elastic

# Unifying Observability: The Power of a Common Schema

**FOSDEM(Monitoring & Observability devroom)**
**Feb 2024**

Alex Wert | Engineering @Elastic
Christos Markou | Engineering @Elastic

# About us...

Alex Wert

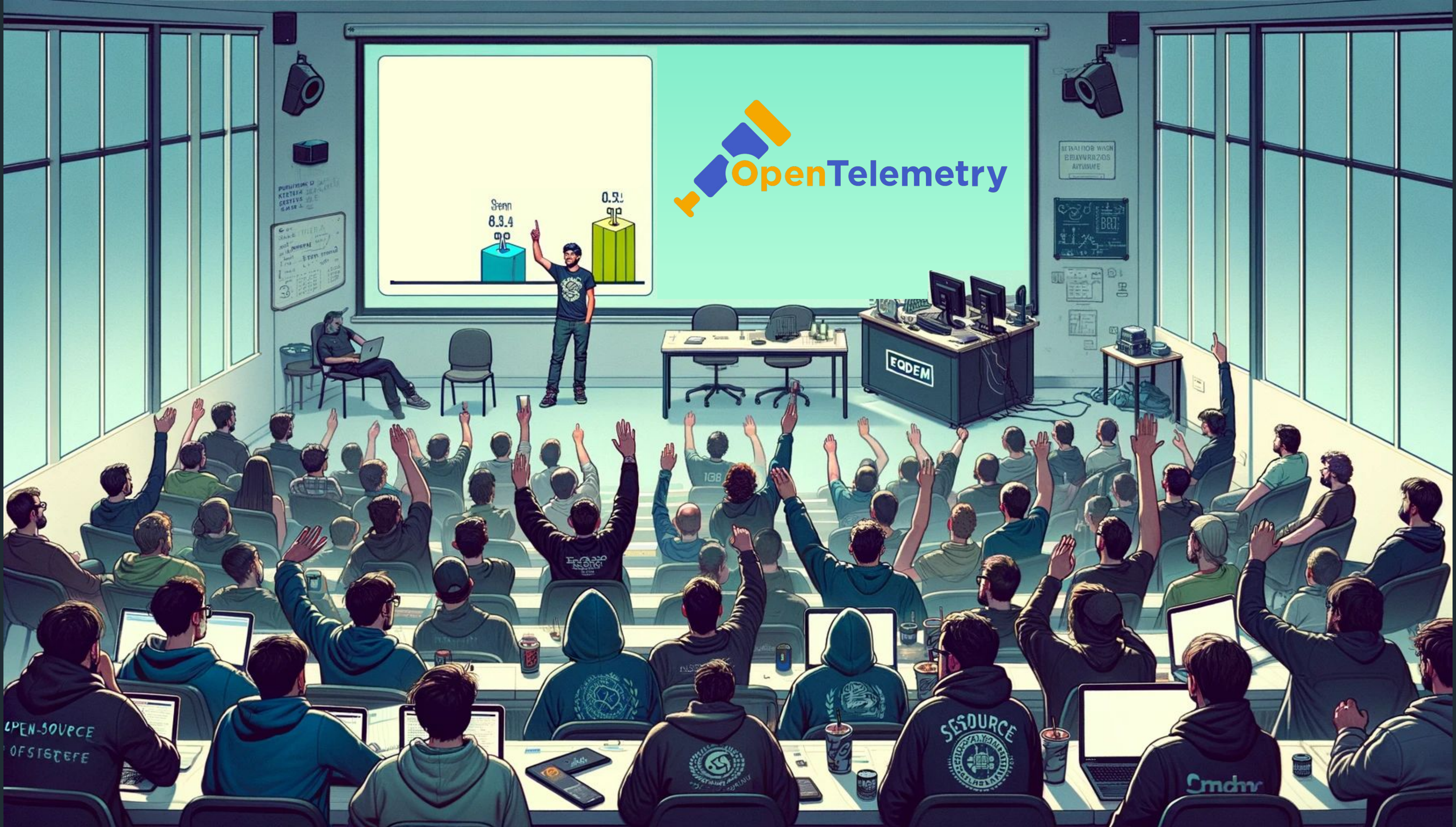
Maintainer @ OTel SemConv

Christos Markou

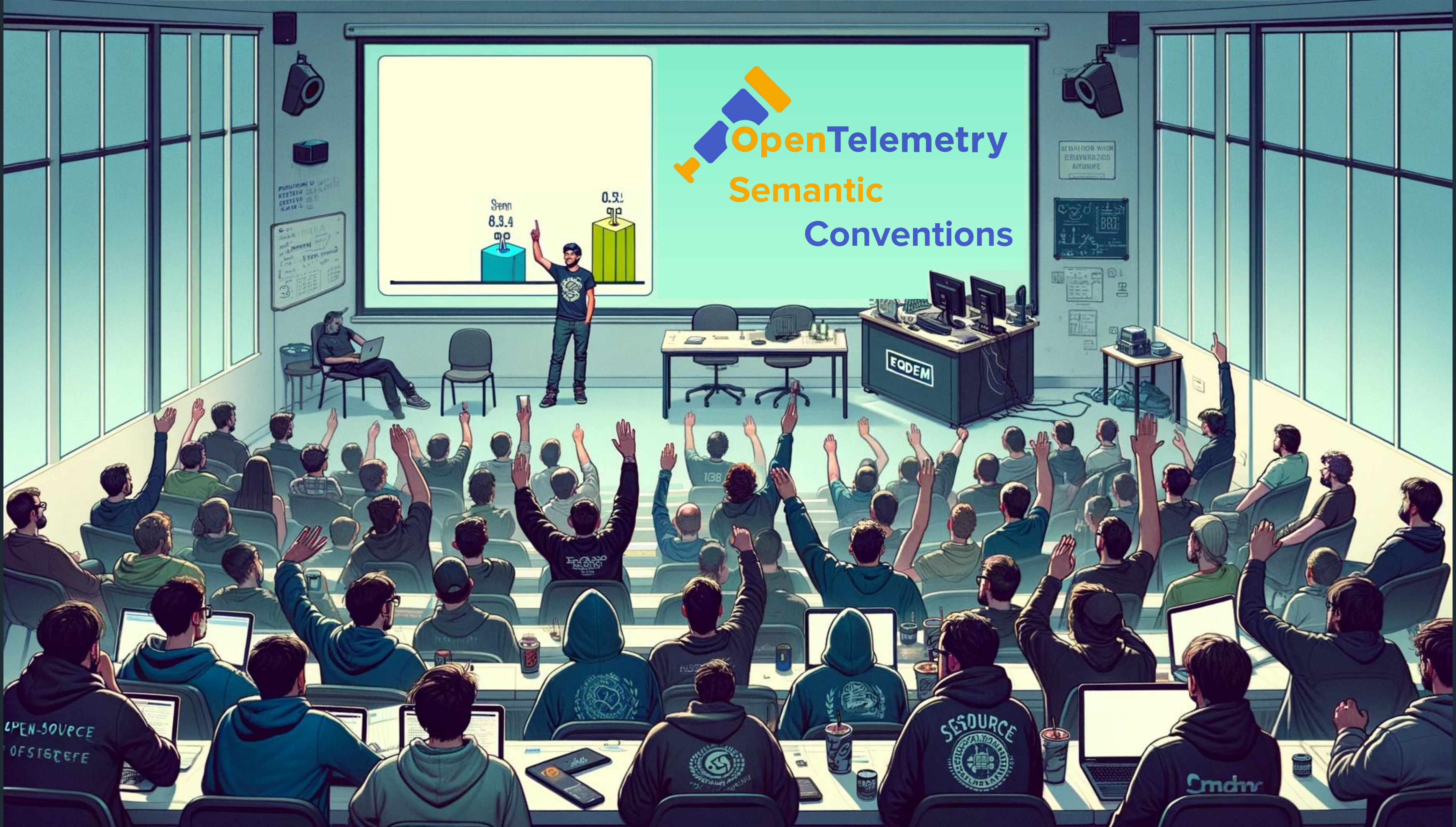
Approver @ OTel SemConv

OpenTelemetry

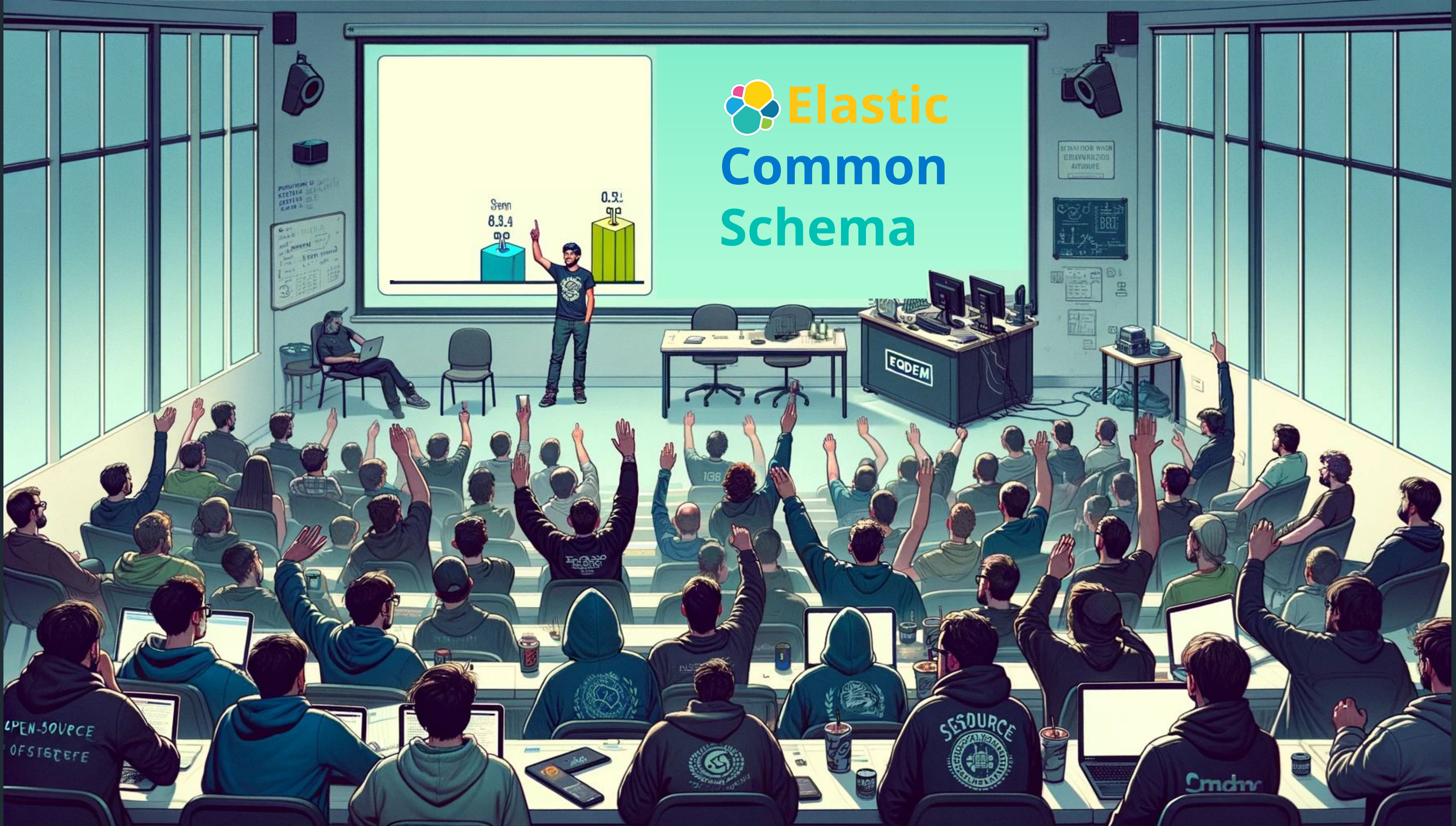
elastic


OpenTelemetry
8.3.4
EQDEM
OPEN-SOURCE
SESDURCE

# OpenTelemetry
# Semantic Conventions

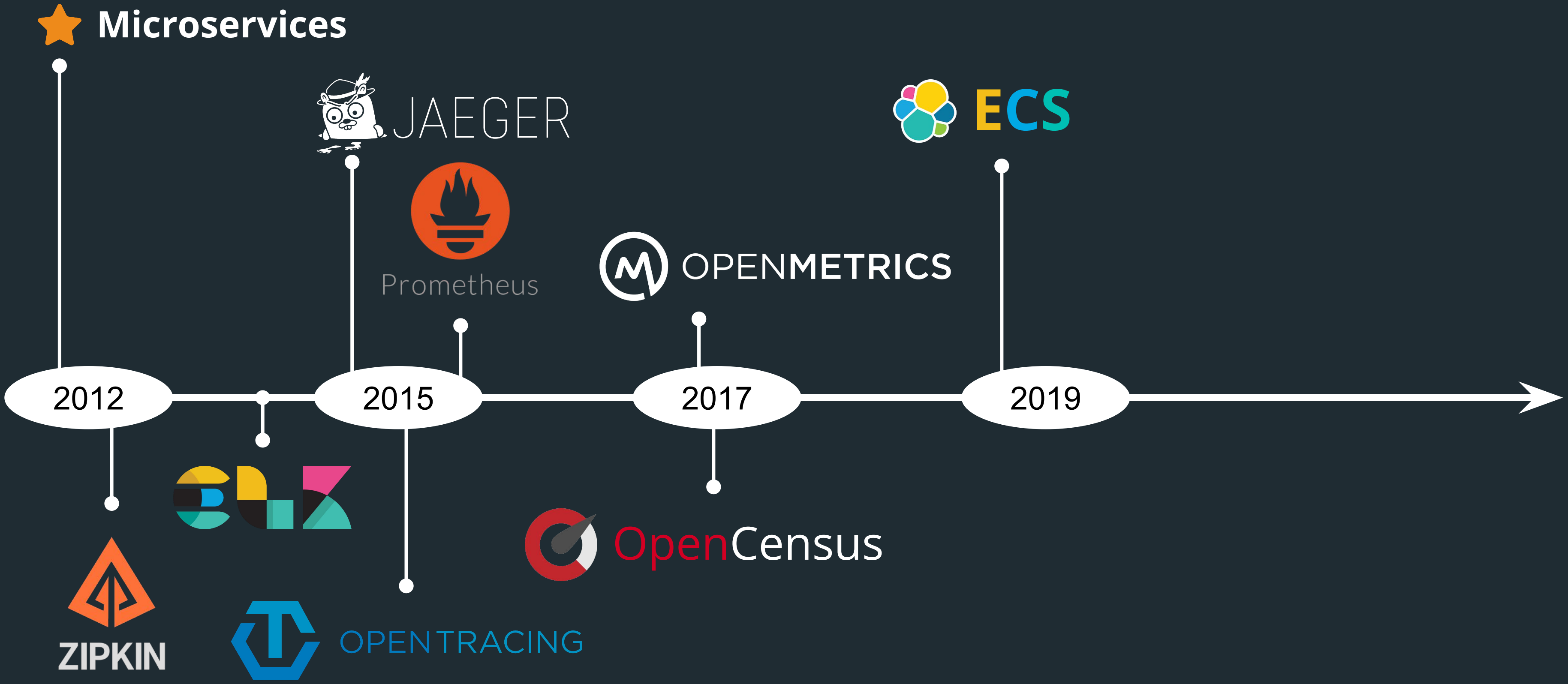


# Elastic Common Schema



# The History of Open Source Tools & Standards for Observability

# O11y - The History of Open Source Tools & Standards



Microservices

2012

ZIPKIN

2015

JAEGER

Prometheus

OPENTRACING

2017

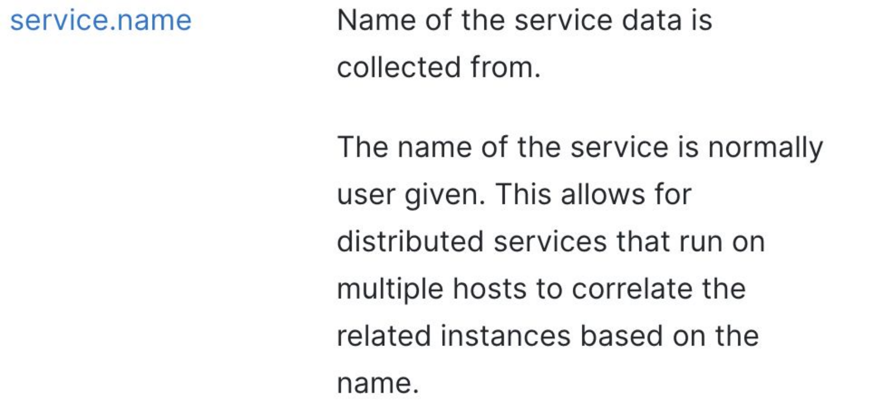
OPENMETRICS

OpenCensus

2019

ECS

# O11y - The History of Open Source Tools & Standards



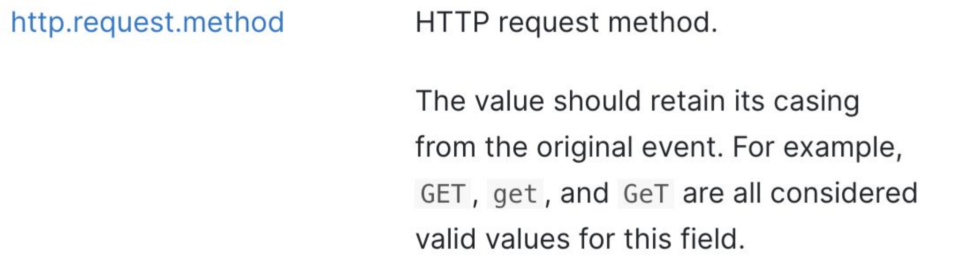
| | |
|---|---|
| service.name | Name of the service data is collected from.<br>The name of the service is normally user given. This allows for distributed services that run on multiple hosts to correlate the related instances based on the name. |

ECS

ENMETRICS

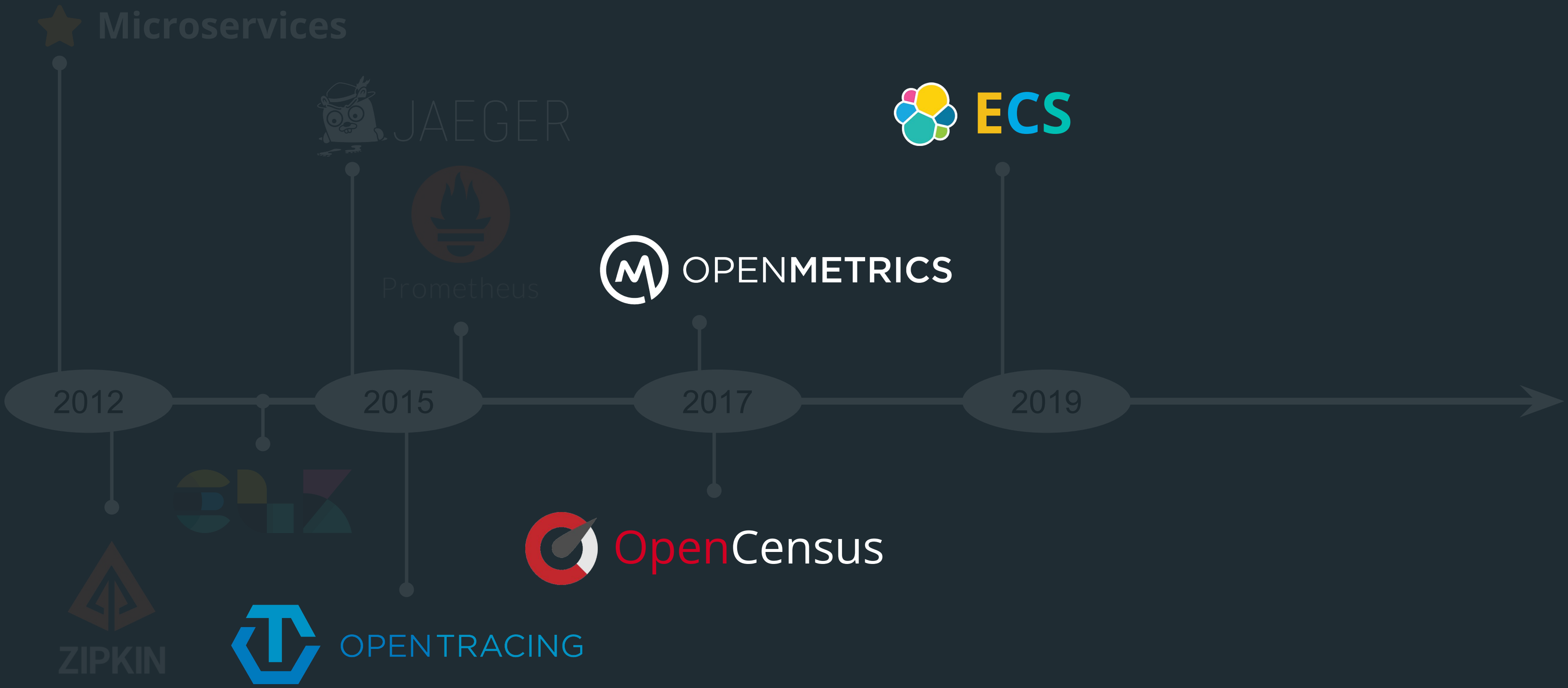
2012 **Elastic Common Schema**

2019



| | |
|---|---|
| http.request.method | HTTP request method.<br>The value should retain its casing from the original event. For example, `GET`, `get`, and `GeT` are all considered valid values for this field. |

# O11y - The History of Open Source Tools & Standards



Microservices

JAEGER

Prometheus

OPENMETRICS

ECS

2012

2015

2017

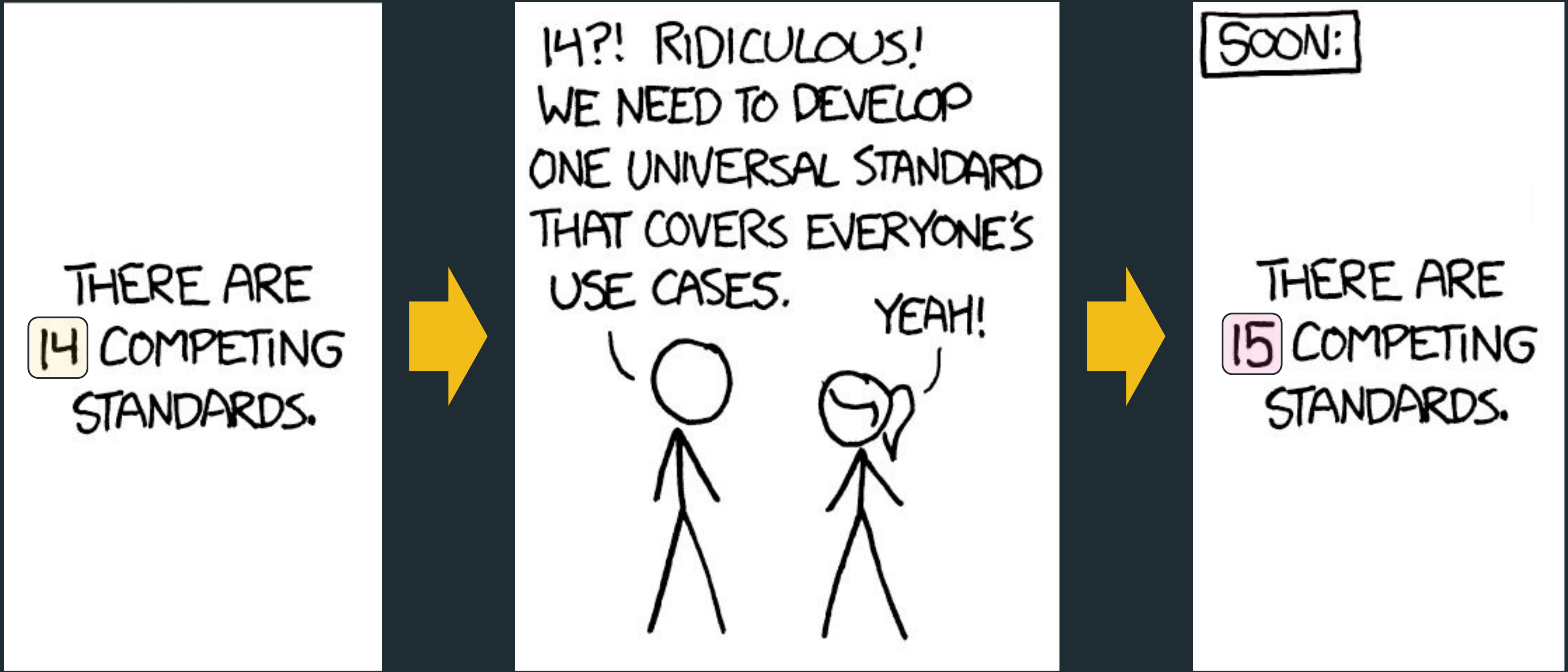
2019

ZIPKIN

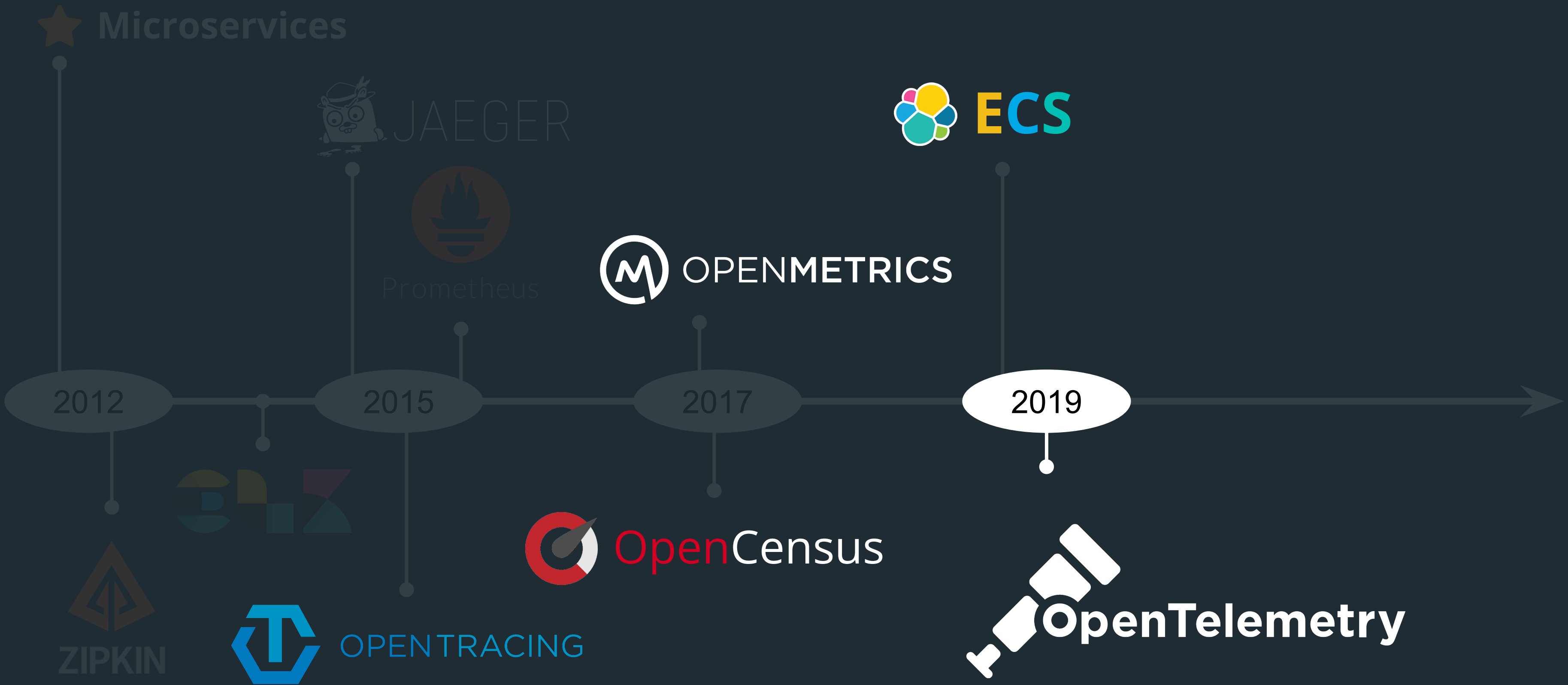
OPENTRACING

OpenCensus

# O11y - The History of Open Source Tools & Standards


THERE ARE 14 COMPETING STANDARDS.
14?! RIDICULOUS! WE NEED TO DEVELOP ONE UNIVERSAL STANDARD THAT COVERS EVERYONE'S USE CASES.
YEAH!
SOON:
THERE ARE 15 COMPETING STANDARDS.

# O11y - The History of Open Source Tools & Standards



Microservices

JAEGER

Prometheus

ECS

OPENMETRICS

2012

2015

2017

2019

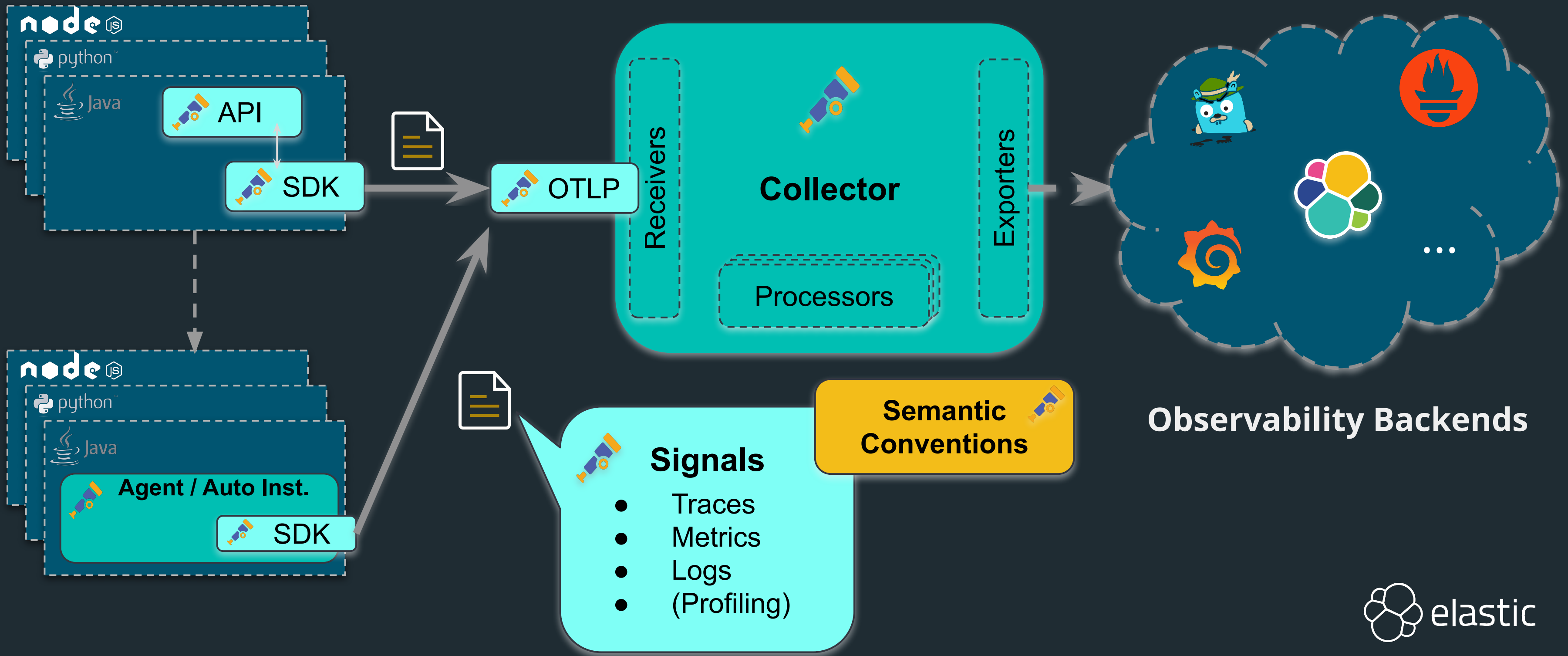
ZIPKIN

OPENTRACING

OpenCensus

OpenTelemetry

# OpenTelemetry



node

python

Java

API

SDK

Agent / Auto Inst.

SDK

OTLP

Receivers

Collector

Processors

Exporters

Signals

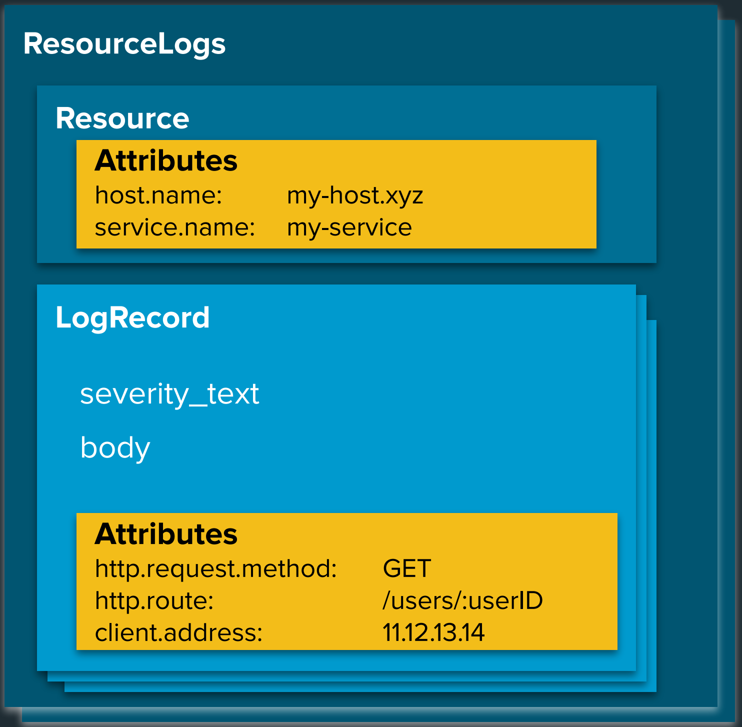
- Traces
- Metrics
- Logs
- (Profiling)

Semantic Conventions

Observability Backends

elastic

# OTLP - Signals - Semantic Conventions



ResourceLogs

Resource

**Attributes**

host.name: my-host.xyz
service.name: my-service

LogRecord

severity_text

body

**Attributes**

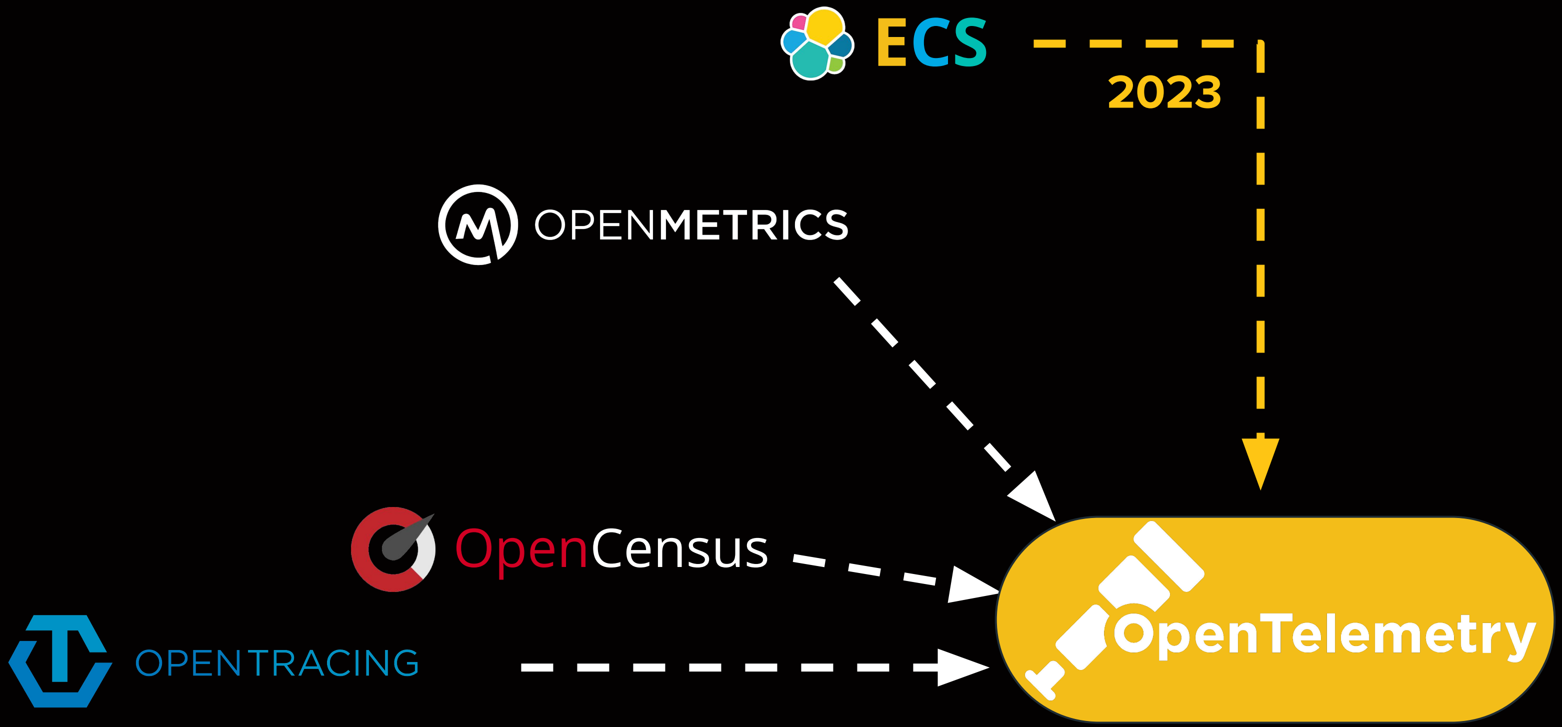
http.request.method: GET
http.route: /users/:userID
client.address: 11.12.13.14

(simplified)

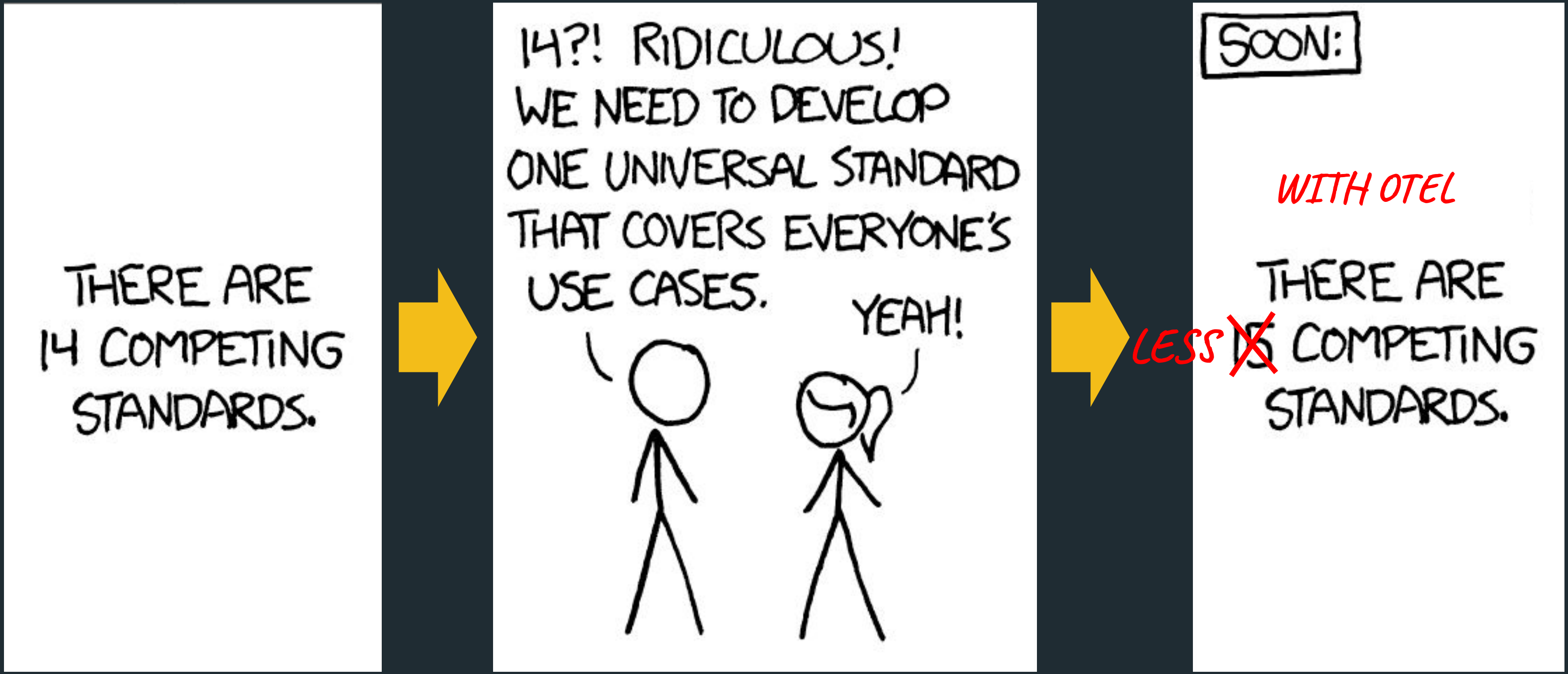
🟦 **OTLP data format**

🟨 **Semantic Conventions**

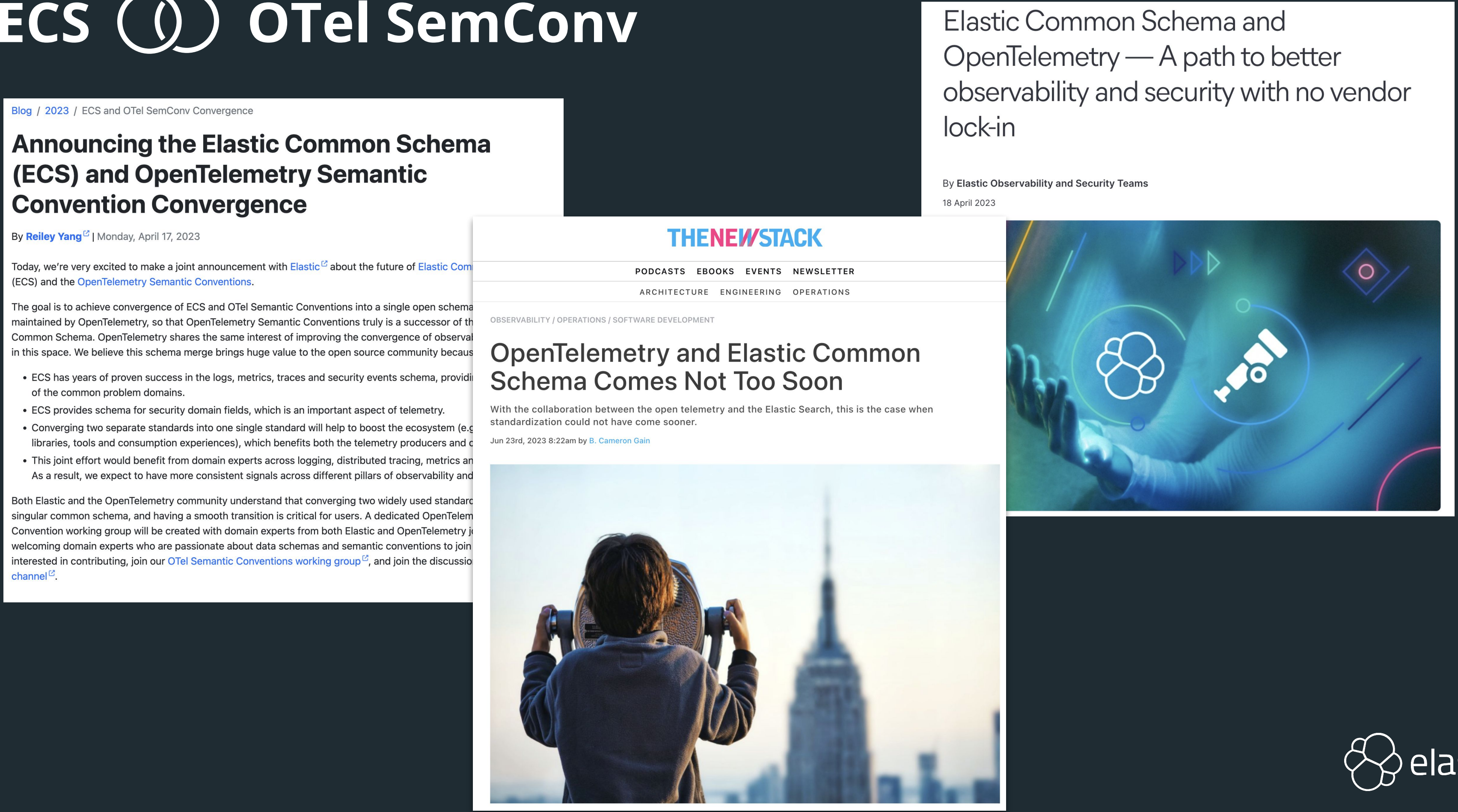
# O11y - The History of Open Source Tools & Standards



# O11y - The History of Open Source Tools & Standards


THERE ARE 14 COMPETING STANDARDS.
14?! RIDICULOUS! WE NEED TO DEVELOP ONE UNIVERSAL STANDARD THAT COVERS EVERYONE'S USE CASES.
YEAH!
SOON:
WITH OTEL
THERE ARE LESS ~~15~~ COMPETING STANDARDS.

# ECS OTel SemConv



Blog / 2023 / ECS and OTel SemConv Convergence

## Announcing the Elastic Common Schema (ECS) and OpenTelemetry Semantic Convention Convergence

By Reiley Yang | Monday, April 17, 2023

Today, we're very excited to make a joint announcement with Elastic about the future of Elastic Com (ECS) and the OpenTelemetry Semantic Conventions.

The goal is to achieve convergence of ECS and OTel Semantic Conventions into a single open schema maintained by OpenTelemetry, so that OpenTelemetry Semantic Conventions truly is a successor of th Common Schema. OpenTelemetry shares the same interest of improving the convergence of observal in this space. We believe this schema merge brings huge value to the open source community becaus

- ECS has years of proven success in the logs, metrics, traces and security events schema, providi of the common problem domains.
- ECS provides schema for security domain fields, which is an important aspect of telemetry.
- Converging two separate standards into one single standard will help to boost the ecosystem (e.g libraries, tools and consumption experiences), which benefits both the telemetry producers and c
- This joint effort would benefit from domain experts across logging, distributed tracing, metrics an As a result, we expect to have more consistent signals across different pillars of observability and

Both Elastic and the OpenTelemetry community understand that converging two widely used standard singular common schema, and having a smooth transition is critical for users. A dedicated OpenTelem Convention working group will be created with domain experts from both Elastic and OpenTelemetry j welcoming domain experts who are passionate about data schemas and semantic conventions to join interested in contributing, join our OTel Semantic Conventions working group, and join the discussio channel.

THENEWSTACK

PODCASTS EBOOKS EVENTS NEWSLETTER

ARCHITECTURE ENGINEERING OPERATIONS

OBSERVABILITY / OPERATIONS / SOFTWARE DEVELOPMENT

## OpenTelemetry and Elastic Common Schema Comes Not Too Soon

With the collaboration between the open telemetry and the Elastic Search, this is the case when standardization could not have come sooner.

Jun 23rd, 2023 8:22am by B. Cameron Gain

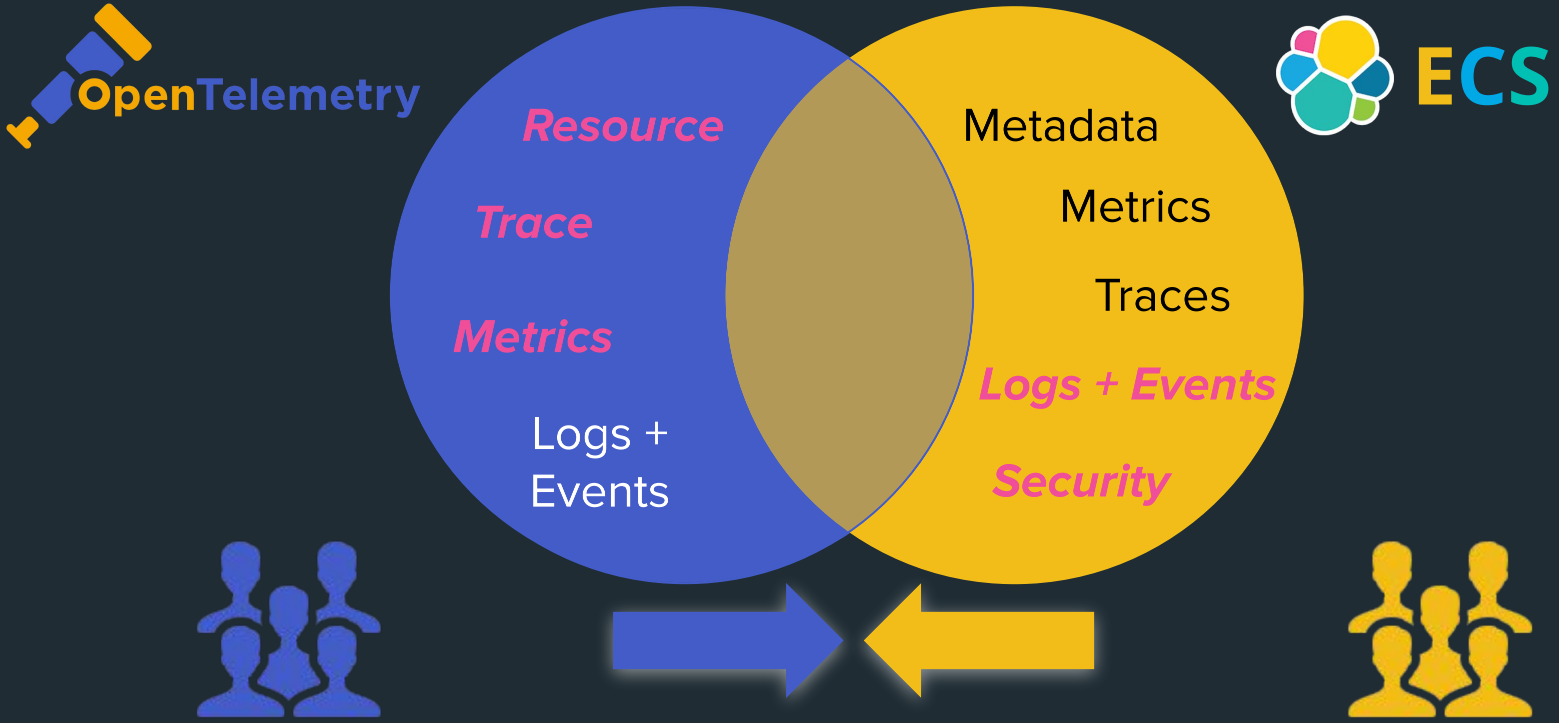
## Elastic Common Schema and OpenTelemetry — A path to better observability and security with no vendor lock-in

By Elastic Observability and Security Teams

18 April 2023

elastic

# Benefits of the Merger

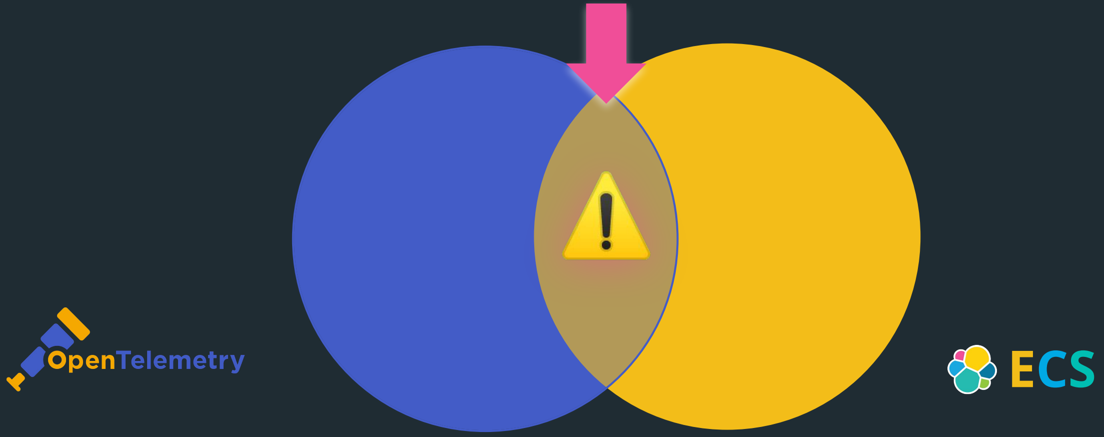


icons8.de

# Challenges …

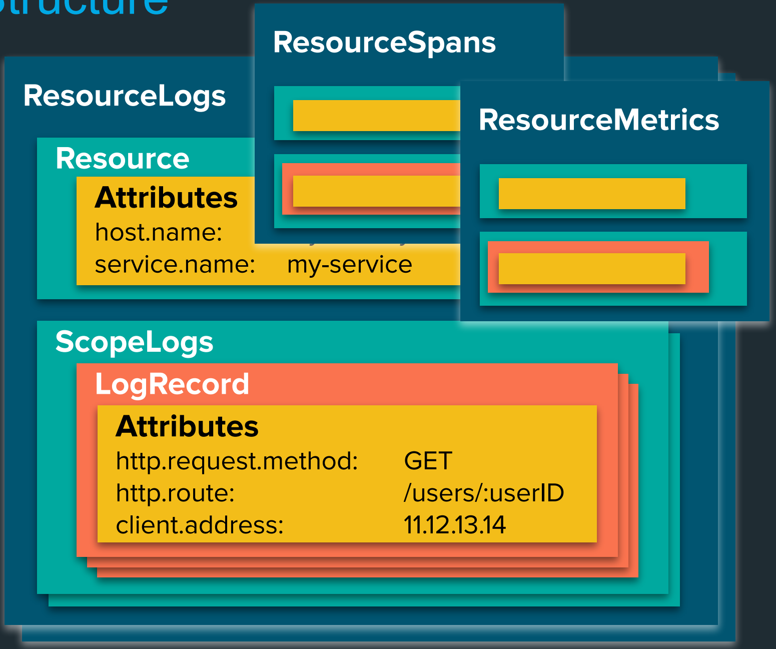
# ECS ⟵⟶ OTel: Challenges & differences

## Breaking Changes / Merging Communities



# ECS ⟵⟶ OTel: Challenges & differences

## Structure



ResourceSpans

ResourceLogs

ResourceMetrics

Resource

Attributes

host.name:

service.name: my-service

ScopeLogs

LogRecord

Attributes

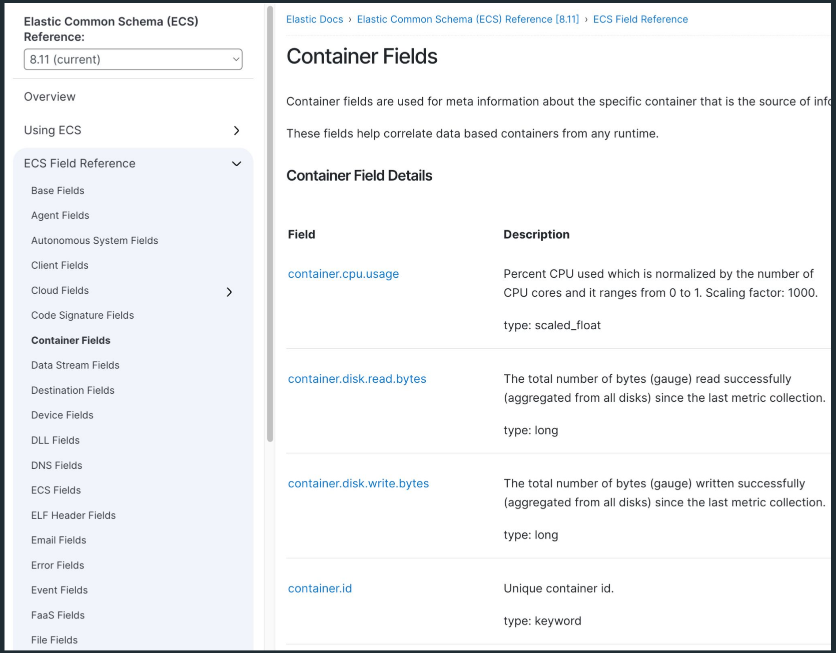
http.request.method: GET

http.route: /users/:userID

client.address: 11.12.13.14

OpenTelemetry

**OTLP + Semantic Conventions**



### Container Fields

Container fields are used for meta information about the specific container that is the source of inf

These fields help correlate data based containers from any runtime.

#### Container Field Details

| Field | Description |
|---|---|
| container.cpu.usage | Percent CPU used which is normalized by the number of CPU cores and it ranges from 0 to 1. Scaling factor: 1000.<br>type: scaled_float |
| container.disk.read.bytes | The total number of bytes (gauge) read successfully (aggregated from all disks) since the last metric collection.<br>type: long |
| container.disk.write.bytes | The total number of bytes (gauge) written successfully (aggregated from all disks) since the last metric collection.<br>type: long |
| container.id | Unique container id.<br>type: keyword |

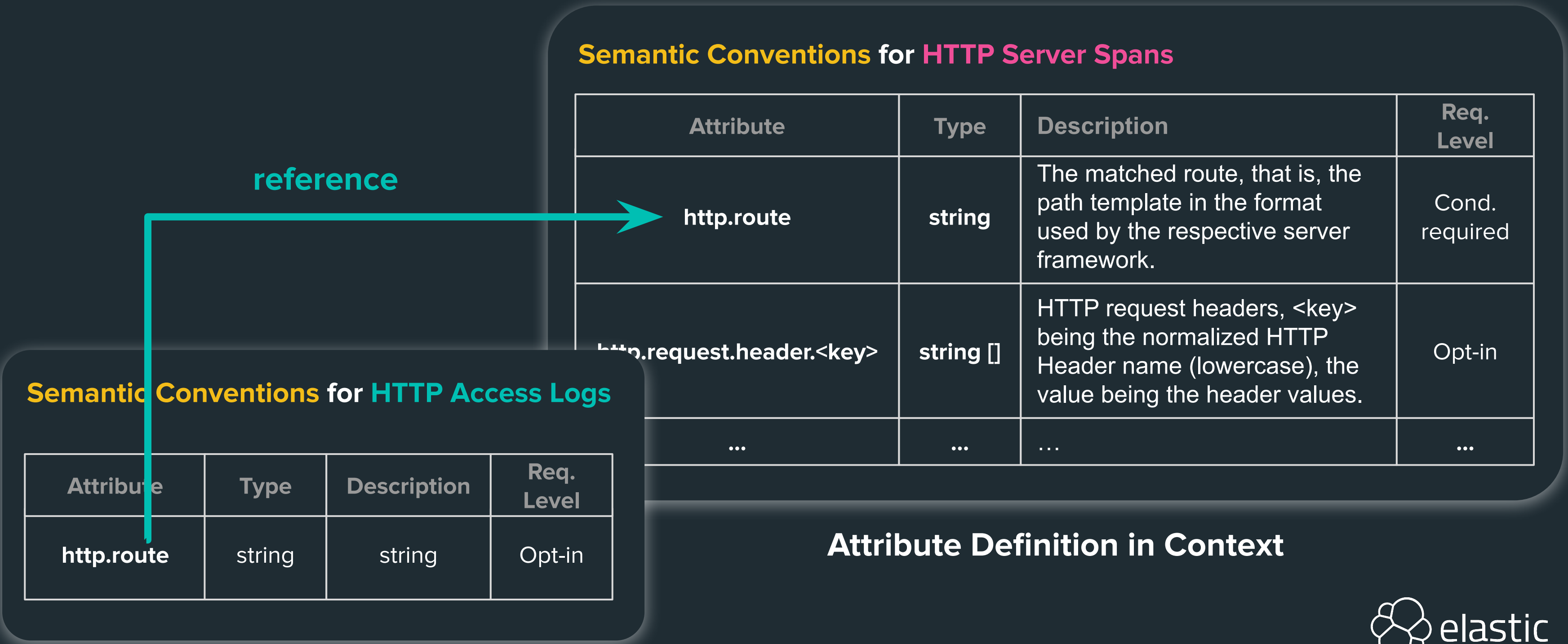
ECS

**Plain field definition**

elastic

# ECS ⟷ OTel: Challenges & differences

## Attribute Definition in Context



reference

**Semantic Conventions for HTTP Access Logs**

| Attribute | Type | Description | Req. Level |
|---|---|---|---|
| **http.route** | string | string | Opt-in |

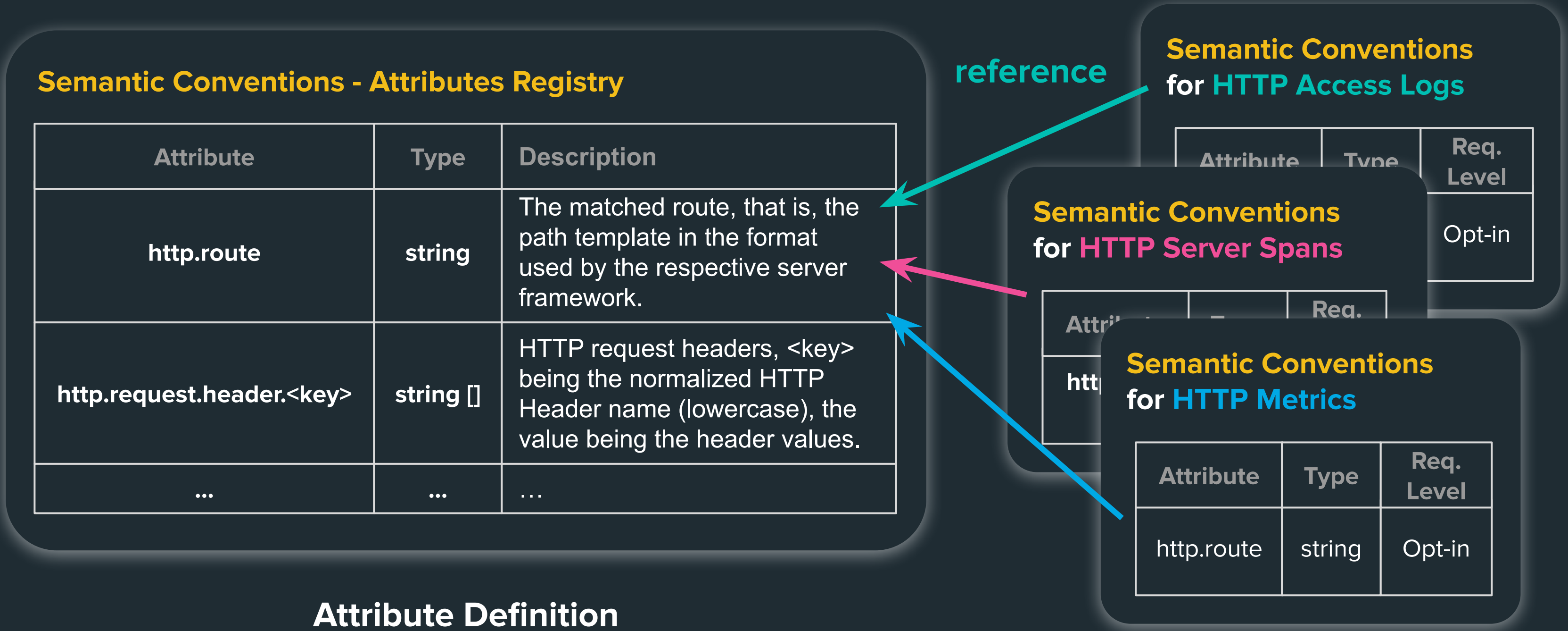
**Semantic Conventions for HTTP Server Spans**

| Attribute | Type | Description | Req. Level |
|---|---|---|---|
| **http.route** | **string** | The matched route, that is, the path template in the format used by the respective server framework. | Cond. required |
| **http.request.header.<key>** | **string []** | HTTP request headers, <key> being the normalized HTTP Header name (lowercase), the value being the header values. | Opt-in |
| ... | ... | … | ... |

**Attribute Definition in Context**

# ECS ⟷ OTel: Challenges & differences

## Attributes Registry



### Semantic Conventions - Attributes Registry

| Attribute | Type | Description |
|---|---|---|
| **http.route** | **string** | The matched route, that is, the path template in the format used by the respective server framework. |
| **http.request.header.<key>** | **string []** | HTTP request headers, <key> being the normalized HTTP Header name (lowercase), the value being the header values. |
| **...** | **...** | … |

reference

### Semantic Conventions for HTTP Access Logs

| Attribute | Type | Req. Level |
|---|---|---|
| | | Opt-in |

### Semantic Conventions for HTTP Server Spans

| Attri | | Req. |
|---|---|---|
| htt | | |

### Semantic Conventions for HTTP Metrics

| Attribute | Type | Req. Level |
|---|---|---|
| http.route | string | Opt-in |

**Attribute Definition**

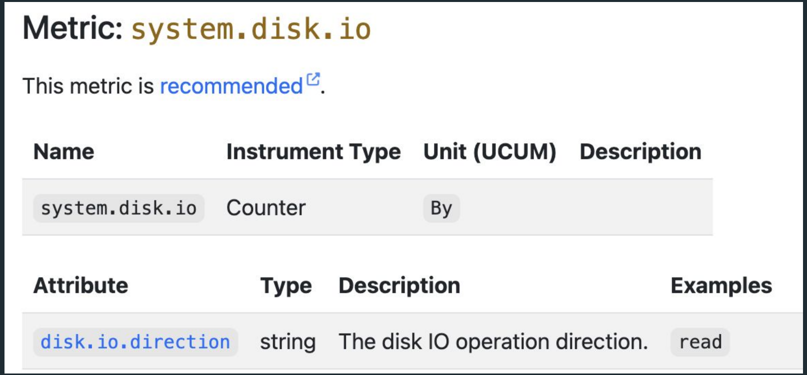
**Attribute Usage in Context**

elastic

# ECS ←→ OTel: Challenges & differences

## Metrics format

OpenTelemetry



Metric: `system.disk.io`

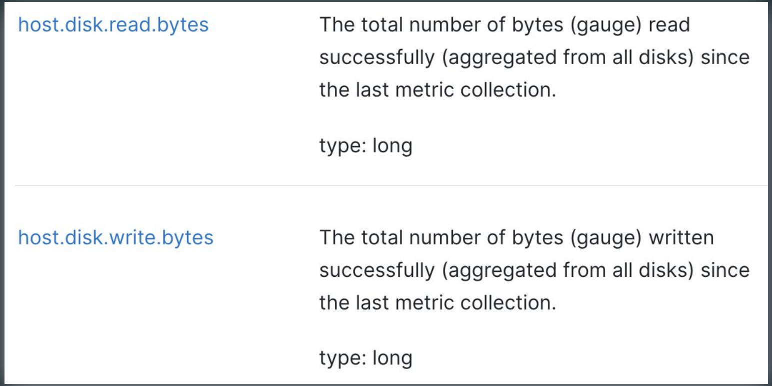
This metric is recommended.

| Name | Instrument Type | Unit (UCUM) | Description |
|---|---|---|---|
| `system.disk.io` | Counter | `By` | |

| Attribute | Type | Description | Examples |
|---|---|---|---|
| `disk.io.direction` | string | The disk IO operation direction. | `read` |

⚠

ECS

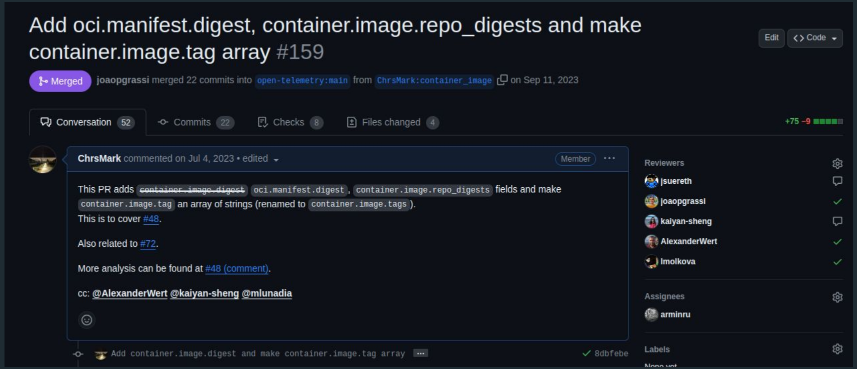


| | |
|---|---|
| host.disk.read.bytes | The total number of bytes (gauge) read successfully (aggregated from all disks) since the last metric collection.<br>type: long |
| host.disk.write.bytes | The total number of bytes (gauge) written successfully (aggregated from all disks) since the last metric collection.<br>type: long |

elastic

NOW
LETS GET SOME WORK DONE
makeameme.org

# Examples



## Add oci.manifest.digest, container.image.repo_digests and make container.image.tag array #159

Merged joaopgrassi merged 22 commits into `open-telemetry:main` from `ChrsMark:container_image` on Sep 11, 2023

**ChrsMark** commented on Jul 4, 2023 • edited

This PR adds ~~`container.image.digest`~~ `oci.manifest.digest`, `container.image.repo_digests` fields and make `container.image.tag` an array of strings (renamed to `container.image.tags`).
This is to cover #48.

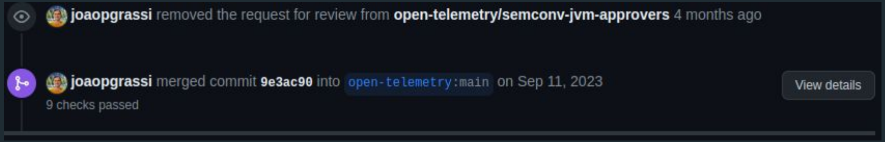
Also related to #72.

More analysis can be found at #48 (comment).

cc: @AlexanderWert @kaiyan-sheng @mlunadia

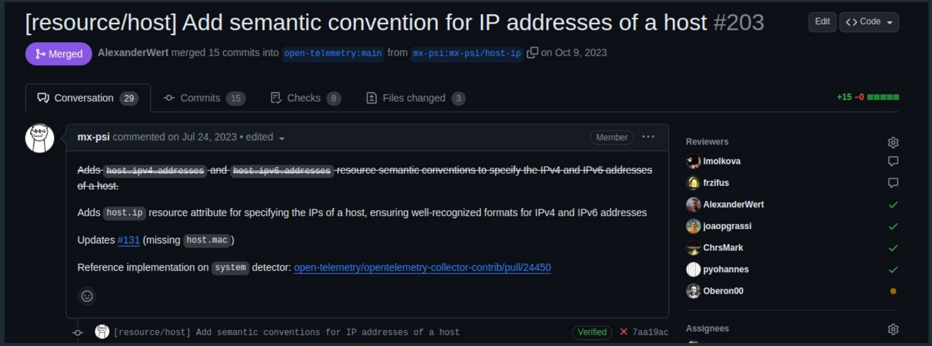
`Add container.image.digest and make container.image.tag array` ✓ 8dbfebe

~20 comments



**joaopgrassi** removed the request for review from **open-telemetry/semconv-jvm-approvers** 4 months ago

**joaopgrassi** merged commit **9e3ac90** into `open-telemetry:main` on Sep 11, 2023
9 checks passed



## [resource/host] Add semantic convention for IP addresses of a host #203

Merged AlexanderWert merged 15 commits into `open-telemetry:main` from `mx-psi:mx-psi/host-ip` on Oct 9, 2023

**mx-psi** commented on Jul 24, 2023 • edited

~~Adds `host.ipv4.addresses` and `host.ipv6.addresses` resource semantic conventions to specify the IPv4 and IPv6 addresses of a host.~~

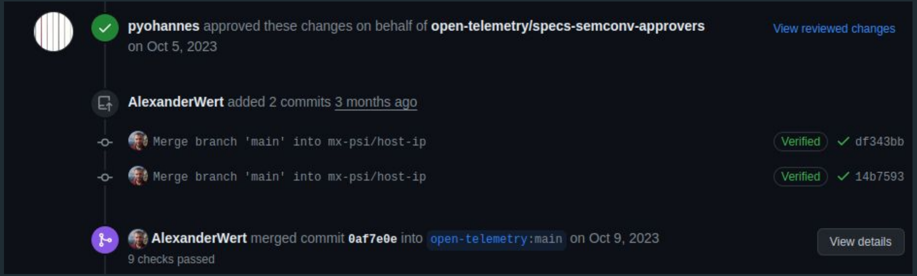
Adds `host.ip` resource attribute for specifying the IPs of a host, ensuring well-recognized formats for IPv4 and IPv6 addresses

Updates #131 (missing `host.mac`)

Reference implementation on `system` detector: open-telemetry/opentelemetry-collector-contrib/pull/24450

`[resource/host] Add semantic conventions for IP addresses of a host` Verified ✗ 7aa19ac

~23 comments



**pyohannes** approved these changes on behalf of **open-telemetry/specs-semconv-approvers** on Oct 5, 2023

**AlexanderWert** added 2 commits 3 months ago

`Merge branch 'main' into mx-psi/host-ip` Verified ✓ df343bb

`Merge branch 'main' into mx-psi/host-ip` Verified ✓ 14b7593

**AlexanderWert** merged commit **0af7e0e** into `open-telemetry:main` on Oct 9, 2023
9 checks passed

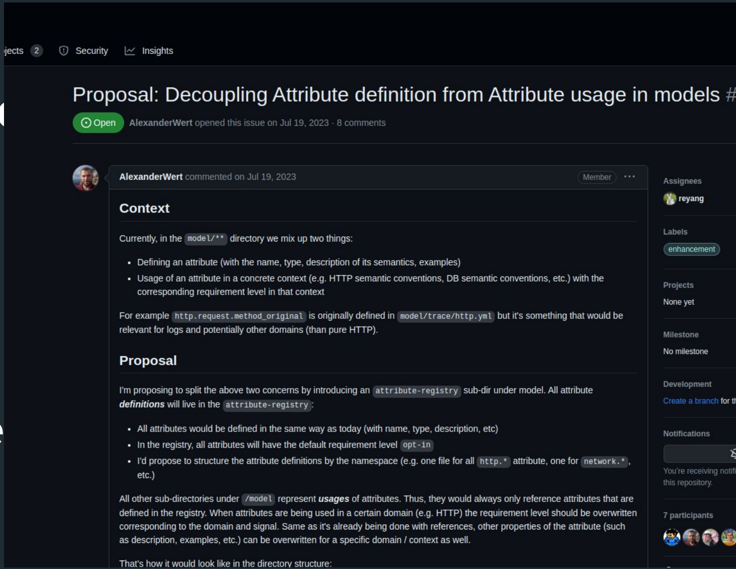
## Evolution of the merger

- system/host metrics: moving towards stability
- process: ““
- container: 60%-> ongoing PR -> 100%
- http, network: ~50%
- databases, mobile: WiP
- cloud: WiP
- k8s: WiP

# Evolution of the Sem Conv project during th

The work is moving forward in a community driven

- joint efforts to improve the tooling
- working on the improvement of the “guideline
- project re-structuring to group by topic
- introduction of attribute’s registry
- field reuse concept for OTel semantic attributes



## Proposal: Decoupling Attribute definition from Attribute usage in models #

Open AlexanderWert opened this issue on Jul 19, 2023 · 8 comments

**AlexanderWert** commented on Jul 19, 2023

### Context

Currently, in the `model/**` directory we mix up two things:

- Defining an attribute (with the name, type, description of its semantics, examples)
- Usage of an attribute in a concrete context (e.g. HTTP semantic conventions, DB semantic conventions, etc.) with the corresponding requirement level in that context

For example `http.request.method_original` is originally defined in `model/trace/http.yml` but it's something that would be relevant for logs and potentially other domains (than pure HTTP).

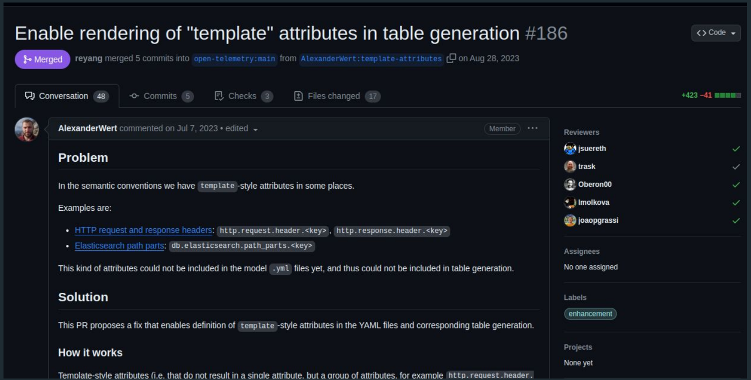
### Proposal

I'm proposing to split the above two concerns by introducing an `attribute-registry` sub-dir under model. All attribute ***definitions*** will live in the `attribute-registry`:

- All attributes would be defined in the same way as today (with name, type, description, etc)
- In the registry, all attributes will have the default requirement level `opt-in`
- I'd propose to structure the attribute definitions by the namespace (e.g. one file for all `http.*` attribute, one for `network.*`, etc.)

All other sub-directories under `/model` represent ***usages*** of attributes. Thus, they would always only reference attributes that are defined in the registry. When attributes are being used in a certain domain (e.g. HTTP) the requirement level should be overwritten corresponding to the domain and signal. Same as it's already being done with references, other properties of the attribute (such as description, examples, etc.) can be overwritten for a specific domain / context as well.

That's how it would look like in the directory structure:



## Enable rendering of "template" attributes in table generation #186

Merged reyang merged 5 commits into `open-telemetry:main` from `AlexanderWert:template-attributes` on Aug 28, 2023

**AlexanderWert** commented on Jul 7, 2023 • edited

### Problem

In the semantic conventions we have `template`-style attributes in some places.

Examples are:

- HTTP request and response headers: `http.request.header.<key>`, `http.response.header.<key>`
- Elasticsearch path parts: `db.elasticsearch.path_parts.<key>`

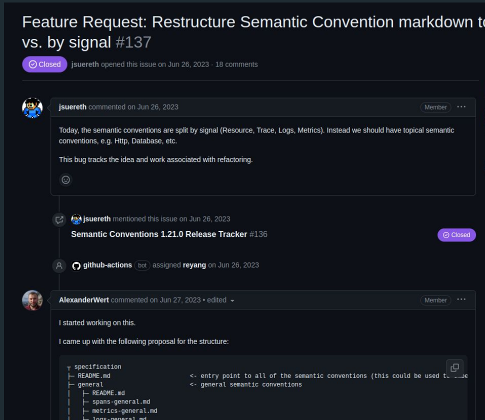
This kind of attributes could not be included in the model `.yml` files yet, and thus could not be included in table generation.

### Solution

This PR proposes a fix that enables definition of `template`-style attributes in the YAML files and corresponding table generation.

### How it works

Template-style attributes (i.e. that do not result in a single attribute, but a group of attributes, for example `http.request.header.`



## Feature Request: Restructure Semantic Convention markdown to be topical vs. by signal #137

Closed jsuereth opened this issue on Jun 26, 2023 · 18 comments

**jsuereth** commented on Jun 26, 2023

Today, the semantic conventions are split by signal (Resource, Trace, Logs, Metrics). Instead we should have topical semantic conventions, e.g. Http, Database, etc.

This bug tracks the idea and work associated with refactoring.

jsuereth mentioned this issue on Jun 26, 2023

**Semantic Conventions 1.21.0 Release Tracker** #136

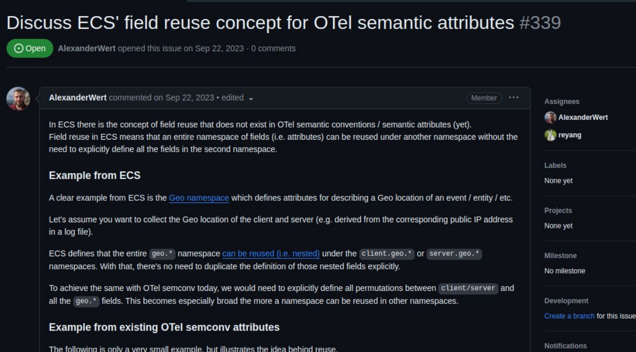
github-actions bot assigned **reyang** on Jun 26, 2023

**AlexanderWert** commented on Jun 27, 2023 • edited

I started working on this.

I came up with the following proposal for the structure:

```
┬ specification
├─ README.md                    <- entry point to all of the semantic conventions (this could be used t
├─ general                      <- general semantic conventions
│   ├─ README.md
│   ├─ spans-general.md
│   ├─ metrics-general.md
│   └─ logs-general.md
```



## Discuss ECS' field reuse concept for OTel semantic attributes #339

Open AlexanderWert opened this issue on Sep 22, 2023 · 0 comments

**AlexanderWert** commented on Sep 22, 2023 • edited

In ECS there is the concept of field reuse that does not exist in OTel semantic conventions / semantic attributes (yet).
Field reuse in ECS means that an entire namespace of fields (i.e. attributes) can be reused under another namespace without the need to explicitly define all the fields in the second namespace.

### Example from ECS

A clear example from ECS is the Geo namespace which defines attributes for describing a Geo location of an event / entity / etc.

Let's assume you want to collect the Geo location of the client and server (e.g. derived from the corresponding public IP address in a log file).

ECS defines that the entire `geo.*` namespace can be reused (i.e. nested) under the `client.geo.*` or `server.geo.*` namespaces. With that, there's no need to duplicate the definition of those nested fields explicitly.

To achieve the same with OTel semconv today, we would need to explicitly define all permutations between `client/server` and all the `geo.*` fields. This becomes especially broad the more a namespace can be reused in other namespaces.

### Example from existing OTel semconv attributes

The following is only a very small example, but illustrates the idea behind reuse.

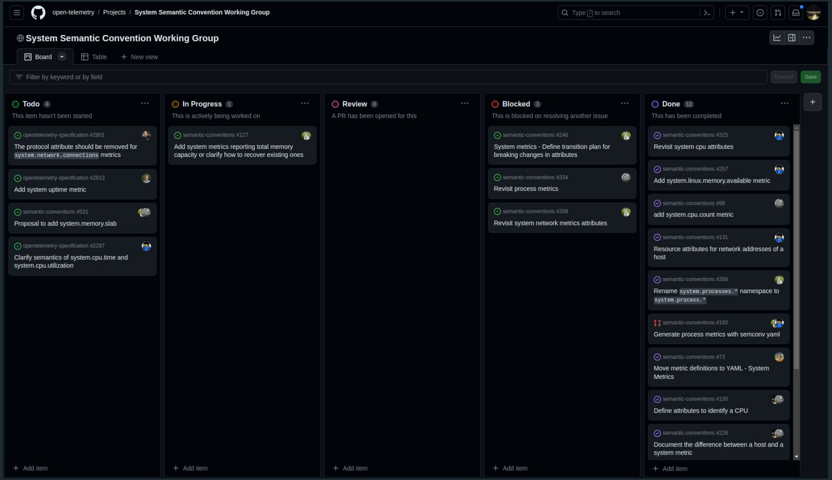
# How the community is organised around this and how the merger is moving forward

Working groups with domain experts focusing on the stability of the area

a) stabilizing the semantic conventions
b) tuning OTel implementations

# How the community is organised around this and how the merger is moving forward

system metrics WG: board
db WG: project
security semconv WG: proposal
mobile area: approvers-group
containers/k8s: approvers-group


open-telemetry / Projects / System Semantic Convention Working Group
Type / to search
Board
Table
New view
Discard
Save
Add item

System Semantic Convention Working Group

Filter by keyword or by field

**Todo** 4
This item hasn't been started

- opentelemetry-specification #2801
  The protocol attribute should be removed for `system.network.connections` metrics
- opentelemetry-specification #2813
  Add system uptime metric
- semantic-conventions #531
  Proposal to add system.memory.slab
- opentelemetry-specification #2287
  Clarify semantics of system.cpu.time and system.cpu.utilization

**In Progress** 1
This is actively being worked on

- semantic-conventions #127
  Add system metrics reporting total memory capacity or clarify how to recover existing ones

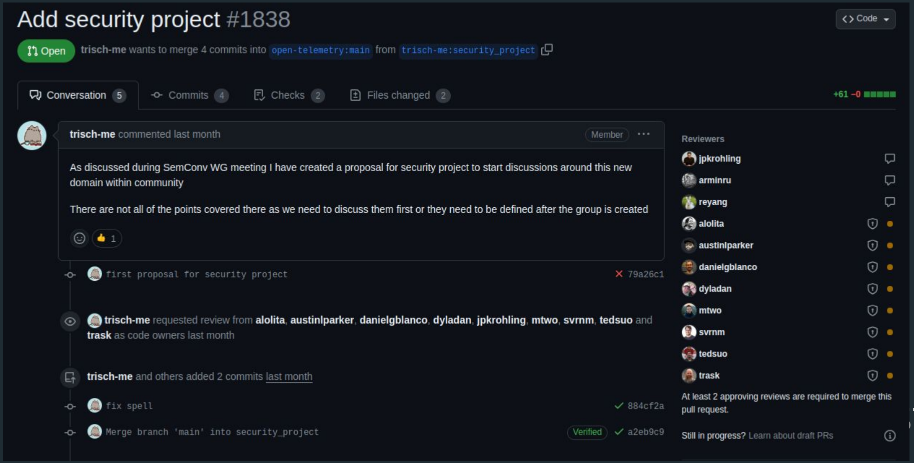
**Review** 0
A PR has been opened for this

**Blocked** 3
This is blocked on resolving another issue

- semantic-conventions #246
  System metrics - Define transition plan for breaking changes in attributes
- semantic-conventions #334
  Revisit process metrics
- semantic-conventions #308
  Revisit system network metrics attributes

**Done** 12
This has been completed

- semantic-conventions #325
  Revisit system cpu attributes
- semantic-conventions #257
  Add system.linux.memory.available metric
- semantic-conventions #98
  add system.cpu.count metric
- semantic-conventions #131
  Resource attributes for network addresses of a host
- semantic-conventions #266
  Rename `system.processes.*` namespace to `system.process.*`
- semantic-conventions #160
  Generate process metrics with semconv yaml
- semantic-conventions #73
  Move metric definitions to YAML - System Metrics
- semantic-conventions #130
  Define attributes to identify a CPU
- semantic-conventions #226
  Document the difference between a host and a system metric



Add security project #1838

trisch-me wants to merge 4 commits into `open-telemetry:main` from `trisch-me:security_project`

**trisch-me** commented last month

As discussed during SemConv WG meeting I have created a proposal for security project to start discussions around this new domain within community

There are not all of the points covered there as we need to discuss them first or they need to be defined after the group is created

👍 1

`first proposal for security project` ✗ 79a26c1

**trisch-me** requested review from **alolita**, **austinlparker**, **danielgblanco**, **dyladan**, **jpkrohling**, **mtwo**, **svrnm**, **tedsuo** and **trask** as code owners last month

**trisch-me** and others added 2 commits last month

`fix spell` ✓ 884cf2a

`Merge branch 'main' into security_project` Verified ✓ a2eb9c9

Reviewers

- jpkrohling
- arminru
- reyang
- alolita
- austinlparker
- danielgblanco
- dyladan
- mtwo
- svrnm
- tedsuo
- trask

At least 2 approving reviews are required to merge this pull request.

Still in progress? Learn about draft PRs

# How the merger takes place in reality

1) Cross check of ECS and OTel SemConv
2) Check what the implementation of OTel collector and language SDKs follow
3) Proposal of merged fields
4) Open discussion in the community
    a) Measure breaking changes in both sides, if any.
    b) review cycles
5) Conclude and merge
6) Handle breaking changes

# Summary

Merger is happening – contributions more than welcome :)

Community driven work

Goal: make OTel SemConv the one, unique and straightforward standard for O11y and Security

# Where to find us / Questions

CNCF Slack

@AlexanderWert

@ChrsMark

**Project Meetings**

Monday 5:00 CET (SemConv working group)
Tuesday 5:00 CET (Specification SIG)
Thursday 5:30 CET (System metrics WG)