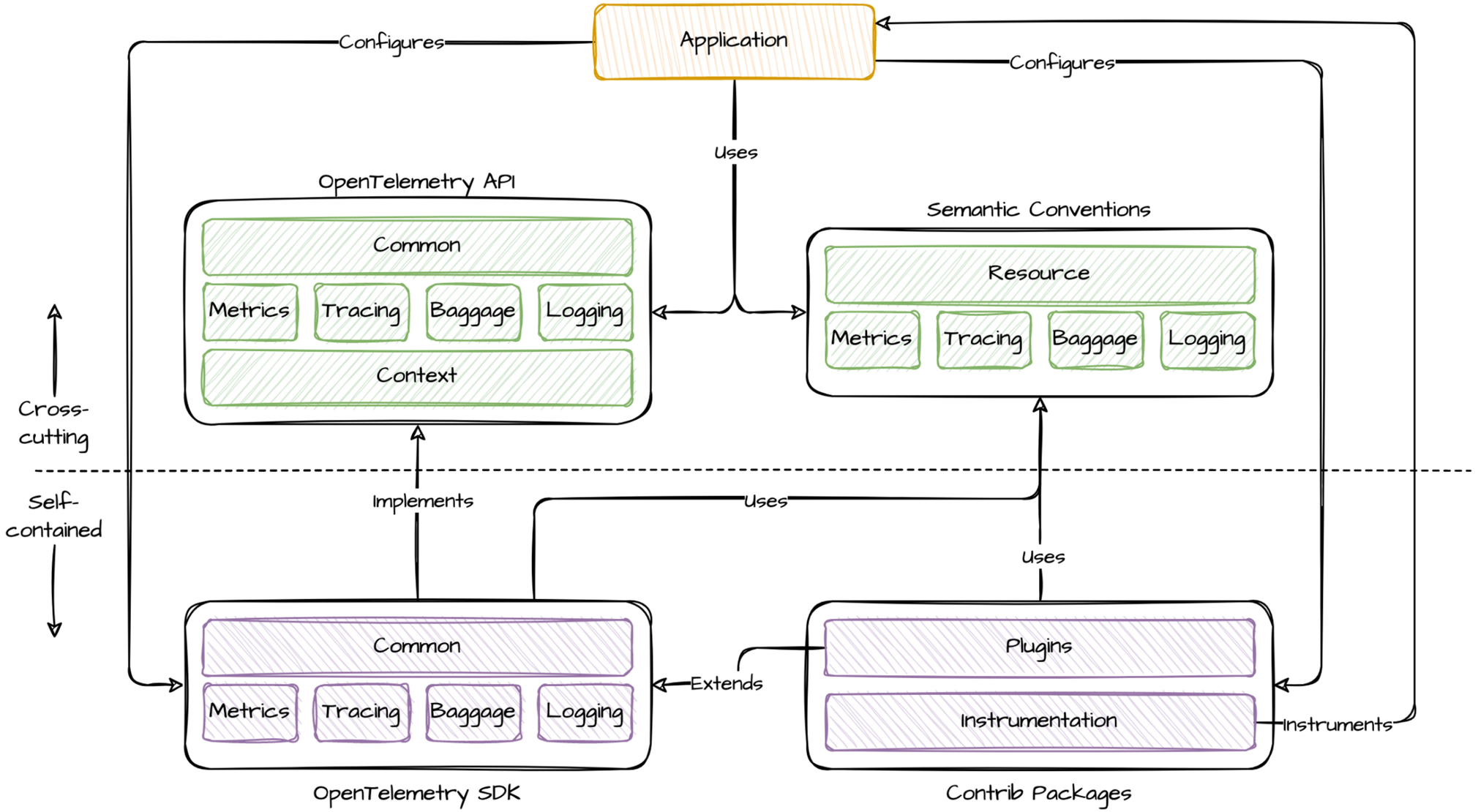
Out-of-the-box telemetry instrumented by experts, following open standards across all platforms

Agenda

1. Why observability matters
2. **How open standards help observability**
3. Telemetry signals and their purpose
4. Adopting observability in practice

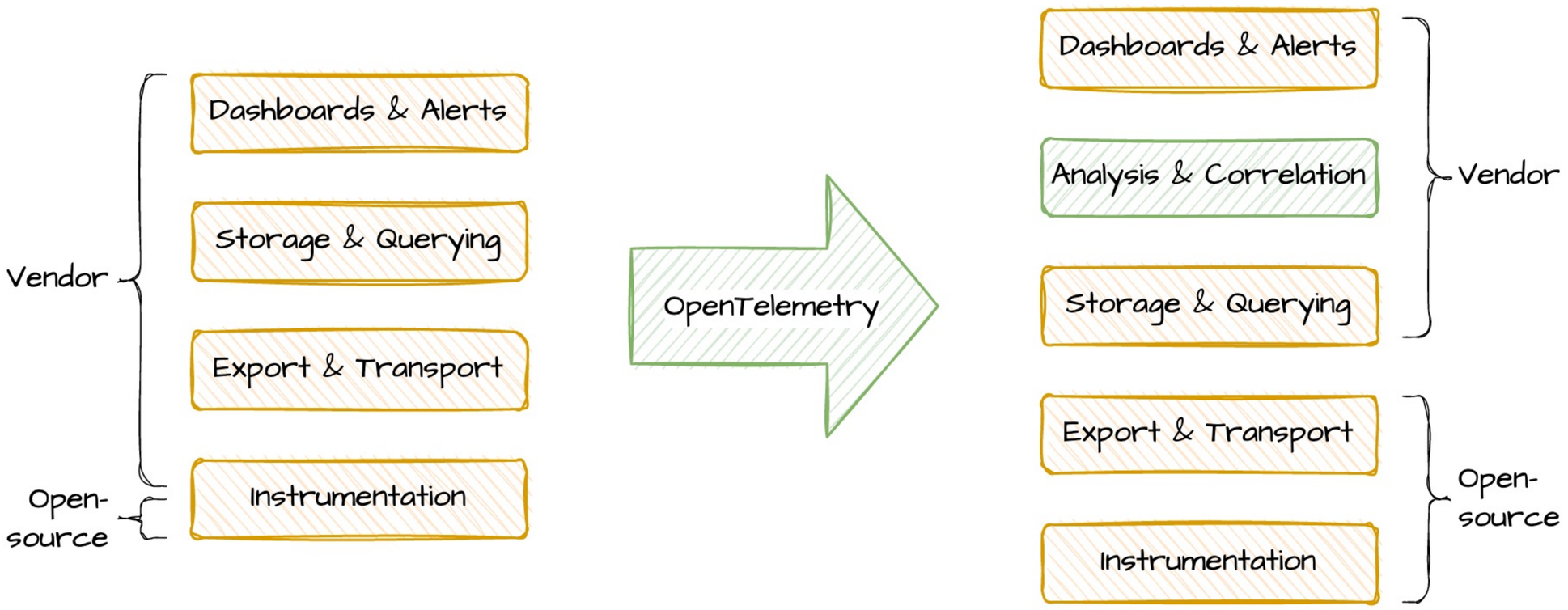
OpenTelemetry's Mission

To **enable effective observability** by making high-quality, portable telemetry ubiquitous



Practical OpenTelemetry: Adopting Open Observability Standards Across Your Organization

Influence in buy-vs-build decisions

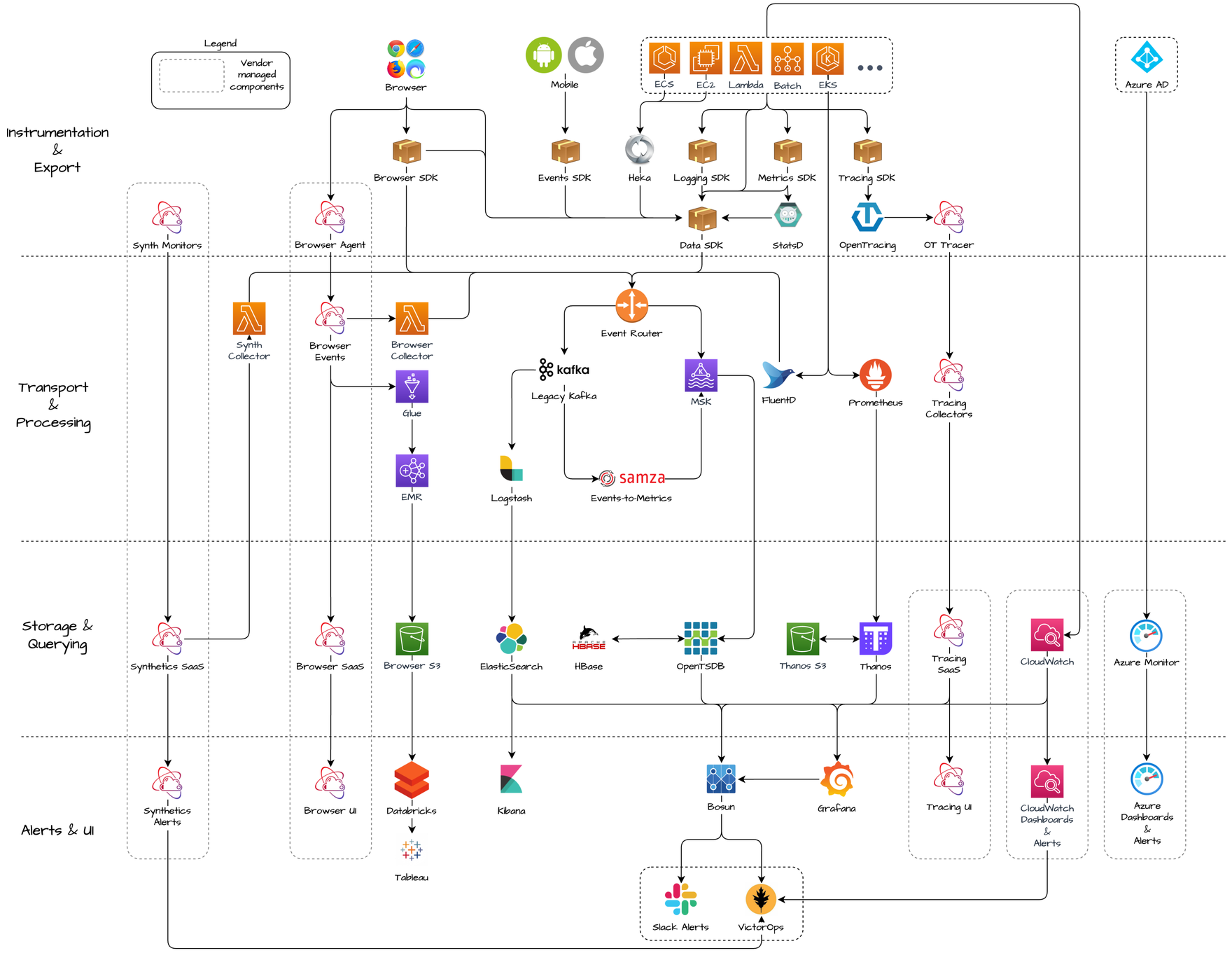


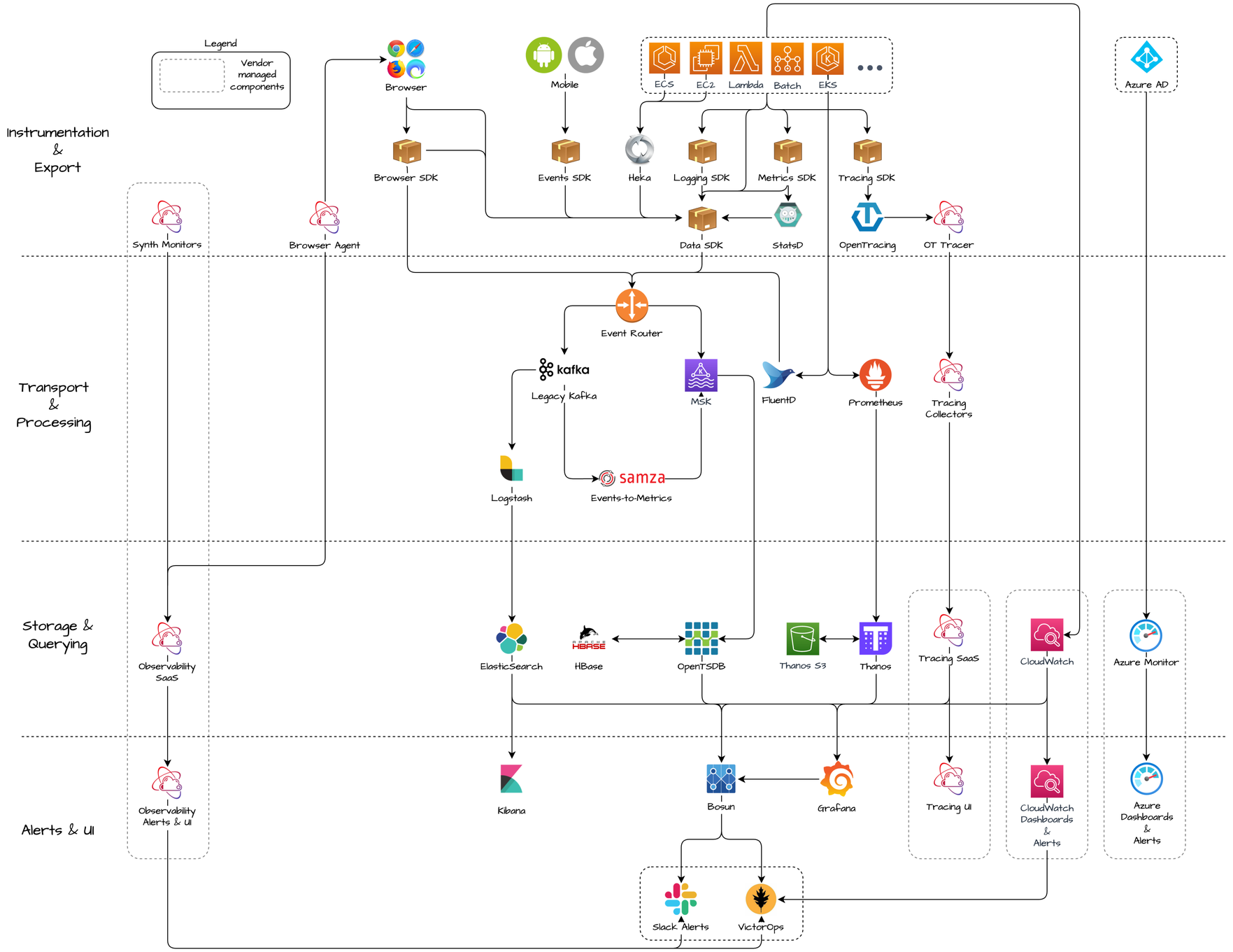
Practical OpenTelemetry: Adopting Open Observability Standards Across Your Organization

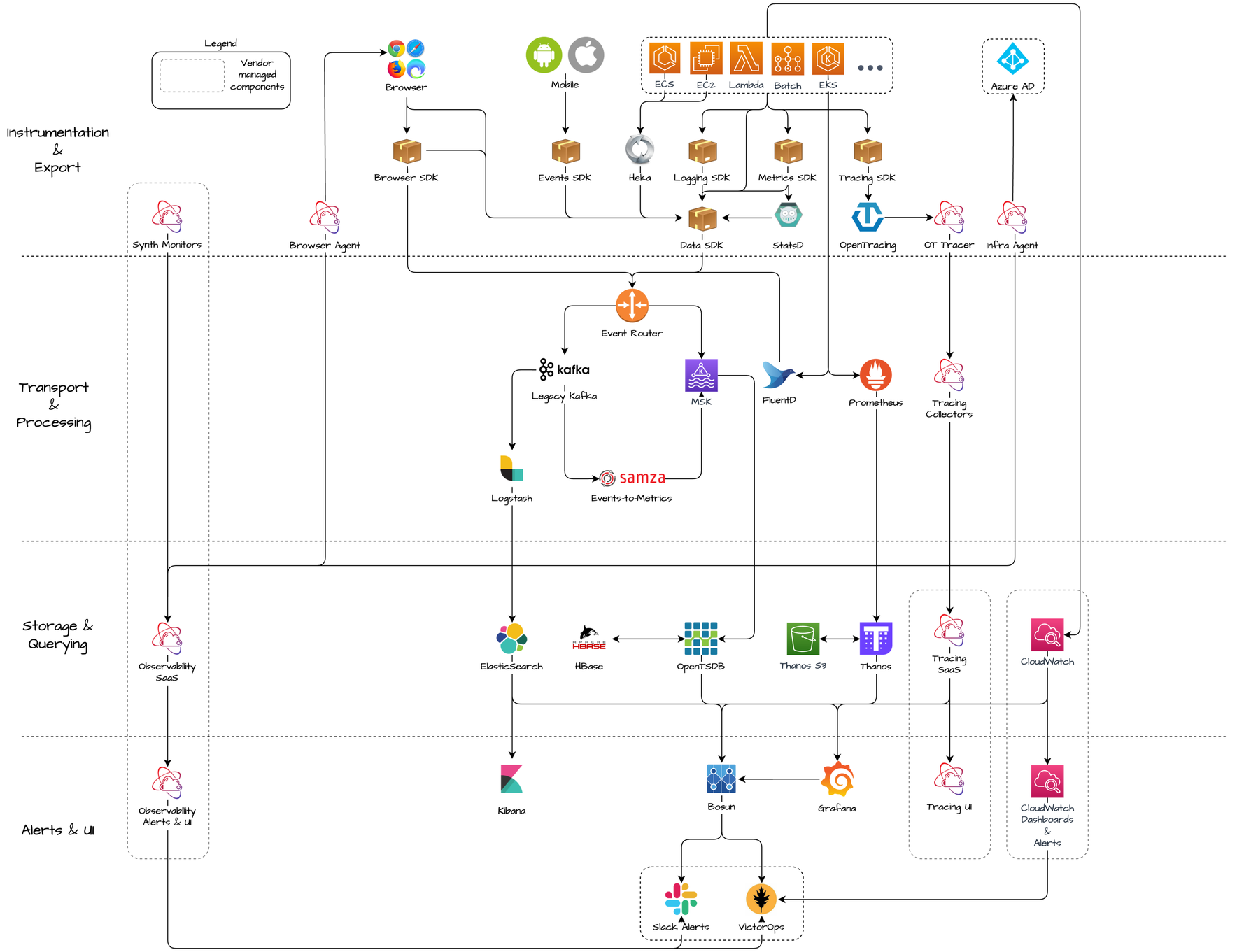
Agenda

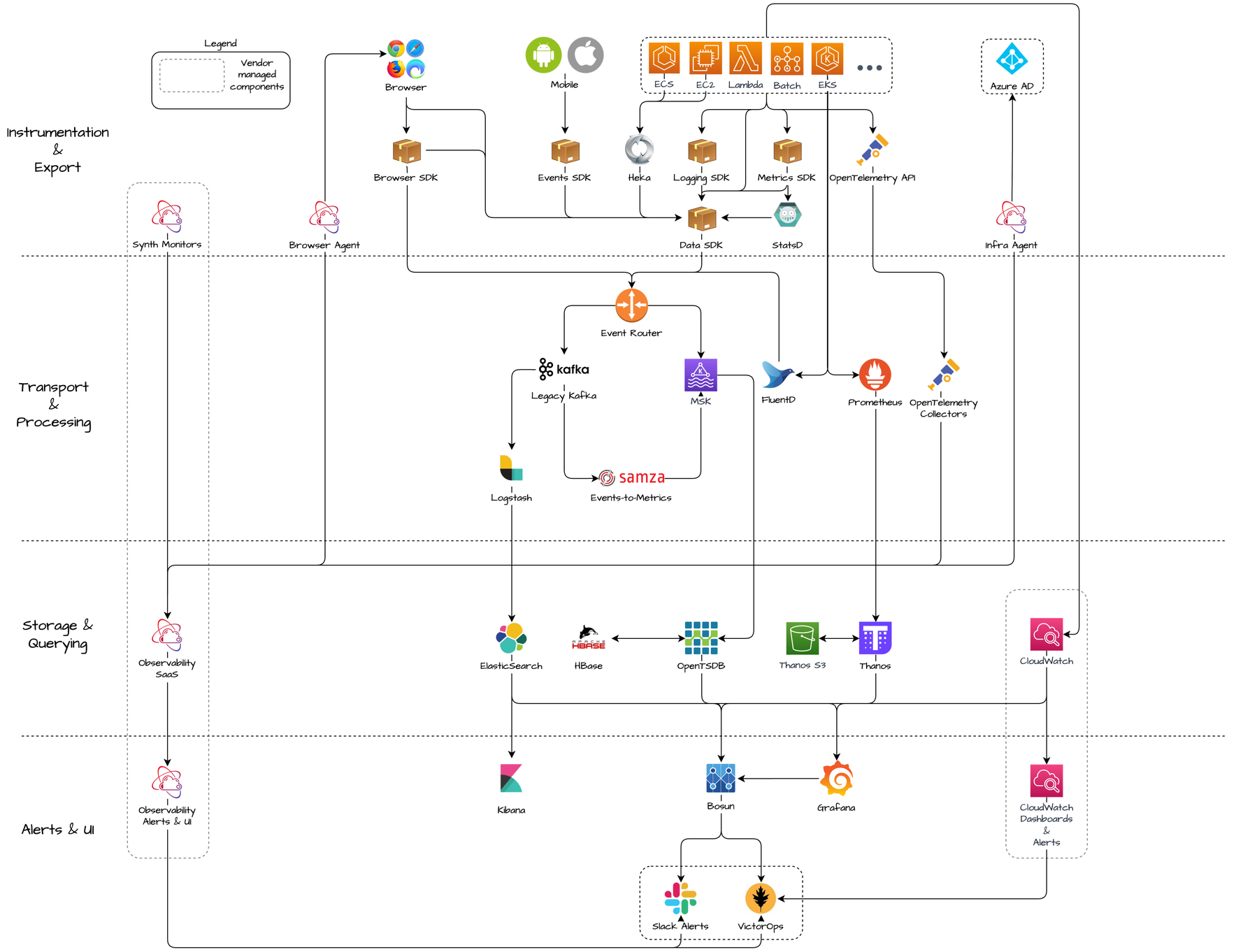
1. Why observability matters
2. How open standards help observability
3. **Rolling out OpenTelemetry**
4. Adopting observability in practice

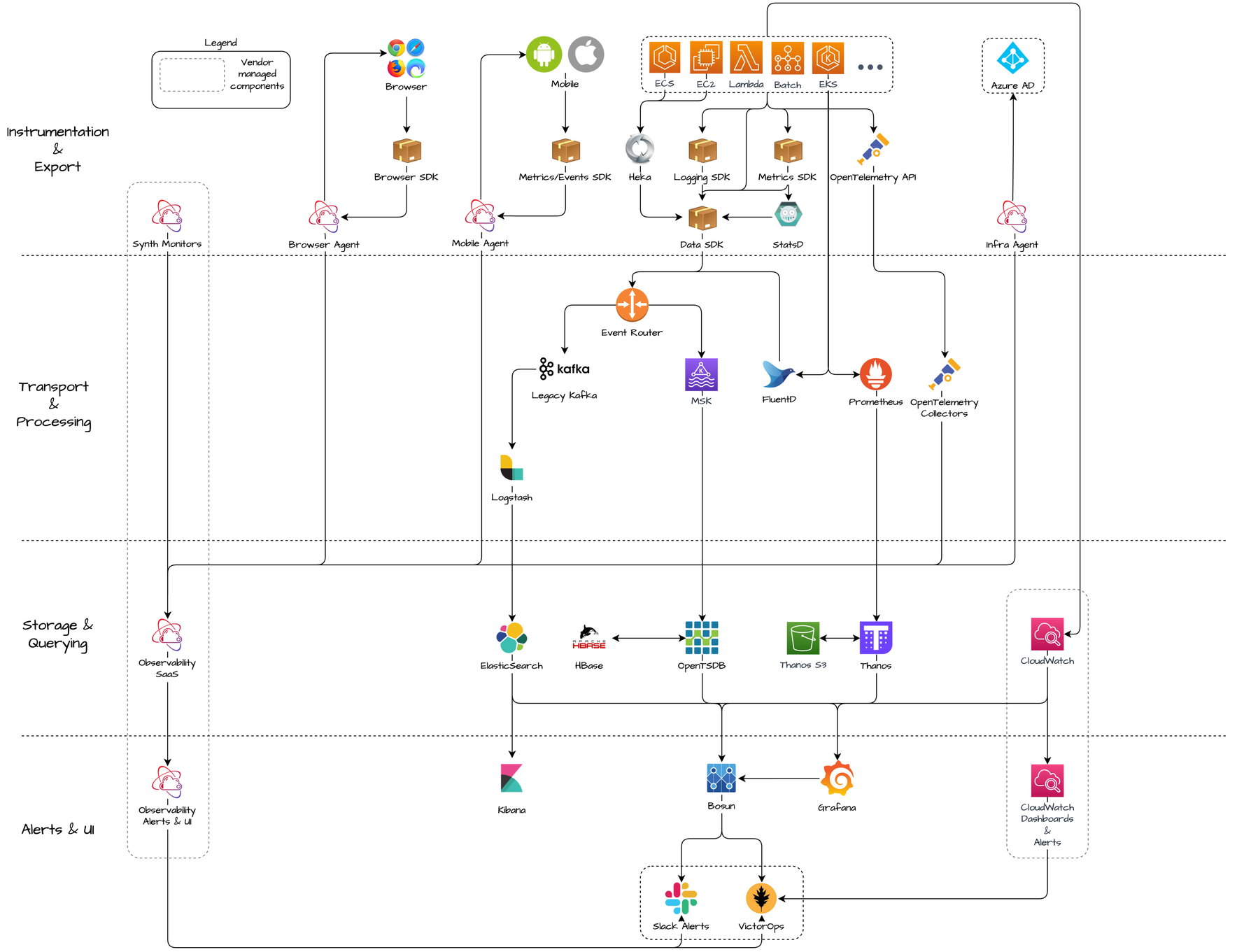
Make the
golden path
the path of
least resistance

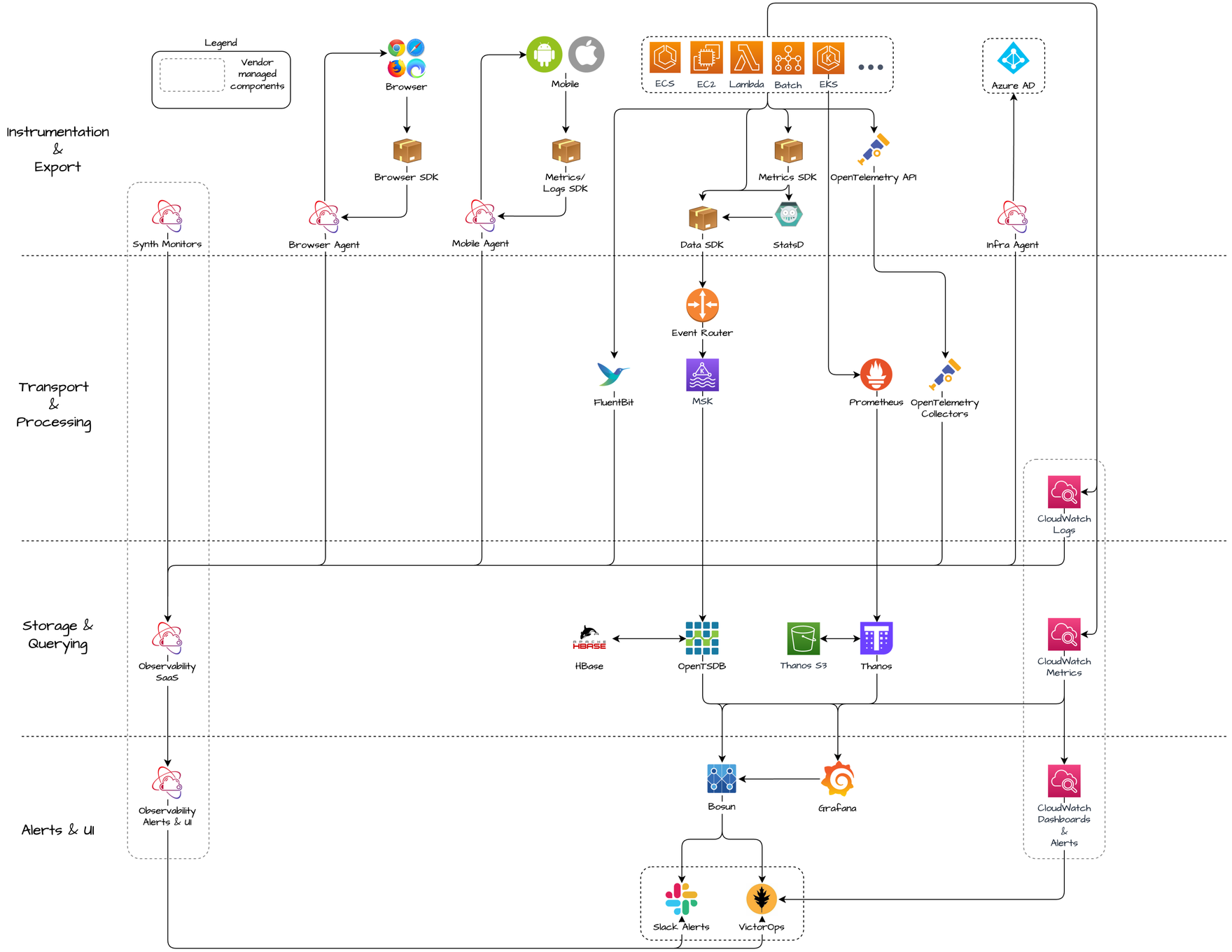

Legend
Vendor managed components
Instrumentation & Export
Transport & Processing
Storage & Querying
Alerts & UI
Browser
Mobile
ECS
EC2
Lambda
Batch
EKS
Azure AD
Browser SDK
Events SDK
Heka
Logging SDK
Metrics SDK
Tracing SDK
Synth Monitors
Browser Agent
Data SDK
StatsD
OpenTracing
OT Tracer
Synth Collector
Browser Events
Browser Collector
Event Router
Glue
kafka
Legacy Kafka
MSK
FluentD
Prometheus
Tracing Collectors
EMR
Logstash
samza
Events-to-Metrics
Synthetics SaaS
Browser SaaS
Browser S3
ElasticSearch
HBase
OpenTSDB
Thanos S3
Thanos
Tracing SaaS
CloudWatch
Azure Monitor
Synthetics Alerts
Browser UI
Databricks
Kibana
Bosun
Grafana
Tracing UI
CloudWatch Dashboards & Alerts
Azure Dashboards & Alerts
Tableau
Slack Alerts
VictorOps

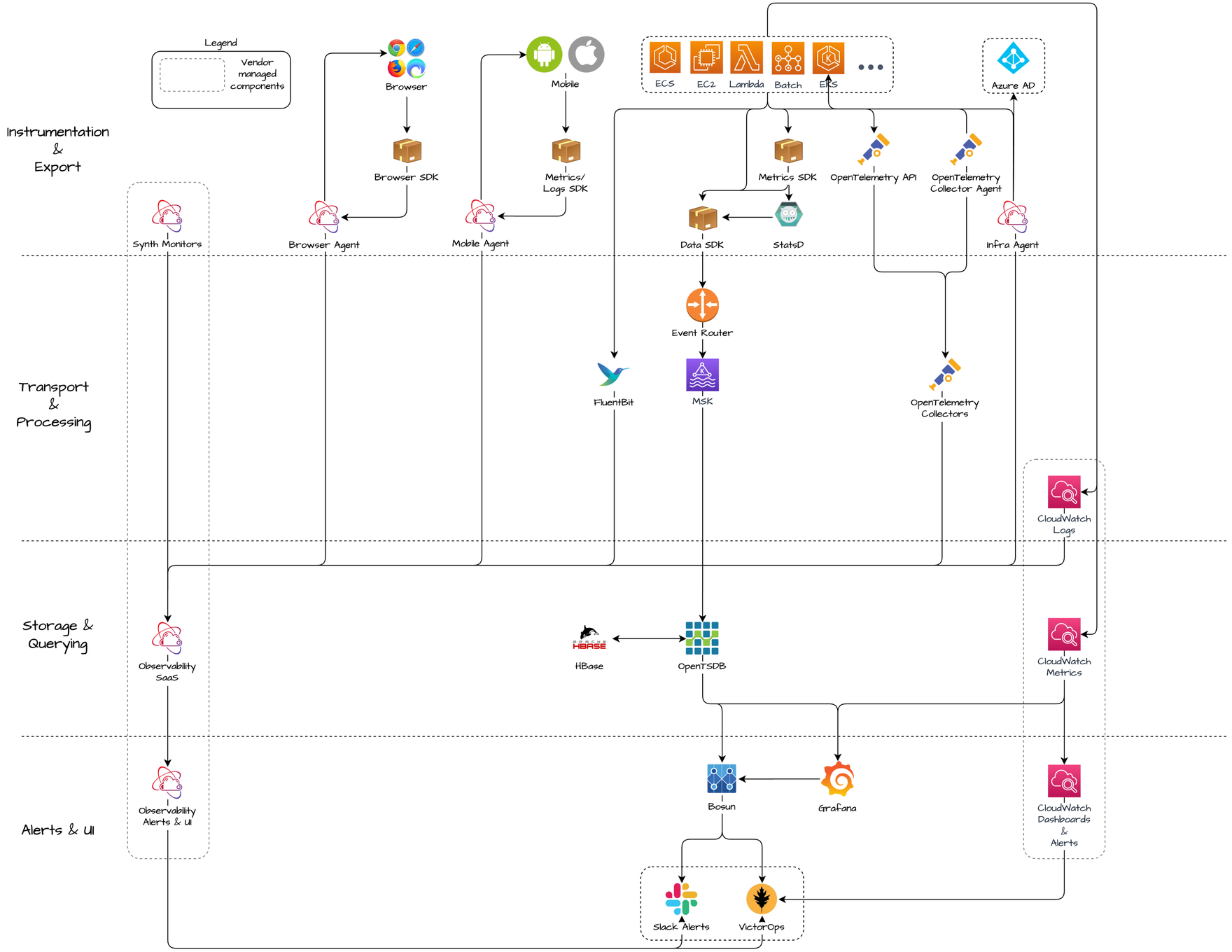

Legend
Vendor managed components
Instrumentation & Export
Transport & Processing
Storage & Querying
Alerts & UI
Browser
Mobile
ECS
EC2
Lambda
Batch
EKS
Azure AD
Browser SDK
Events SDK
Heka
Logging SDK
Metrics SDK
Tracing SDK
Synth Monitors
Browser Agent
Data SDK
StatsD
OpenTracing
OT Tracer
Event Router
kafka
Legacy Kafka
MSK
FluentD
Prometheus
Tracing Collectors
Logstash
samza
Events-to-Metrics
Observability SaaS
ElasticSearch
HBase
OpenTSDB
Thanos S3
Thanos
Tracing SaaS
CloudWatch
Azure Monitor
Observability Alerts & UI
Kibana
Bosun
Grafana
Tracing UI
CloudWatch Dashboards & Alerts
Azure Dashboards & Alerts
Slack Alerts
VictorOps

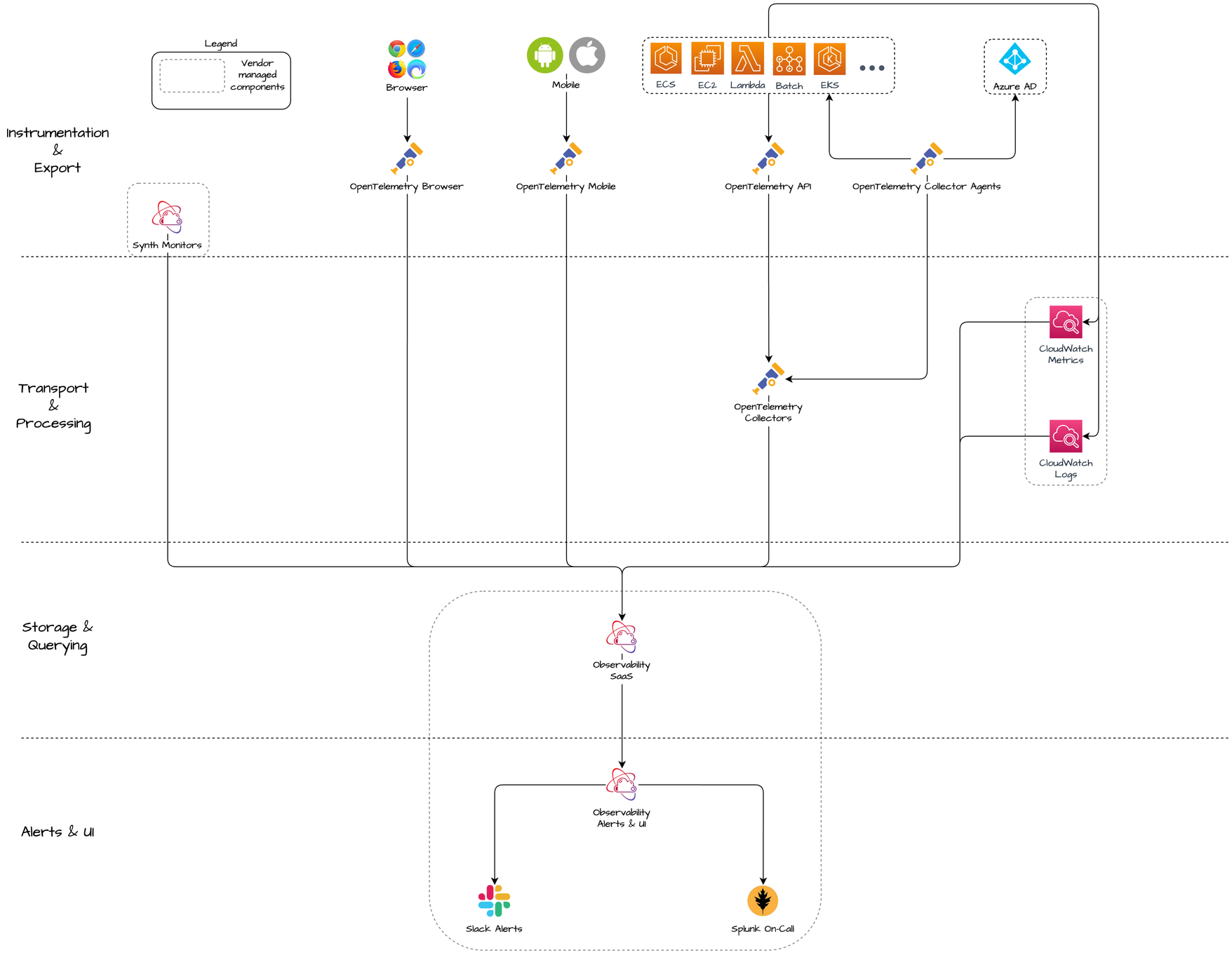

Legend
Vendor managed components
Instrumentation & Export
Transport & Processing
Storage & Querying
Alerts & UI
Browser
Mobile
ECS
EC2
Lambda
Batch
EKS
Azure AD
Browser SDK
Events SDK
Heka
Logging SDK
Metrics SDK
Tracing SDK
Synth Monitors
Browser Agent
Data SDK
StatsD
OpenTracing
OT Tracer
Infra Agent
Event Router
kafka
Legacy Kafka
MSK
FluentD
Prometheus
Tracing Collectors
Logstash
samza
Events-to-Metrics
Observability SaaS
ElasticSearch
HBase
OpenTSDB
Thanos S3
Thanos
Tracing SaaS
CloudWatch
Kibana
Bosun
Grafana
Tracing UI
CloudWatch Dashboards & Alerts
Observability Alerts & UI
Slack Alerts
VictorOps


Legend
Vendor managed components
Instrumentation & Export
Transport & Processing
Storage & Querying
Alerts & UI
Browser
Mobile
ECS
EC2
Lambda
Batch
EKS
Azure AD
Browser SDK
Events SDK
Heka
Logging SDK
Metrics SDK
OpenTelemetry API
Synth Monitors
Browser Agent
Data SDK
StatsD
Infra Agent
Event Router
kafka
Legacy Kafka
MSK
FluentD
Prometheus
OpenTelemetry Collectors
Logstash
samza
Events-to-Metrics
Observability SaaS
ElasticSearch
HBase
OpenTSDB
Thanos S3
Thanos
CloudWatch
Observability Alerts & UI
Kibana
Bosun
Grafana
CloudWatch Dashboards & Alerts
Slack Alerts
VictorOps

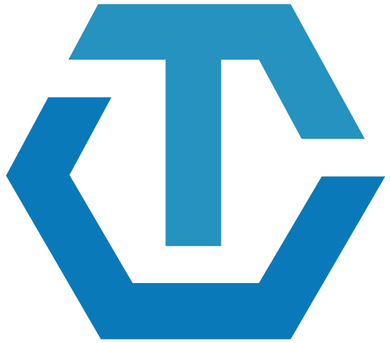

Legend
Vendor managed components
Instrumentation & Export
Transport & Processing
Storage & Querying
Alerts & UI
Browser
Mobile
ECS
EC2
Lambda
Batch
EKS
Azure AD
Browser SDK
Metrics/Events SDK
Heka
Logging SDK
Metrics SDK
OpenTelemetry API
Synth Monitors
Browser Agent
Mobile Agent
Data SDK
StatsD
Infra Agent
Event Router
kafka
Legacy Kafka
MSK
FluentD
Prometheus
OpenTelemetry Collectors
Logstash
Observability SaaS
ElasticSearch
HBase
OpenTSDB
Thanos S3
Thanos
CloudWatch
Kibana
Bosun
Grafana
Observability Alerts & UI
CloudWatch Dashboards & Alerts
Slack Alerts
VictorOps


Legend
Vendor managed components
Instrumentation & Export
Transport & Processing
Storage & Querying
Alerts & UI
Browser
Mobile
ECS
EC2
Lambda
Batch
EKS
Azure AD
Browser SDK
Metrics/ Logs SDK
Metrics SDK
OpenTelemetry API
Synth Monitors
Browser Agent
Mobile Agent
Data SDK
StatsD
Infra Agent
Event Router
FluentBit
MSK
Prometheus
OpenTelemetry Collectors
CloudWatch Logs
Observability SaaS
HBase
OpenTSDB
Thanos S3
Thanos
CloudWatch Metrics
Bosun
Grafana
Observability Alerts & UI
CloudWatch Dashboards & Alerts
Slack Alerts
VictorOps


Legend
Vendor managed components
Instrumentation & Export
Transport & Processing
Storage & Querying
Alerts & UI
Browser
Mobile
ECS
EC2
Lambda
Batch
EKS
Azure AD
Browser SDK
Metrics/ Logs SDK
Metrics SDK
OpenTelemetry API
OpenTelemetry Collector Agent
Synth Monitors
Browser Agent
Mobile Agent
Data SDK
StatsD
Infra Agent
Event Router
FluentBit
MSK
OpenTelemetry Collectors
CloudWatch Logs
Observability SaaS
HBase
OpenTSDB
CloudWatch Metrics
Bosun
Grafana
Observability Alerts & UI
CloudWatch Dashboards & Alerts
Slack Alerts
VictorOps


Legend
Vendor managed components
Instrumentation & Export
Transport & Processing
Storage & Querying
Alerts & UI
Browser
Mobile
ECS
EC2
Lambda
Batch
EKS
Azure AD
OpenTelemetry Browser
OpenTelemetry Mobile
OpenTelemetry API
OpenTelemetry Collector Agents
Synth Monitors
CloudWatch Metrics
CloudWatch Logs
OpenTelemetry Collectors
Observability SaaS
Observability Alerts & UI
Slack Alerts
Splunk On-Call

Compatibility with existing solutions



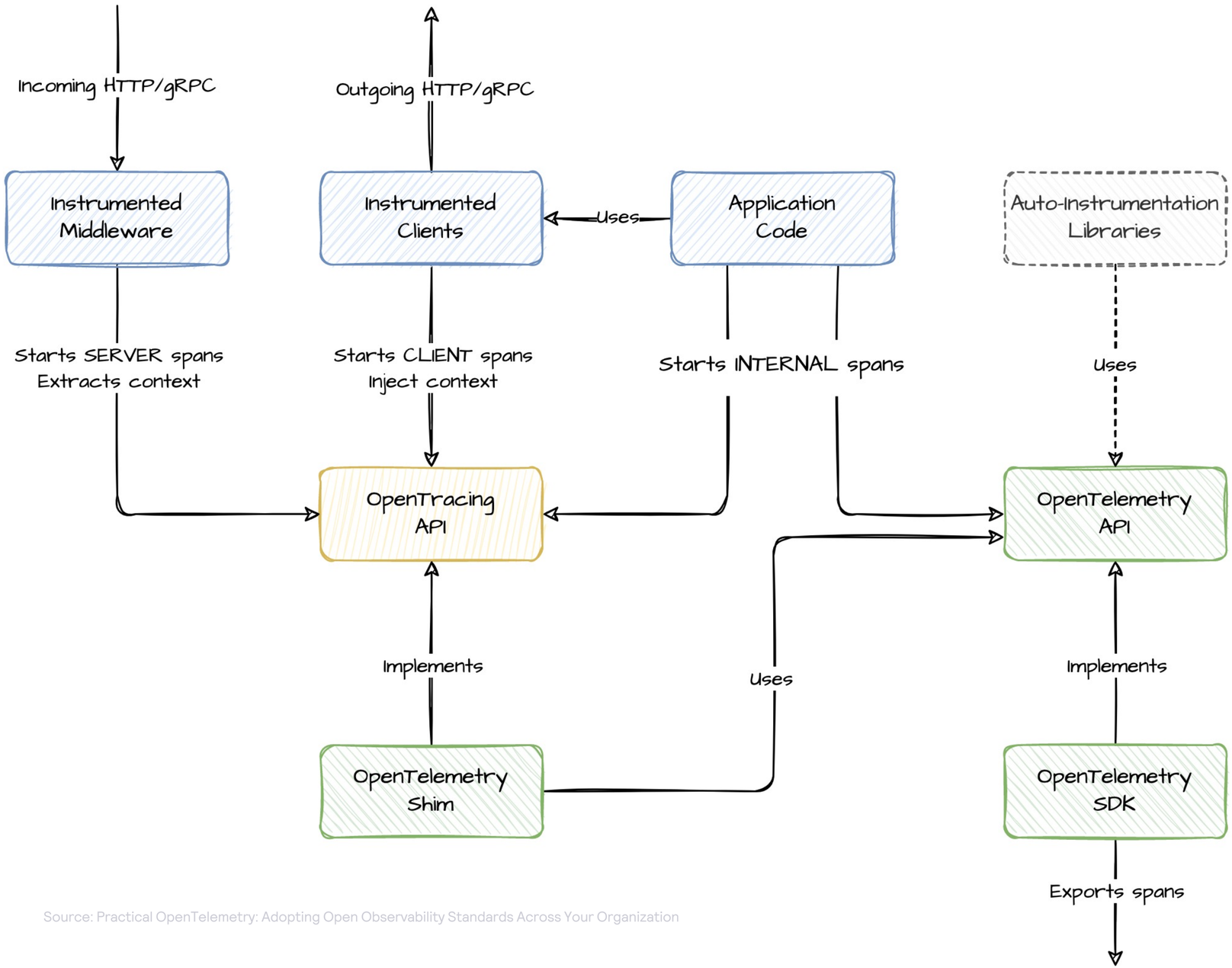
OPENTRACING

+

OpenCensus

=

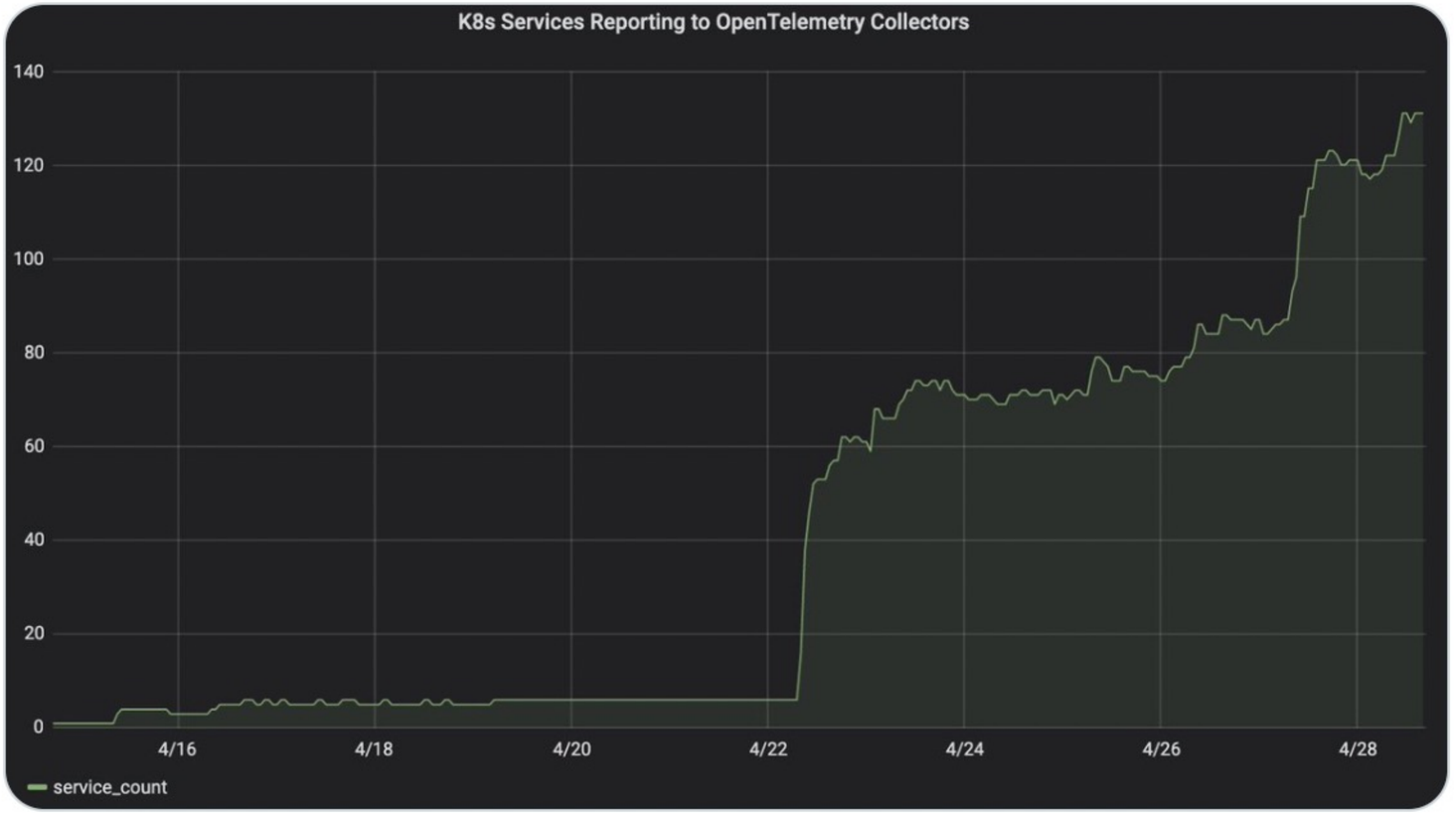
OpenTelemetry



Source: Practical OpenTelemetry: Adopting Open Observability Standards Across Your Organization

Dan Gomez Blanco
@dan_gomezblanco

It pays off when your migration to @opentelemetry involves a minor version bump 🙂



9:19 PM · Apr 28, 2021

OTel Collectors are incredibly powerful

Receiving OTLP, Zipkin, Prometheus

Removing unwanted attributes

Unsetting span status for spans matching regex

Generate metrics from spans (bye Istio Mixer)

Converting from cumulative to delta temporality

Renaming attributes to follow semantic conventions

Exporting data OTLP and Prometheus

>1.8M

Spans per second

>90k

Traces per second

<125

Total used CPU cores

<100

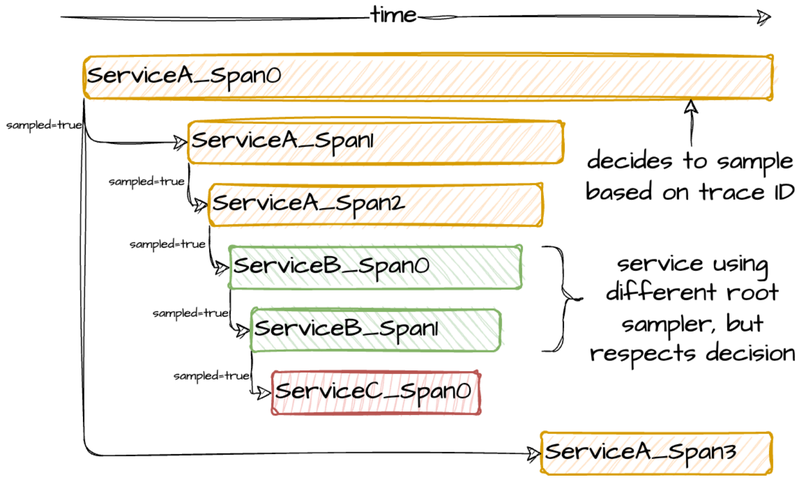
Replicas

Agenda

1. Why observability matters
2. How open standards help observability
3. Rolling out OpenTelemetry
4. **Adopting observability in practice**

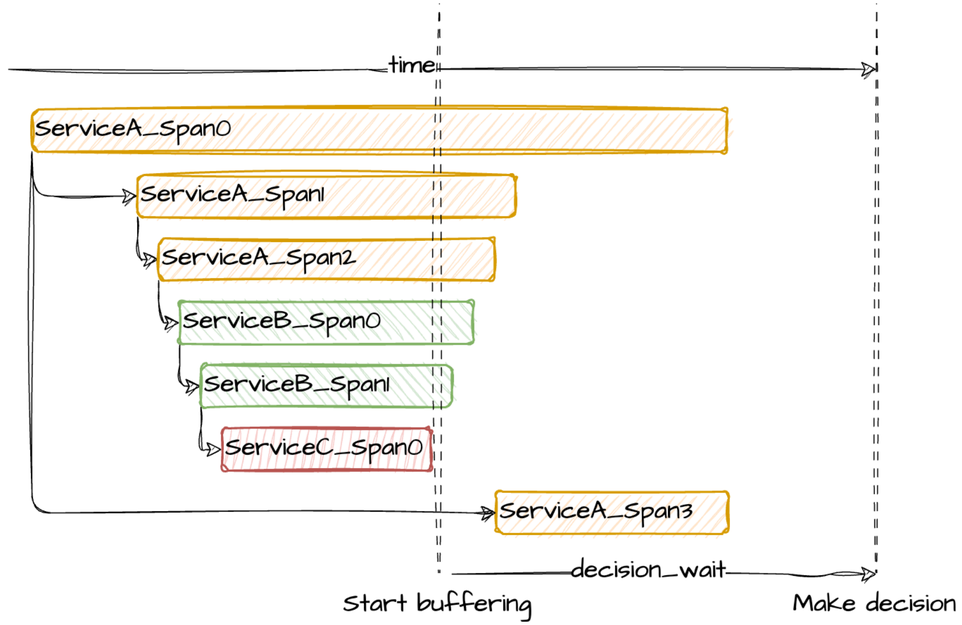
Most data
gathered for
debug purposes is
never used

Keep useful data, discard the rest



Probability sampling

A span is sampled based on its properties or the propagated trace context. Simpler to configure.



Tail based sampling

A span is sampled based on properties of the whole trace (e.g. slow traces or those containing errors). More powerful but requires external components.

How much tracing data do we keep?

4.5%

sampled from all traces

Keeping telemetry data valuable

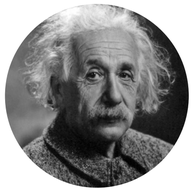
Limiting telemetry production with excessive controls

- Slows down team velocity
- Generates toil for telemetry admins
- Falls out of date soon, defeating its purpose

Visualising cost and encouraging good use of telemetry signals

- Leverage semantic conventions to segment costs to services and namespaces
- Review telemetry along with other service costs
- Reward learning and product health

Failure is success in progress



Albert Einstein

Progress towards success requires action

Foster a learning and improvement culture by:

1. Establishing targets for time-to-detect and time-to-resolve
2. Discussing post-mortems to find areas of improvement
3. Encouraging observability champions to join those discussions
4. Creating a guild/chapter across the company to gather external feedback
5. Running sessions where teams can evaluate telemetry together

- Complex systems require effective observability
- Open standards empower simplification
- OpenTelemetry enables signals to be used efficiently


QCon
LONDON by InfoQ
Thank you
Skyscanner