

From 0 to developer my one year serendipity journey with OpenTelemetry

Diana Todea - Technical Advocate @Aircall



technical advocate
SRE focused on observability
OpenTelemetry contributor
`#otel-docs-localization`
`#otel-localization-es`
passionate about OSS projects, ML/AI



Diana Todea
didiViking · she/her

Hi, I'm Diana Todea, Senior SRE
passionate about Observability and
Machine learning. Welcome to my repo!

[Edit profile](#)

9 followers · 6 following

Spain

[in/diana-todea-b2a79968](#)

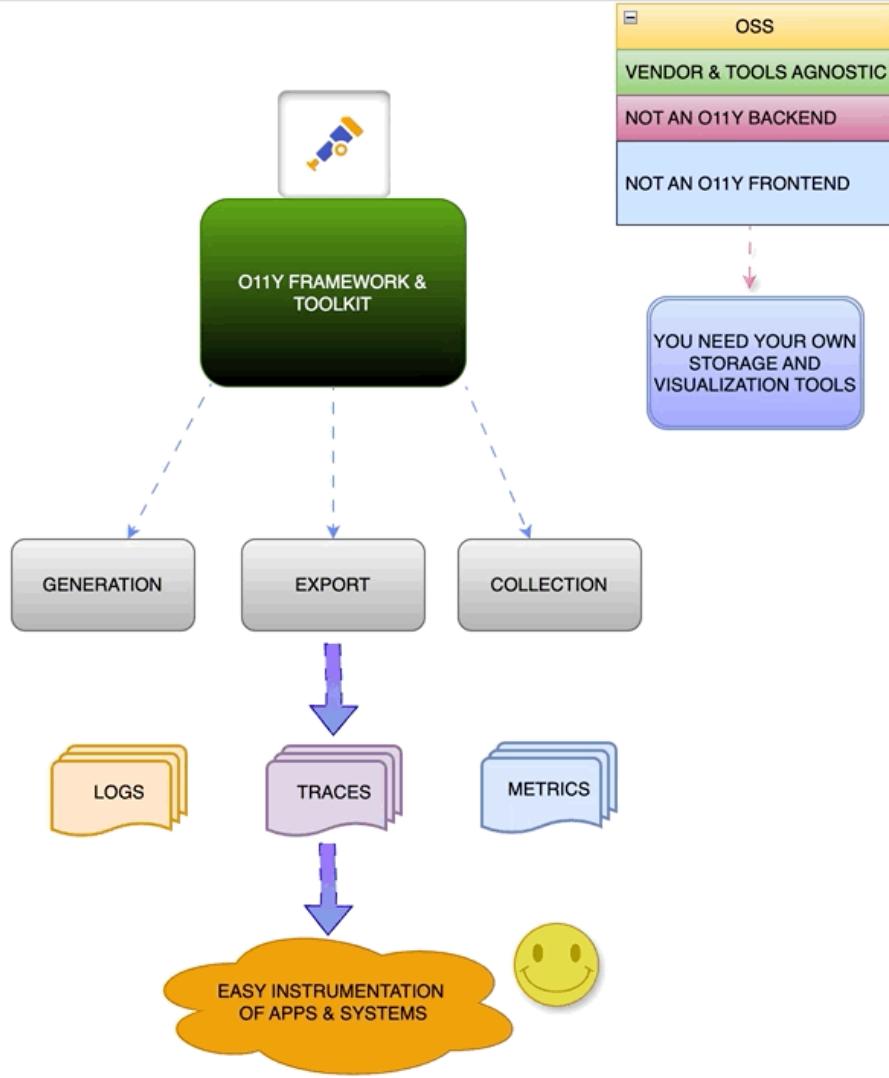
Achievements



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

a beginning

“OTel craze and an SRE’s curiosity”



“The OTel call for o11y SMEs”



Organization
Cloud Native Computing Foundation

Item Writing 1 My Items 21

Organization	Name	
Filter...	Filter...	
 Cloud Native Computing Foundation	OpenTelemetry Certified Associate (OTCA)	WRITE ITEMS

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!**



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

a middle

“Gee, what a great start, can I do more?”



“An SRE’s purpose is to turn everything into OTel”

“How about a POC?”



I played with OTEL and I liked it

 Diana Todea
4 min read · Jan 7, 2025

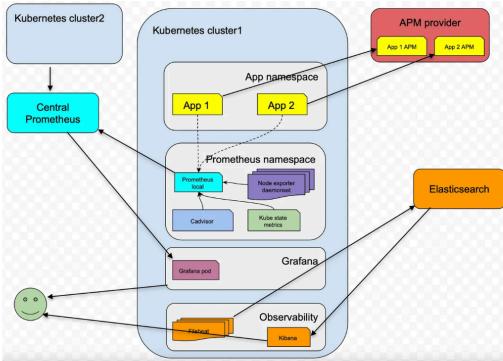
25 Q 1



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.



Great SRE+SWE collaboration

Bottlenecks:

- instrumentation comes with a lot of manual effort
- not the most smooth onboarding: confusing OTel docs
- not enough engineering time/power
- not satisfied with OSS visualization frontend
- OTel is not yet stable

Provide your developer feedback to the OTel community

A screenshot of a GitHub issue comment page. The URL is github.com/open-telemetry/opentelemetry-specification/issues/4255#issuecomment-2410636107. The repository is 'open-telemetry / opentelemetry-specification'. The issue is titled 'Truly auto instrumentation #4255'. A comment by user 'ostrolucky' is shown, dated Oct 11, 2024. The comment discusses the need for auto-instrumentation, mentioning 'opentelemetry-auto-*' packages and the desire for a more dynamic approach.

ostrolucky opened on Oct 11, 2024 - edited by ostrolucky

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 seems to work is you have to specify which functions to instrument. There are `opentelemetry-auto-*` packages, however all those are 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 threshold. 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

A screenshot of a GitHub issue page for issue #2406. The URL is github.com/open-telemetry/community/issues/2406. The repository is 'open-telemetry / community'. The issue is titled '[Donation Proposal]: Beyla, eBPF auto-instrumentation tool for metrics and traces #2406'. A comment by user 'grcevski' is shown, dated Oct 23, 2024. The comment describes Beyla as an eBPF-based auto-instrumentation tool for OpenTelemetry metrics and traces, highlighting its use cases and core features.

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

grcevski opened on Oct 23, 2024 - edited by grcevski

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:

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

“Don’t despair if it’s not in production.”

“I’m doing this by myself. You’ll see!”

A screenshot of a GitHub pull request (PR) discussion thread. The PR is titled "#6134 feat: [es] Create docs/contributing/style-guide.md". The first comment is from Diana Todea (@krol) at 2:41 PM, asking for help with heading IDs. Another comment from open-telemetry/opentelemetry.io follows, mentioning PR #6371. The thread shows 1 reply and 3 replies. The last reply was 2 months ago.

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. MGIV
#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

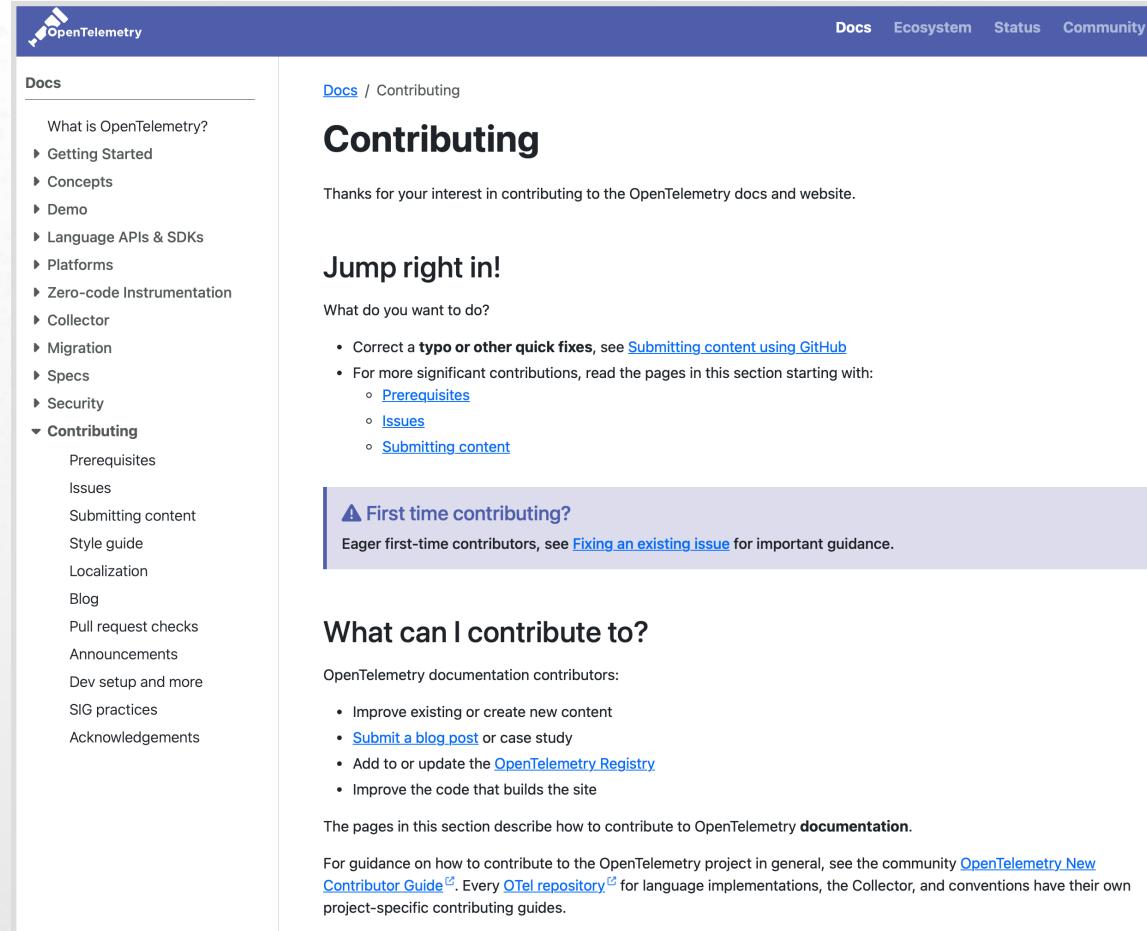
when in doubt, start small
talk to the maintainers, active contributors
get involved with “close to your heart projects”

A list of three GitHub pull requests (PRs) from the open-telemetry/opentelemetry.io repository:

- #6371 by didiViking was merged yesterday · Approved. Status: lang:es. 49 replies.
- #6134 by didiViking was merged on Feb 19 · Approved. Status: lang:es. 38 replies.
- #6088 by didiViking was merged on Feb 14 · Approved. Status: lang:es. 51 replies.

#otel-docs-localization
#otel-localization-es

be kind & supportive
be modest
avoid negativity
provide clear
feedback



The screenshot shows the OpenTelemetry website's "Contributing" page. At the top right, there are links for "Docs", "Ecosystem", "Status", and "Community". The main content area has a dark blue header with the "OpenTelemetry" logo. Below the header, the page title is "Contributing". A sub-header "Jump right in!" is followed by a list of steps for first-time contributors. Another section, "What can I contribute to?", lists ways to help improve the documentation.

Docs / Contributing

Contributing

Thanks for your interest in contributing to the OpenTelemetry docs and website.

Jump right in!

What do you want to do?

- 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](#)
 - [Submitting content](#)

⚠ First time contributing?

Eager first-time contributors, see [Fixing an existing issue](#) for important guidance.

What can I contribute to?

OpenTelemetry documentation contributors:

- Improve existing or create new content
- [Submit a blog post](#) or case study
- Add to or update the [OpenTelemetry Registry](#)
- Improve the code that builds the site

The pages in this section describe how to contribute to OpenTelemetry **documentation**.

For guidance on how to contribute to the OpenTelemetry project in general, see the community [OpenTelemetry New Contributor Guide](#). Every [Otel repository](#) for language implementations, the Collector, and conventions have their own project-specific contributing guides.

pedantic



aim for a more “user-friendly”
developer experience

“Soon the whole world will be OTel”

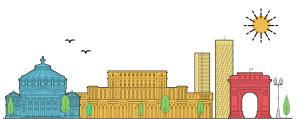
Community work

<https://cloudnatedays.ro/>

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.



<https://devopsdays.org/events/2025-amsterdam>



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 to tech conferences and writing articles.

<https://events.linuxfoundation.org/openobservabilitycon-otel-community-day>

June 26
Denver, Colorado
#OpenO1lyCon #OTelDay



⚡ Lightning Talk: From Zero To Developer: My One Year Serendipity Journey With OpenTelemetry - Diana Todea, Aircall

Thursday June 26, 2025 1:50pm - 2:05pm MDT

OTel Room

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.

Speakers



Diana Todea

Technical Advocate, 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 te... [Read More](#)

OTel Community Day, Practical OpenTelemetry

Level Any

<https://www.devbcn.com/>



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 tec...

Sessions

▶ [From zero to developer: my one year serendipity journey with OpenTelemetry](#)
[Go back](#)

Why feedback matters for open source



Diana Todea · 3 min read · Jan 7, 2025

2 Q

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.

Write & share with the community

Document writing: a great way to contribute to Open Source projects



Diana Todea · 2 min read · Feb 20, 2025

W+ Q

This year I have started further contributing to Open Source projects, in my case this was [OpenTelemetry](#). OTel has a great need for documentation translation in various languages, you can check their announcement [here](#).

Since my favorite Open Source project needed help I decided to add my contribution helping translating documentation for the [Spanish website](#).

We are just getting started so the need for contributors is acute. It doesn't necessarily take to write code in order to contribute to an open source project, it can be a small step as helping translating a page.

All the steps to start contributing to site localization in different languages is explained in this [directory](#).

In this short article I wanted to share my experience so far as a first time contributor to OTel documentation. It was been an easy onboarding from the first moment and I received support from the maintainers and other contributors of this site via the Slack channel: [#otel-docs-localization](#). Each

We need your contributions!

#otel-system-metrics

#otel-profiles

#otel-contributor-experience



Join CNCF Slack channel



#otel channels

“There’s no end, just keep going”

Give it a read, why not?

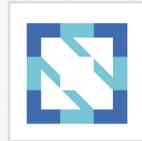
- [I played with OTel and I liked it](#)
- [Why feedback matters for Open Source](#)
- [Contribute to OTel \(OTel blog\)](#)
- [Slack: #otel-sig-end-user, #otel-demo](#)
- [<https://opentelemetry.io/blog/2025/devex-survey>](#)
- [<https://training.linuxfoundation.org/certification/opentelemetry-certified-associate-otca/>](#)
- [<https://ottl.run/>](#)

Multumesc!

Do you have any questions?

github@didiViking

www.linkedin.com/in/diana-todea-b2a79968



DM

