

From 0 to <developer>: my one year serendipity journey with OpenTelemetry

Diana Todea - Technical Advocate @Aircall

9th of July 2025



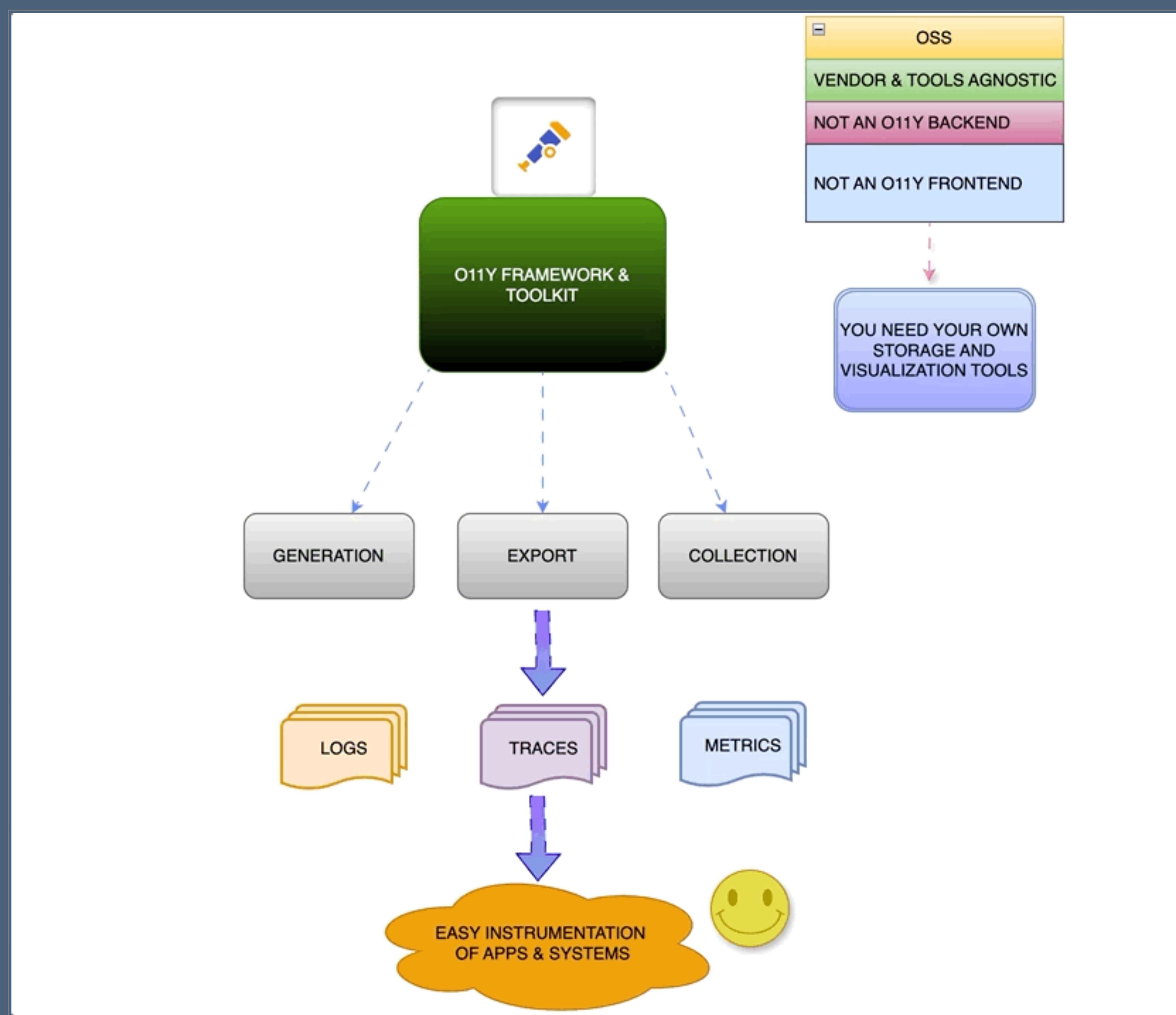


Diana Todea

Developer Advocate, Observability
SME, OTEL contributor | Keynote & In...



[github@didiviking/Conferences_Talks](https://github.com/didiviking/Conferences_Talks)



Journeys start with a grain of curiosity

Linux Foundation community call for observability SMEs

Creating first ever certification and exam for OpenTelemetry

Great first contribution to the OTel project as an exam developer

CERTIFICATION

OpenTelemetry Certified Associate (OTCA)

As cloud native systems grow more complex, the demand for professionals who can leverage telemetry data is growing rapidly. Open new career paths – prove your expertise in OpenTelemetry – the industry standard for tracing, metrics & logs.

OTCA includes:

- ✓ 12-months to schedule & take the exam
- ✓ Two exam attempts

Now Available! You can bundle OTCA with an [annual THRIVE subscription](#) to get access to more than 100 educational products and [SkillCreds](#) for only \$495!



Continue with practice

- ➡ Create a proof of concept: explain the OTel framework to an entire organization
- ➡ Validating OTel in medium sized companies with a fragmented observability turns into a struggle
- ➡ PROs: vendor neutrality, standard for OSS only
- ➡ CONS: docs, manual instrumentation achievable with great effort, count for overhead, not satisfied with visualization frontend (e.g. dashboards), questionable support in production
- ➡ OTel considers profiling and instrumenting as a different scope

I played with OTEL and I liked it



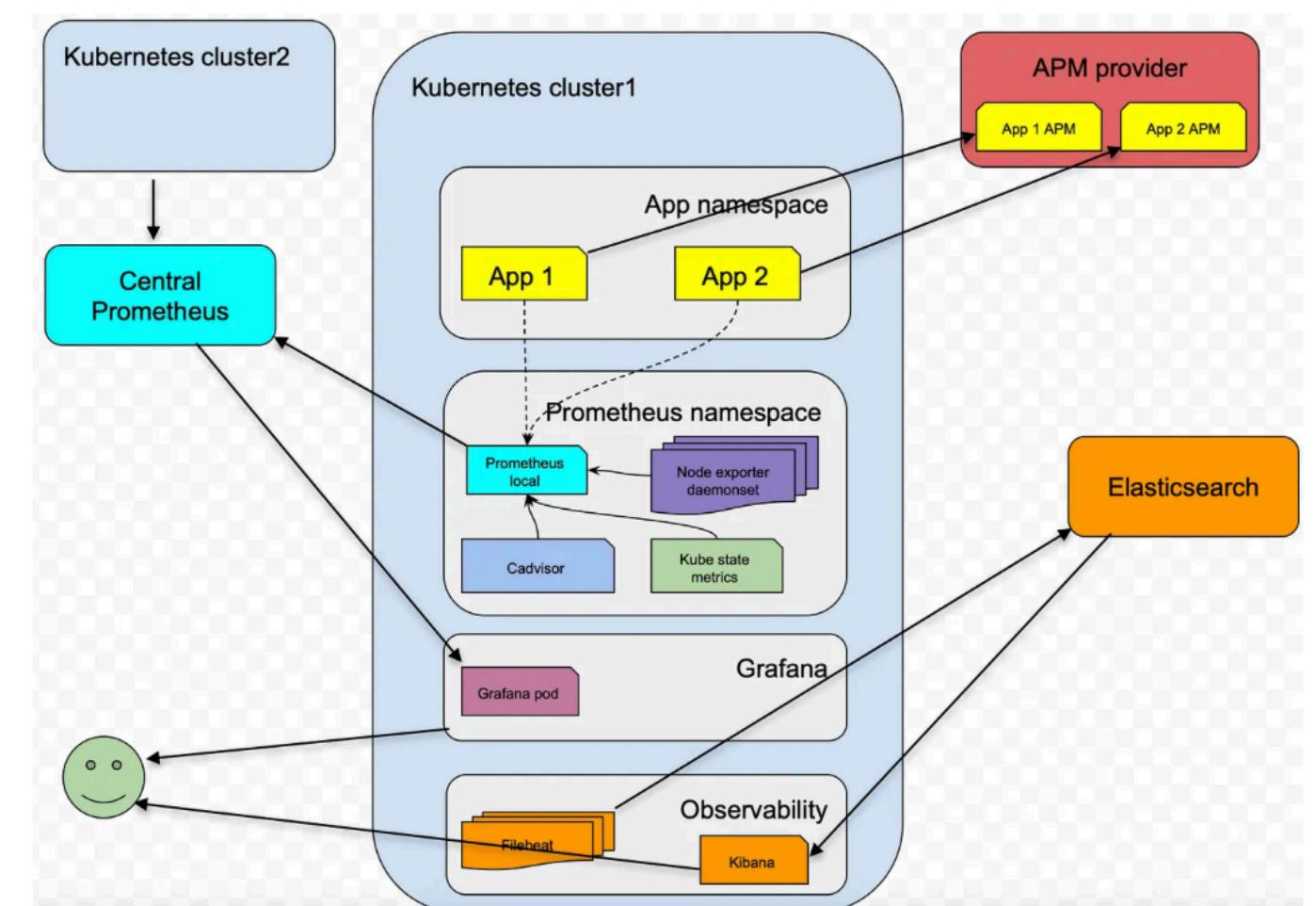
Diana Todea
4 min read · Jan 7, 2025

25 1 ⌂ ⌄ ⌅ ...

As mentioned in my previous [article](#), in 2024 I had the opportunity to work on an observability project where I wrote a proof of concept on [OpenTelemetry](#). As an SRE I saw the window to experiment with one of [the most popular](#) CNCF projects on distributed tracing and provide an open source alternative to monitor the performance of my client's applications. I had the chance to give a more in depth talk on this topic at [KCD Accra Ghana](#) as well as at [OSMC 2024](#). You can access my presentations directly on my [Github repo](#).

Architecture

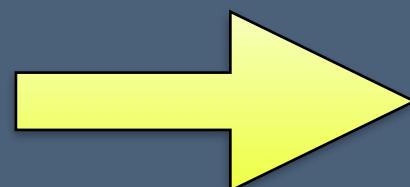
This particular observability architecture was mainly composed of [Prometheus metrics](#), [Victoria Metrics](#), [Grafana](#) for visualizations and alerting, [ELK](#) for logging and a commercial APM vendor. A fragmented observability architecture leads to a confusing developer and user experience, working in observability for almost 5 years now it's easy to say that most people prefer having a unified monitoring pane of glass, which is easier to onboard and manage.



Observability architecture explaining how the K8s clusters ship metrics, traces and logs

Provide your feedback to the community

<https://github.com/open-telemetry/opentelemetry-specification/issues/4255>



<https://github.com/open-telemetry/community/issues/2406>

#otel-ebpf-instrumentation

github.com/open-telemetry/opentelemetry-specification/issues/4255#issuecomment-2410636107

open-telemetry / opentelemetry-specification

Issues 587 Pull requests 8 Discussions Actions Projects 1 Security Insights

Truly auto instrumentation #4255

Open

 ostrolucky opened on Oct 11, 2024 · edited by ostrolucky

[Edits](#)

What are you trying to achieve?

Having a way to find unexpected bottlenecks

Additional context.

My biggest problem with Opentelemetry is that the way it seem to work is you have to specify which functions to instrument. There are `opentelemetry-auto-*` packages, however all those are is predefined lists of such functions. No matter if you use `opentelemetry-auto-*` packages or not, problem remains: You have to specify exact functions to instrument, which is not how commercial solutions do it.

Why is this a problem?

Since you have to pick functions to instrument, once bottleneck happens in a function you didn't expect would be a bottleneck, you will have no clue from your spans where is the bottleneck, since you don't instrument it. All you are left with at that point is to check all function calls from the place that is identified by parent span, add some or all of them to your instrumentation and redeploy.

To make opentelemetry more useful, we should have a way to instrument on something else than some static list of functions, for example random sampling instrumentation of all function calls, or any function call which exceed time treshold. That would be closer to the way what people are used to and I would argue more useful than what we have now, which leaves you quite stuck once you run into some unexpected slowdown and struggling to find the root cause.

I would also suggest to rename current `opentelemetry-auto` concept to `opentelemetry-preset`, as auto here is very misleading if what those packages mostly do is to predefine list of functions to trace

github.com/open-telemetry/community/issues/2406

open-telemetry / community Type / to search

Issues 105 Pull requests 6 Actions Projects 4 Security Insights

[Donation Proposal]: Beyla, eBPF auto-instrumentation tool for metrics and traces #2406

Open

grcevski opened on Oct 23, 2024 · edited by grcevski

Edits ⚙️ ...

Description

Grafana Labs would like to offer the donation of Beyla to the OpenTelemetry project.

Beyla is a mature eBPF-based auto-instrumentation tool for OpenTelemetry metrics and traces, for multiple languages and protocols. It enables cluster-wide/system-wide auto-instrumentation of applications without the need for application code/configuration changes or application restarts. To achieve this, Beyla uses a combination of protocol-level instrumentation based on network events and language/runtime-level instrumentation where needed. While Beyla works on bare metal installations, virtual machines, etc., the tool is also fully Kubernetes-aware and can be deployed as a daemonset or as a sidecar. Beyla is used by a number of customers in production, including Grafana Labs itself for the Grafana Cloud hosted offering.

Some of the main uses of Beyla are:

- Provide auto-instrumentation for programming languages where OpenTelemetry SDK zero-code auto-instrumentation is not supported, such as Rust, C++, Erlang, Zig, Ruby, Swift, Perl, Lua, Dart, R, Java GraalVM Native, Julia...
- Provide auto-instrumentation for legacy applications, where it's not easy to migrate the codebase to the OpenTelemetry SDK compatible frameworks.
- Provide auto-instrumentation for applications where the source is not available or are proprietary and/or distributed in binary form.
- Provide a unified way to capture application-level metrics across all different technologies used by a customer.
- Provide network-level metrics, regardless of the L3/L4/L7 protocol used for the purpose of building service graphs and reachability reports.
- Provide process-level metrics for instrumented applications.

Some of the core features of Beyla include:

Assignees
No one assigned

Labels
area/donation

Type
No type

Projects
No projects

Milestone
No milestone

Relationships
None yet

Development
Code with Co
No branches or pull re

Don't minimize no code contributions

Diana Todea 2:41 PM
@krol Hola, por ese PR no entiendo muy bien el ultimo comentario para añadir los heading IDs, es la misma cosa que falla por mi ultimo PR. Si sabes de que va o como se soluciona, podemos hacer una llamada. MGTV

#6134 feat: [es] Create docs/contributing/style-guide.md
open-telemetry/opentelemetry.io | Jan 31st | Added by GitHub

#6371 feat: [es] Create docs/contributing/pr-checks.md
open-telemetry/opentelemetry.io | Feb 18th | Added by GitHub

1 reply 3 replies Last reply 2 months ago

open-telemetry/opentelemetry.io feat: [es] Create docs/contributing/pr-checks.md ✓ 49
lang:es
#6371 by didiViking was merged yesterday • Approved

open-telemetry/opentelemetry.io feat: [es] Create docs/contributing/style-guide.md ✓ 38
lang:es
#6134 by didiViking was merged on Feb 19 • Approved

open-telemetry/opentelemetry.io feat: [es] Create docs/contributing/issues.md ✓ 51
lang:es
#6088 by didiViking was merged on Feb 14 • Approved

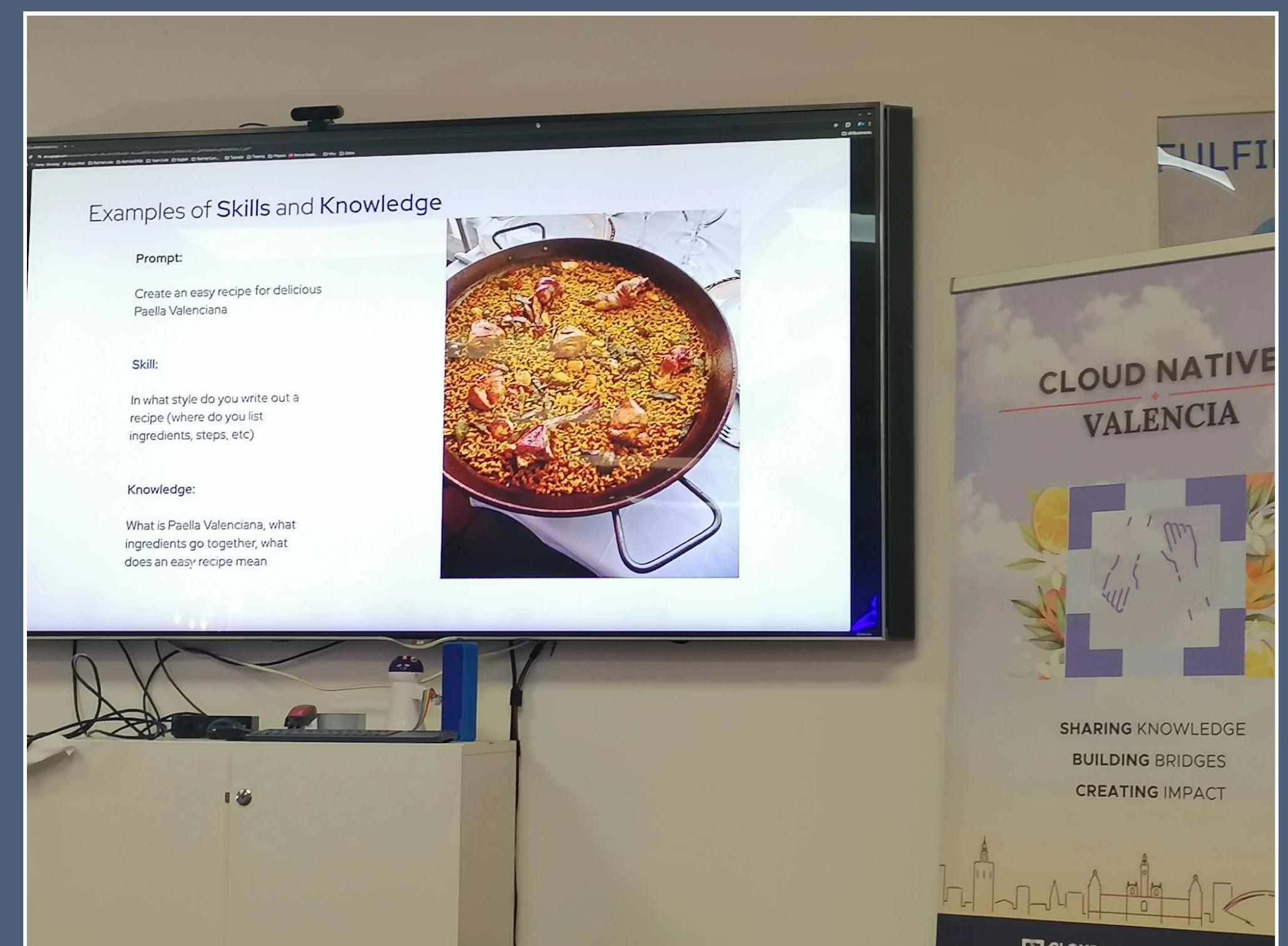
Nurture local communities



The screenshot shows the Cloud Native Valencia community group page on the Cloud Native Community Groups website. The top navigation bar includes 'Upcoming KCDs', 'Chapters', 'Events', 'Apply', and a search icon. A 'Log in' button is also present. Below the navigation is a large image of the futuristic architecture of the City of Arts and Sciences in Valencia. At the bottom, it says 'Europe' and 'Cloud Native Valencia' with a 'Join' button and '54 members'.

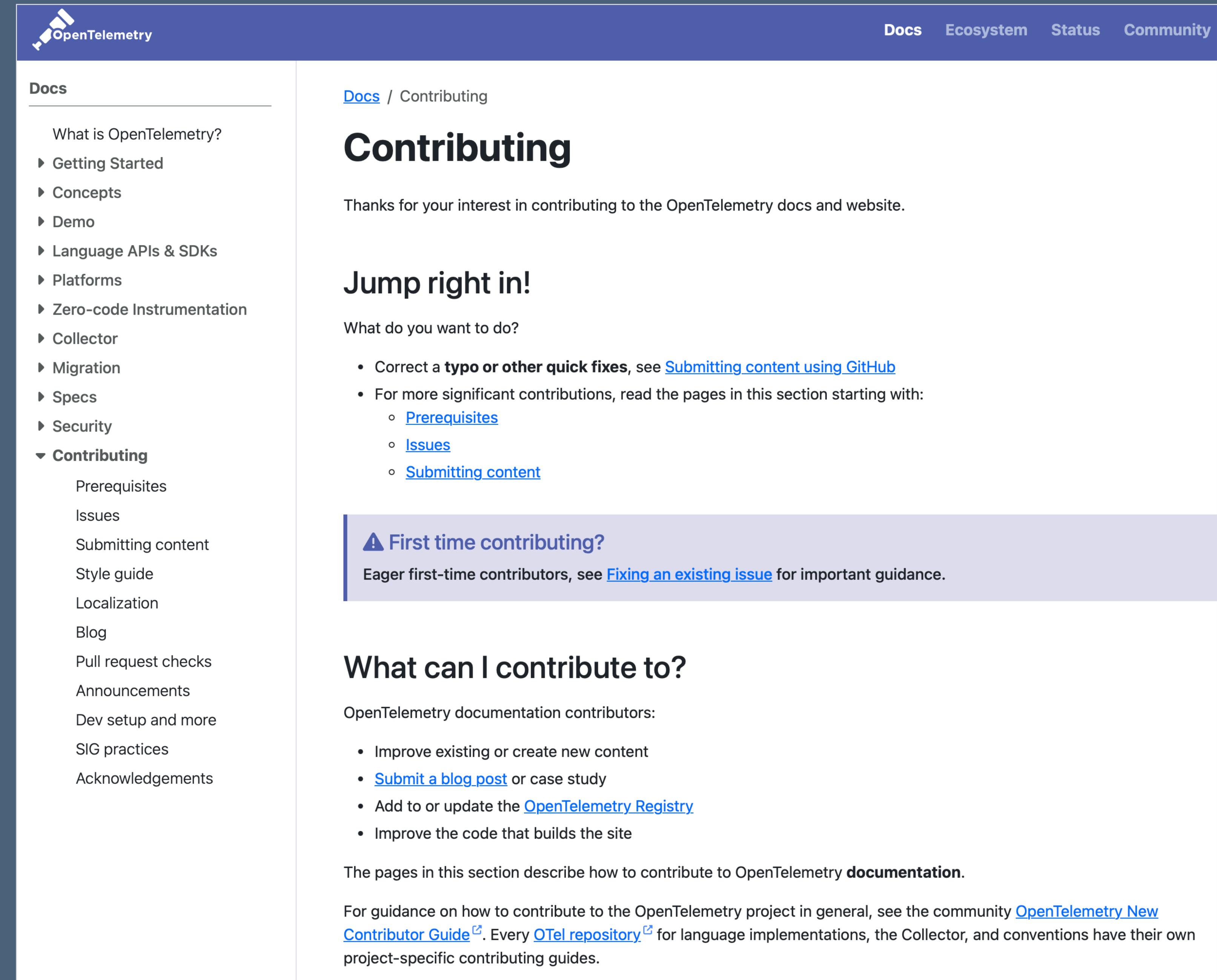
otel-docs-localization
otel-localization-es

<https://community.cncf.io/cloud-native-valencia/>



Be responsible of your contributions

★ Our contributions echo in many other open source projects and communities



The screenshot shows the OpenTelemetry documentation website. The top navigation bar includes links for Docs, Ecosystem, Status, and Community. The left sidebar has a 'Docs' heading and a list of topics: What is OpenTelemetry?, Getting Started, Concepts, Demo, Language APIs & SDKs, Platforms, Zero-code Instrumentation, Collector, Migration, Specs, Security, Contributing (which is expanded to show Prerequisites, Issues, Submitting content, Style guide, Localization, Blog, Pull request checks, Announcements, Dev setup and more, SIG practices, and Acknowledgements), and Acknowledgements.

The main content area shows the 'Contributing' page under the 'Docs / Contributing' path. It starts with a thank you message: "Thanks for your interest in contributing to the OpenTelemetry docs and website." Below it is a section titled "Jump right in!" with the sub-question "What do you want to do?". A bulleted list provides options: "Correct a **typo or other quick fixes**, see [Submitting content using GitHub](#)", "For more significant contributions, read the pages in this section starting with: [Prerequisites](#), [Issues](#), and [Submitting content](#)".

A purple sidebar box contains the heading "⚠ First time contributing?" and the sub-question "Eager first-time contributors, see [Fixing an existing issue](#) for important guidance."

The "What can I contribute to?" section lists tasks for documentation contributors: "Improve existing or create new content", "Submit a [blog post](#) or case study", "Add to or update the [OpenTelemetry Registry](#)", and "Improve the code that builds the site". It also notes that the pages describe how to contribute to the documentation.

The bottom section provides general guidance: "For guidance on how to contribute to the OpenTelemetry project in general, see the community [OpenTelemetry New Contributor Guide](#)". It mentions that every [OTel repository](#) for language implementations, the Collector, and conventions have their own project-specific contributing guides.

Share with communities

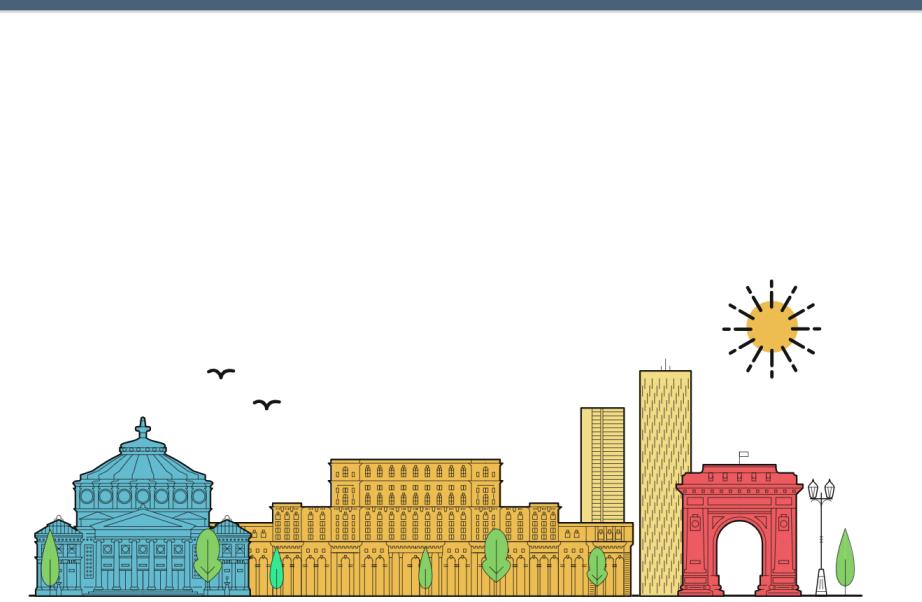


Romania

5–6 MAY 2025

Cloud Native Days Romania

The Kubernetes & Cloud Native community will gather at the Radisson Blu in Bucharest, Romania. Join us for one day technical event loaded with exciting talks and networking opportunities. Cloud Native Days Romania is aimed at developers, platform people, and other IT professionals with an interest in cloud native technologies.



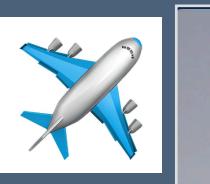
United States



June 26
Denver, Colorado

#OpenO1yCon #OTelDay

Spain



DIANA TODEA

Developer Advocate at Aircall

Diana is a Developer Advocate at Aircall. She has worked as a Senior Site Reliability Engineer focused on Observability. She is passionate about serverless, SecOps and machine learning. She is an active contributor to the OpenTelemetry open source project and supports women in tech.

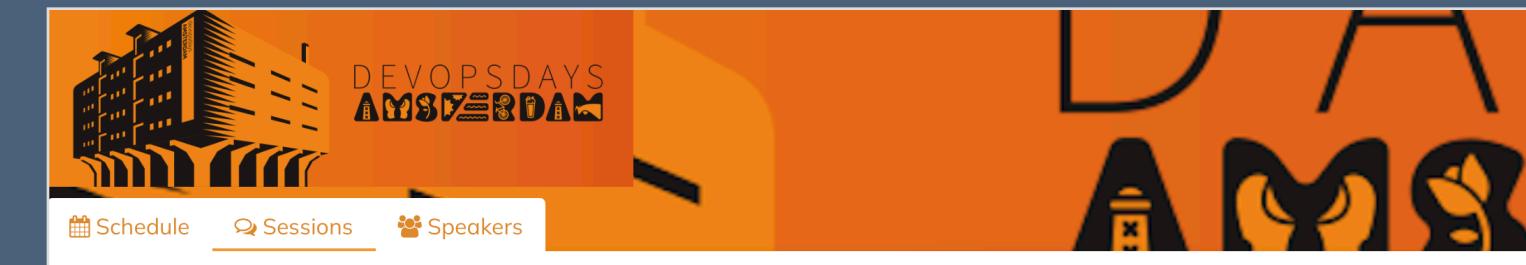
Sessions

- > From zero to developer: my one year serendipity journey with OpenTelemetry

Go back



The Netherlands



From zero to developer: my one year serendipity journey with OpenTelemetry

2025-06-19 15:05–15:30, Grote Zaal

Becoming a contributor to an open-source project is a transformative step in any developer's career. This session explores the journey from first-time contributor to active developer, covering best practices for navigating project communities, understanding codebases, and making meaningful contributions. Learn strategies for selecting the right project, mastering collaboration tools, and embracing the culture of open-source development. The audience will be inspired about my one year journey with the open source project OpenTelemetry and how I have built a proof of concept for it and achieved developer status for this project. By the end of this talk, the public will gain insights into the tools to become a better developer and how to build more engagement with the community.



Virginia Diana Todea

Diana is a Developer Advocate and a Senior Site Reliability Engineer. She is passionate about observability, machine learning and open source projects. She is an OpenTelemetry contributor and loves sharing her knowledge with the community by participating in tech conferences and writing articles.



Germany

OSMC

FROM ZERO TO DEVELOPER: MY ONE YEAR SERENDIPITY JOURNEY WITH OPENTELEMETRY

Becoming a contributor to an open-source project is a transformative step in any developer's career. This session explores the journey from first-time contributor to active developer, covering best practices for navigating project communities, understanding codebases, and making meaningful contributions. Learn strategies for selecting the right project, mastering collaboration tools, and embracing the culture of open-source development. The audience will be inspired about my one year journey with the open source project OpenTelemetry, how I have built a proof of concept for it and achieved developer status for this project. By the end of this talk, the public will gain insights into the tools to become a better developer and how to build more engagement with the community.

SPEAKER

Diana Todea
Aircall

Dominican Republic

From zero to developer: my one year serendipity journey with OpenTelemetry

Cloud Native Santo Domingo

Jul 12, 4:00 – 5:30 PM (GMT+2)

Virtual event

Why feedback matters for open source

 Diana Todea · 3 min read · Jan 7, 2025



My 2024 has been a year where I finally immersed myself in the open source realm. It has been on my mind for a while to start contributing to an open source project, however I wanted to pick the right project that resonated with my current on-going work.

Putting doubts aside, I was really impressed with [OpenTelemetry](#)'s popularity and the fact that so many organizations already adopted it into their production environments. When I started drafting my [GenerativeAI talk](#), I interacted with OpenTelemetry, APM and Elastic's Observability stack. Seeing how everything blended together was a great experience, so when the chance opened up to contribute to OpenTelemetry project, I jumped right into it.

[OTCA certification from Linux Foundation](#) has been released on the market just very recently. It is the first exam and certification covering the OpenTelemetry topic at its foundational level. I got the chance to contribute as a developer to OTCA exam, writing questions, double checking the documentation sources and putting myself in the shoes of a person who wants to get certified on OpenTelemetry. For me it has been the perfect opportunity to get closer to an open source project I admired from afar and interacting with the open source community.

Educate through written content

OpenObservabilitySummit and OTEL Community Day 2025 recap

 Diana Todea · 6 min read · 2 hours ago



Cloud Native Days Romania 2025-tales from the field

 Diana Todea · 4 min read · May 8, 2025



I had the honor to participate at [Cloud Native Days Romania](#)'s 2nd edition, hosted in Bucharest, Romania, on the 5th and 6th of May and honestly I was amazed. Meeting with a vibrant open source community, talented organizers and an overall curious audience that made me feel that I belonged, what a dream come true.



Take-aways

- ➡ Diversify your contributions
- ➡ Distinguish between code and no code contributions
- ➡ Even the tinniest feedback can help the community
- ➡ Prioritize and nurture local communities and docs localization projects
- ➡ Crosspost your findings on different channels
- ➡ Pay it forward: keep the collaboration open between OSS projects

Your journey matters!



Join the CNCF Slack
channel

#otel channels



#otel-profiles

#otel-system-metrics

#otel-contributor-experience



Diana Todea

Developer Advocate, Observability
SME, OTEL contributor | Keynote & In...



🤓 I played with OTel and I liked it

🤓 Why feedback matters for Open Source

🤓 Contribute to OTel (OTel blog)

🤓 <https://opentelemetry.io/blog/2025/devex-survey>

🤓 <https://opentelemetry.io/blog/2025/tribex-survey-results/>

🤓 <https://training.linuxfoundation.org/certification/opentelemetry-certified-associate-otca/>

🤓 <https://www.cncf.io/people/ambassadors/application-process/>

MOLTES GRÀCIES!

MUCHAS GRACIAS!

THANK YOU!



QUESTIONS? FEEDBACK?