



OPEN SOURCE
MONITORING
CONFERENCE

NOV
18–20
2025

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

DIANA TODEA
VICTORIAMETRICS

OTel member and contributor

Cloud Native Days Romania organizer

Co-lead CNCF Merge-Forward Neurodiversity



Diana Todea
DX@VictoriaMetrics | Observability
SME | OpenTelemetry docs localizatio...



Have the curiosity to begin

Linux Foundation community call for observability SMEs

Creating the first ever certification and exam for OpenTelemetry

Find your personal motivation when starting an OSS contribution



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

OTel adoption in companies

Create a POC: validate OTel in orgs with fragmented o11y

PROs: vendor neutrality, standard for OSS only

To be improved: docs, manual instrumentation, count for overhead

Limited: customize OSS visualization/dashboards

Not available: support in production

OTel considers profiling and instrumenting as a different scope!



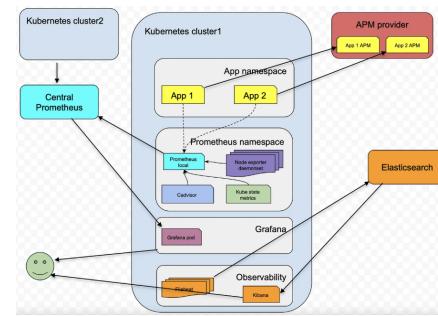
 Diana Todea · 4 min read · Jan 7, 2022



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

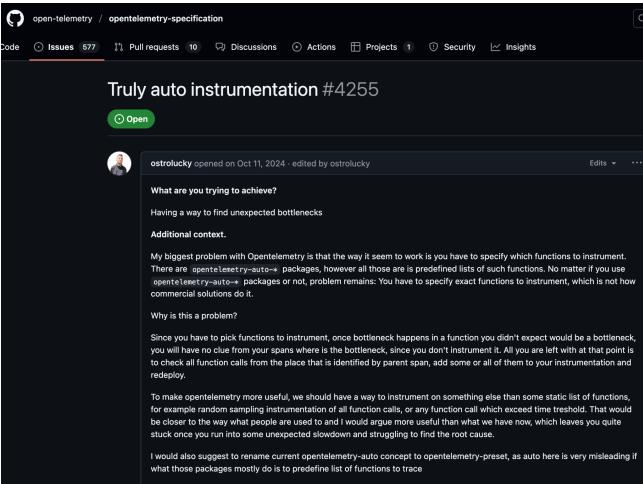


Simple, Reliable, Efficient Monitoring

victoriameetrics.com

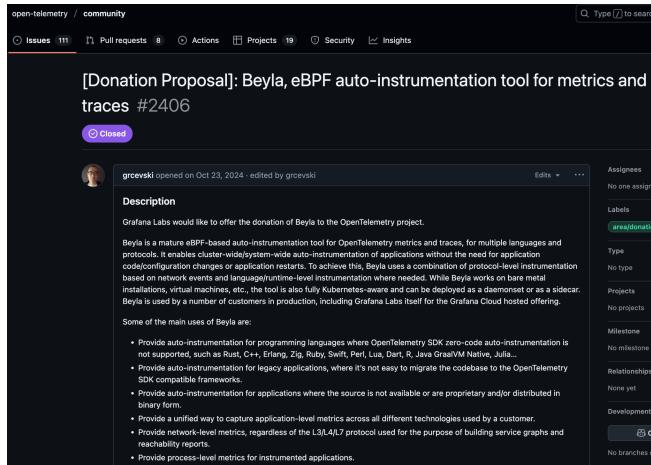
Provide developer feedback

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



A screenshot of a GitHub issue page for the OpenTelemetry specification repository. The issue is titled "Truly auto instrumentation #4255". It has 111 comments, 10 pull requests, and 19 discussions. The issue itself is labeled "Open". The description discusses the current state of auto-instrumentation in OpenTelemetry, noting that it requires explicit function lists and suggesting a way to find unexpected bottlenecks. The body of the issue contains several paragraphs of text and code snippets.

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



A screenshot of a GitHub issue page for the OpenTelemetry community repository. The issue is titled "[Donation Proposal]: Beyla, eBPF auto-instrumentation tool for metrics and traces #2406". It has 8 comments, 1 pull request, and 19 discussions. The issue is labeled "Closed". The description is from Grafana Labs, proposing to donate the Beyla tool to the project. The Beyla tool is described as a mature eBPF-based auto-instrumentation tool for OpenTelemetry metrics and traces, supporting multiple languages and protocols. It enables automatic codeless auto-instrumentation of applications without the need for application code changes or recompilation. To achieve this, Beyla uses eBPF-based auto-instrumentation based on network events and language-specific-level instrumentation where needed. Beyla works on bare metal installations, virtual machines, etc., and 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 `gRPC` protocol used for the purpose of building service graphs and reachability reports.
- Provide process-level metrics for instrumented applications.

Individual Contributor in Cloud Native

Not org
dependent

Code
NO Code
contribution

Multiple
OSS
projects

Support local communities

#otel-docs-localization
#otel-localization-es
#cloud-native-valencia
#cnd-romania



Localization



NO Code
contributions

= **HUGE IMPACT**

Empower OTel localization

Submit an issue, create a first PR
Build a community around localization
New SIG: #otel-localization-ro

Adding Romanian as new docs localization language #7626

[Open](#)

didiViking opened last month · edited by svrmn

Edits · Member · ...

Language info:
◦ ISO 639-1 language code: RO
◦ Language name: Romanian

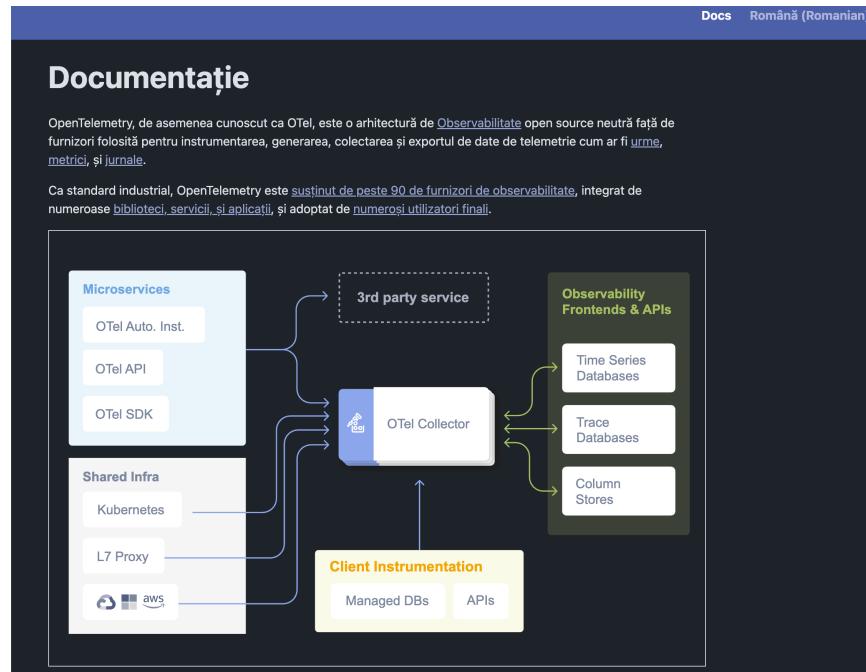
Locate team info:
◦ Locale mentor: [@didiViking](#)
◦ Contributors: [@victorvarza](#)

Read through [Localization](#) and all other pages in the Contributing section

Localize site homepage to Romanian [I+ feat: \[ro\] Create content/ro/_index.md #7626 #7628](#)

OTel maintainers:
◦ Update hugo.yaml
◦ Configure cSpell and other tooling support
◦ Create an issue label for lang:RO
◦ Create org-level group for RO - approvers

Update components owners for content/RO



Share with larger audiences



Developer Experience contributions



DIANA TODEA

Developer Advocate at Aircall

Sessions

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

Go back

OSMC

SUMMIT D'ATELIER

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 by 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 the 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

5-6 MAY 2025

Cloud Native Days Romania

The CloudNative & CloudNative community will gather at the Bucharest Convention Center for a two-day technical event loaded with exciting talks and networking opportunities. Cloud Native Days Romania is aimed at developers, system administrators, and other professionals with an interest in cloud-native technologies.

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

Cloud Native Santo Domingo

[Online] Women in Cloud Era @ Cloud Native Kuala Lumpur Meetup - September 2025



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 by 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.



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.

Open ObservabilityCon + OTel Community Day

June 26

Denver, Colorado

#OpenO11yCon #OTelDay



VictoriaMetrics

Simple, Reliable, Efficient Monitoring

victoriametrics.com

Educate through written content

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.



Formerly known as Kubernetes Community Days Romania, CND Romania has reinvented itself this year adding an extra day for workshops. From CI/CD pipelines with Dagger, to GitOps with FluxOperator, Ray and Kubernetes for ML projects, Istio Service Mesh and many other exciting topics, no wonder the workshop day reached its full capacity.

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.

OpenTelemetry project has a vast documentation with a tendency to get complicated really fast. Some parts are still developing and definitely are in need of developers' constant feedback. It's hard to perceive OpenTelemetry's level of difficulty from afar, you actually have to put your working gloves and deploy it. Later on in 2024, I had the chance to get my hands "dirty" with such an observability project and build the proof of concept for OpenTelemetry with several SDKs, documenting my experience in this article.

OpenObservabilitySummit and OTEL Community Day 2025 recap

Diana Todea · 6 min read · Jun 30, 2025



Last Thursday in Denver, US, OTEL Community Day event took place as part of the Open Observability Summit and it was a blast. I had the chance to participate with a lightning talk and learn about all the latest releases within the Open Observability realm.

Below, you can find the summary based on the sessions I participated at, the official videos will be released on the CNCF YouTube channel in a couple of weeks. The OTEL Community Day had two parallel tracks and some of the presentations are already uploaded on the schedule page.

Inspire other Cloud Native contributors

MOTIVATION = 



VictoriaMetrics

Simple, Reliable, Efficient Monitoring



Robert Klonner  · 1st
DevOps Engineer | Golden Kubestronaut
6d · 

...

Celebrating my first-time Open Source contribution in the ArgoCD Diff Preview project 

Why I am sharing this

I just overcome the initial barrier to do my first OS contribution and I want to motivate others to try it too. It needed several attempts for myself in the last months until it finally worked out. So give it a try and don't give up!

Which strategy finally worked for me

If you are just starting – do not look actively for random projects where you can contribute. Use open source tools and when you stumble upon a problem, raise a well described issue, make suggestions, create documentation you were missing or even improve some lines of code. The benefit of this strategy is that you bring in domain knowledge as you already worked with the tool and probably did some debugging. This fact makes it much easier to make a valuable suggestion.

What brought me to the ArgoCD Diff Preview project

I was looking for a safe way to refactor k8s manifests managed by Kustomize by automatically posting the rendered diff as a pull request comment.
If you also want more insight in manifest changes and use ArgoCD for your deployments I can definitely recommend to try it out

👉 <https://lnkd.in/gt3WpG3k>

What was my contribution

I setup the environment for ArgoCD Diff Preview on Openshift during a POC and had to overcome some issues to make it work. While documenting this for myself I thought I could share that directly with the project and give something back. This resulted in a new documentation section in the project (<https://lnkd.in/gJEE-kU>) and a few lines of Go for a fall back to make the login also work for ArgoCD instances managed by the ArgoCD Operator (<https://lnkd.in/guW8t5Ft>).

Special thanks to:

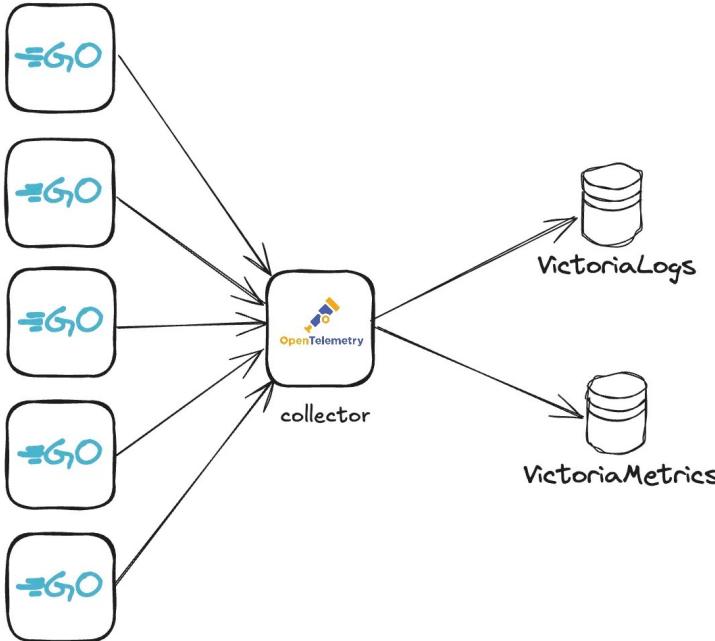
- **Dag Bjerre Andersen:** For the warm welcome to the project, your super fast reactions, paying attention to all kinds of issues and the straightforward discussions.
- **Kostis Kapelonis:** For the discussion on k8s manifest refactoring strategies and recommending to try ArgoCD Diff Preview.
- **Diana Todea:** For sharing your open source contribution story at the [Cloud Native Days Austria](#) and spreading the motivation to others.

victoriametrics.com

OpenTelemetry Community Award 2025 - KubeCon Atlanta



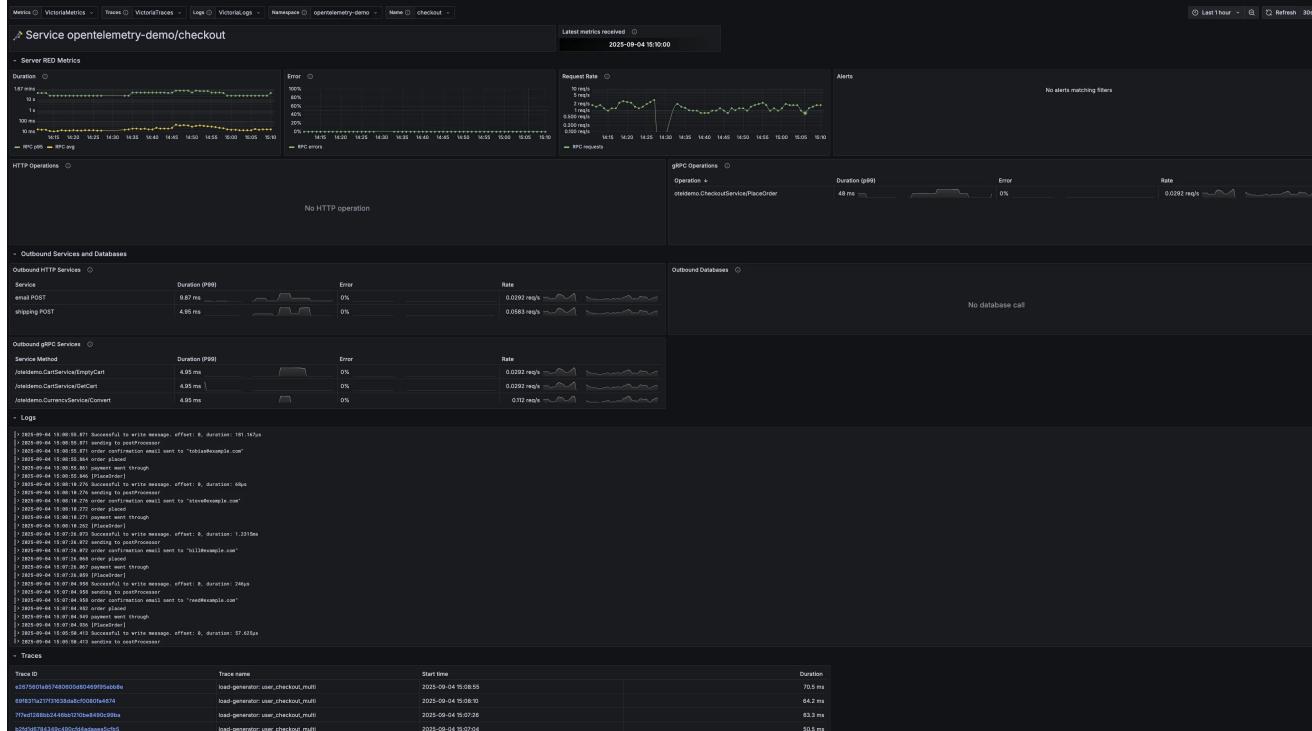
VictoriaMetrics + OpenTelemetry



VictoriaMetrics and VictoriaLogs support ingestion of metrics and logs in the OTEL format.
Pre-requirements:
k8s cluster, kubectl, helm



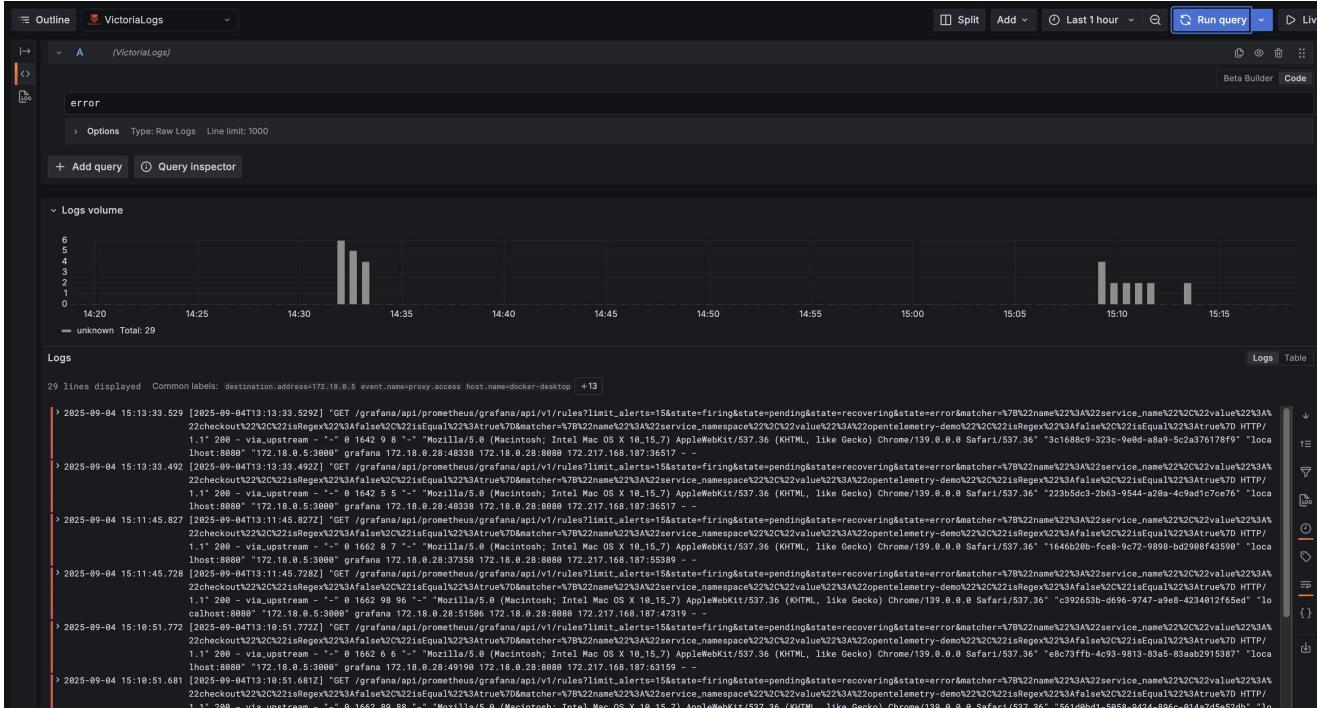
VictoriaMetrics + OpenTelemetry Demo



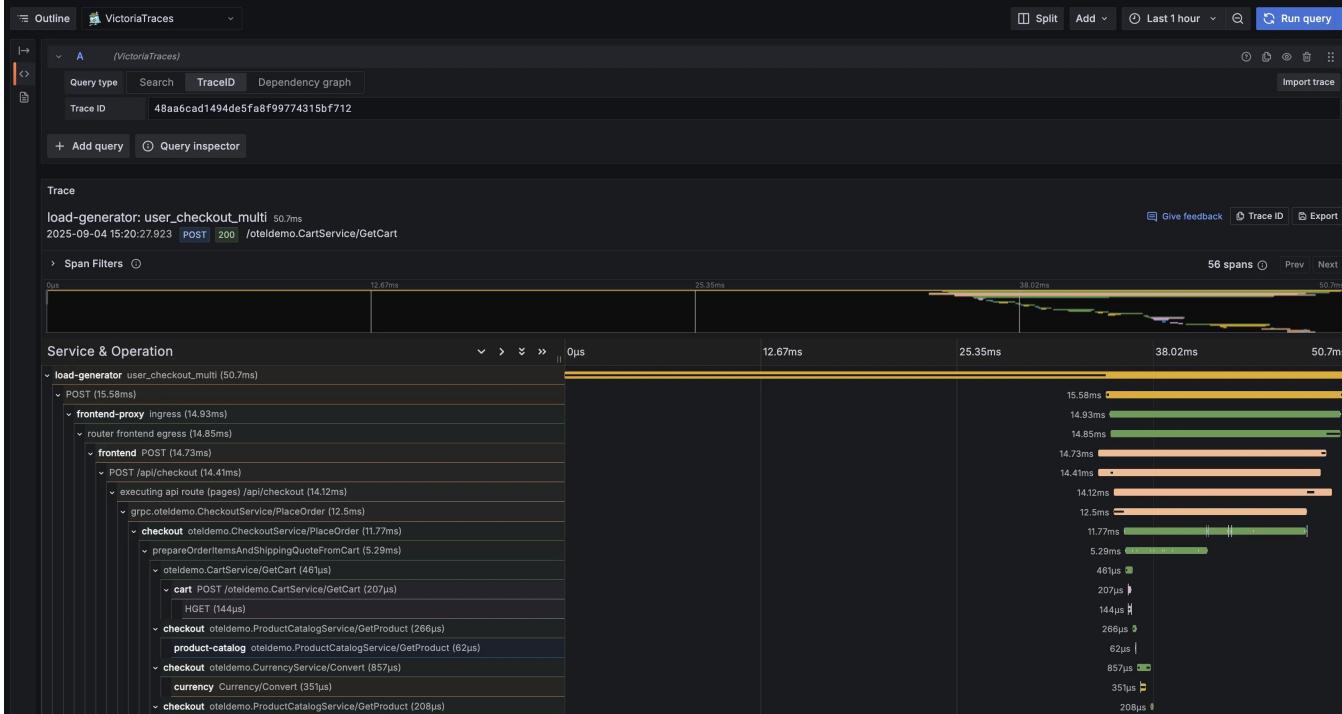
Explore VictoriaMetrics in the OTel demo



Explore VictoriaLogs in the OTel Demo



Explore VictoriaTraces in the OTel Demo



Keep your on these OpenTelemetry initiatives

[README](#) [Code of conduct](#) [Contributing](#) [Apache-2.0 license](#) [Security](#)

OpenTelemetry Weaver



Observability by Design
Treat your telemetry like a public API

CI [passing](#) [codecov](#) 78% [Security Audit](#) [passing](#) License Apache 2.0 Slack #otel-weaver

OpenTelemetry Weaver helps teams build observability by design, enabling consistent, type-safe, and automated telemetry through semantic conventions. With Weaver, you can define, validate, and evolve your telemetry schemas, ensuring reliability and clarity across your systems.

What is Observability by Design?

Have you ever experienced:

- Broken alerts after a deployment because metric names changed?
- Complex, hard-to-understand queries due to inconsistent naming?
- Teams struggling to interpret unclear or undocumented signals?
- Missing critical instrumentation discovered only in production?

Observability by Design solves these problems by treating your observability signals (metrics, traces, logs) as a first-class public API that requires the same quality standards as your code.

An introduction to Weaver and Observability by Design is presented in the official blog post: [Observability by Design: Unlocking Consistency with OpenTelemetry Weaver](#)

[README](#) [Code of conduct](#) [Contributing](#) [Apache-2.0 license](#) [Security](#)

OpenTelemetry Injector

The OpenTelemetry injector is a shared library (written in [Zig](#)) that is intended to be used via the environment variable `LD_PRELOAD`, the `/etc/ld.so.preload` file, or similar mechanisms to inject environment variables into processes at startup.

It serves two main purposes:

- Inject an OpenTelemetry Auto Instrumentation agent into the process to capture and report distributed traces and metrics to the [OpenTelemetry Collector](#) for supported runtimes.
- Set resource attributes automatically, (for example Kubernetes related resource attributes and service related resource attributes in environments where this is applicable).

The injector can be used to enable automatic zero-touch instrumentation of processes. For this to work, the injector binary needs to be bundled together with the OpenTelemetry auto-instrumentation agents for the target runtimes.

Official RPM and DEB packages that contain the injector as well as the auto-instrumentation agents are available, and can be downloaded from the [releases page](#). The OpenTelemetry injector Debian/RPM packages install the OpenTelemetry auto-instrumentation agents, the `libotelinject.so` shared object library, and a default configuration file to automatically instrument applications and services to capture and report distributed traces and metrics to the [OpenTelemetry Collector](#).

The `opentelemetry-injector` deb/rpm package installs and supports configuration of the following Auto Instrumentation agents:

- [Java](#)
- [Node.js](#)
- [.NET](#)

Make your own journey!

- Your motivation is your super power!
- Diversify your contributions
-  code-no-code contributions
- Pair up and mentor
- Share your end user stories!



Resources

[I played with OTel and I liked it](#)

[Why feedback matters for Open Source](#)

[Full-Stack Observability with VictoriaMetrics in the OTel Demo](#)

<https://opentelemetry.io/community/>

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

<https://github.com/open-telemetry/opentelemetry-injector>

<https://opentelemetry.io/blog/2025/otel-injector/#what-it-does>

<https://github.com/open-telemetry/weaver>

<https://opentelemetry.io/blog/2025/otel-weaver/>

<https://docs.victoriametrics.com/guides/getting-started-with-opentelemetry/>

<https://github.com/VictoriaMetrics-Community/opentelemetry-demo>



Danke!

Bsky: @didiviking.bsky.social

X: @dianavtodea

Github: @didiViking/Conferences_Talks

LinkedIn: @diana-todea-b2a79968



VictoriaMetrics Community