Observability Now:
A Hybrid Approach to Open Telemetry Instrumentation

Justin Castilla
Senior Developer Advocate @ Elastic

Why Observability Matters

Observability isn't just collecting data—it’s about making sense of it.

The Three Pillars:

- **Logs** (what happened),
- **Metrics** (how it's performing),
- **Traces** (how it flows).

Your Approach Matters:

- **Automatic instrumentation** gives broad coverage but lacks business-specific details.
- **Manual instrumentation** provides deep insights but takes more effort.
- **Hybrid instrumentation** = best of both worlds.

Open Telemetry: The Open Standard

- OpenTelemetry (OTel) is an open-source framework for collecting telemetry data
- Supports multiple backends—send your data to Elastic, Grafana, Datadog, or any OpenTelemetry-compatible system.
- Works with Flask, Django, and FastAPI and many common libraries
- Widely Adopted: Backed by CNCF, used in cloud-native and microservices architectures.

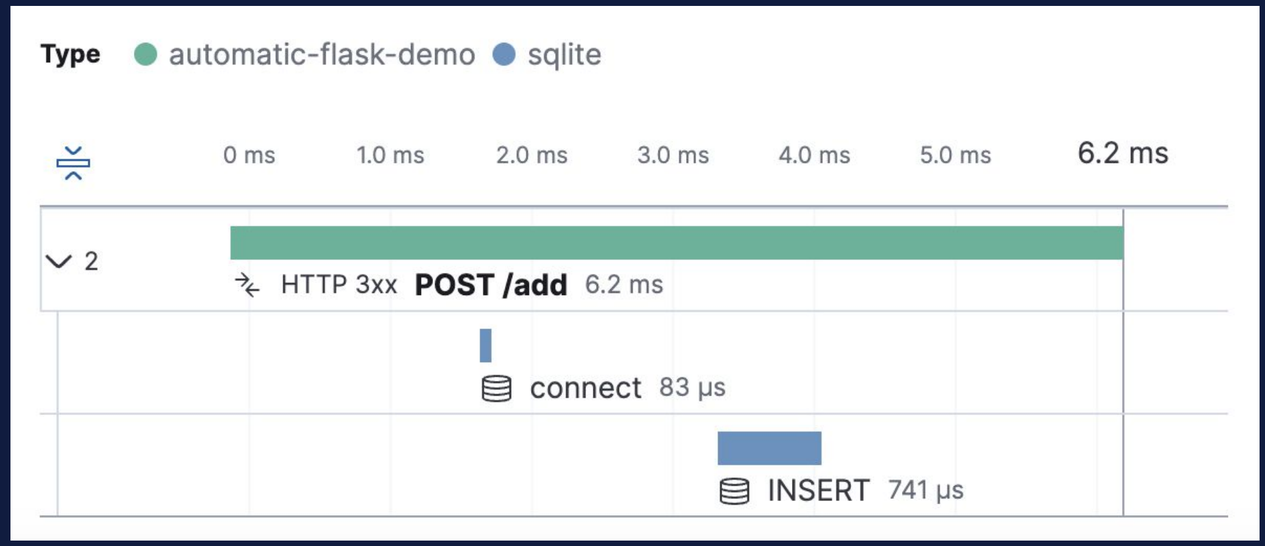
Automatic Instrumentation - Quick and Easy

Pros:

- Fast setup, minimal code changes.
- Covers HTTP, database calls, gRPC, and other libraries.
- Great for getting started.

Cons:

- Generates a lot of spans—some unnecessary and costly.
- Lacks business logic context (e.g., order processing steps).
- Might not always provide enough detail for debugging.



elastic

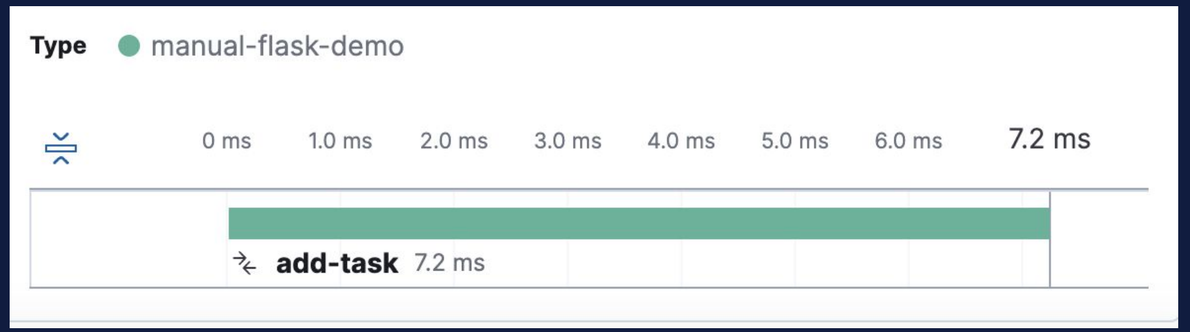
Manual Instrumentation - Granularity and Control

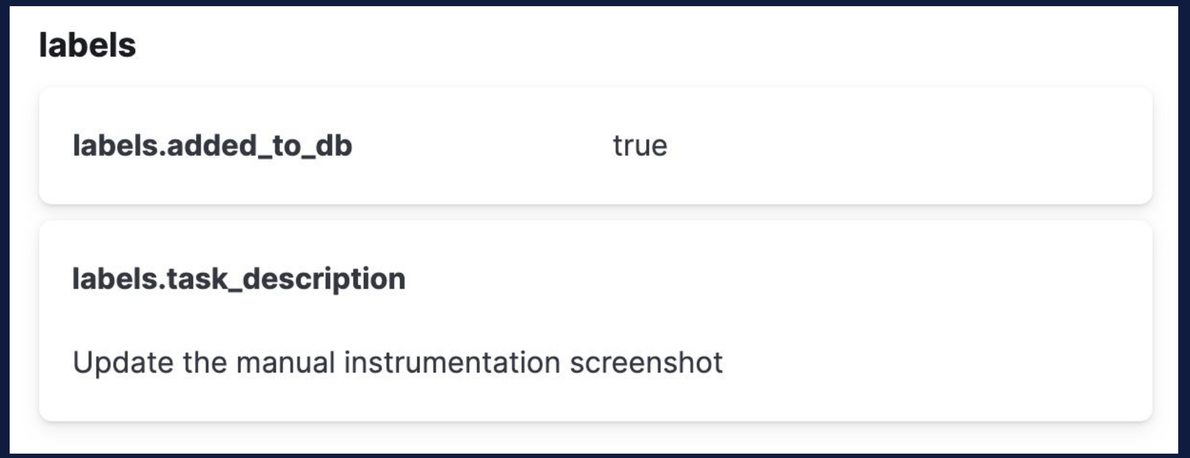
Pros:

- Lets you define meaningful spans where it matters most.
- Helps track critical business flows (e.g., “Order Validation,” “Payment Processing”).
- Reduces noise—focuses only on what matters.

Cons:

- Requires developer effort.
- Risk of missing instrumentation without careful planning.





elastic

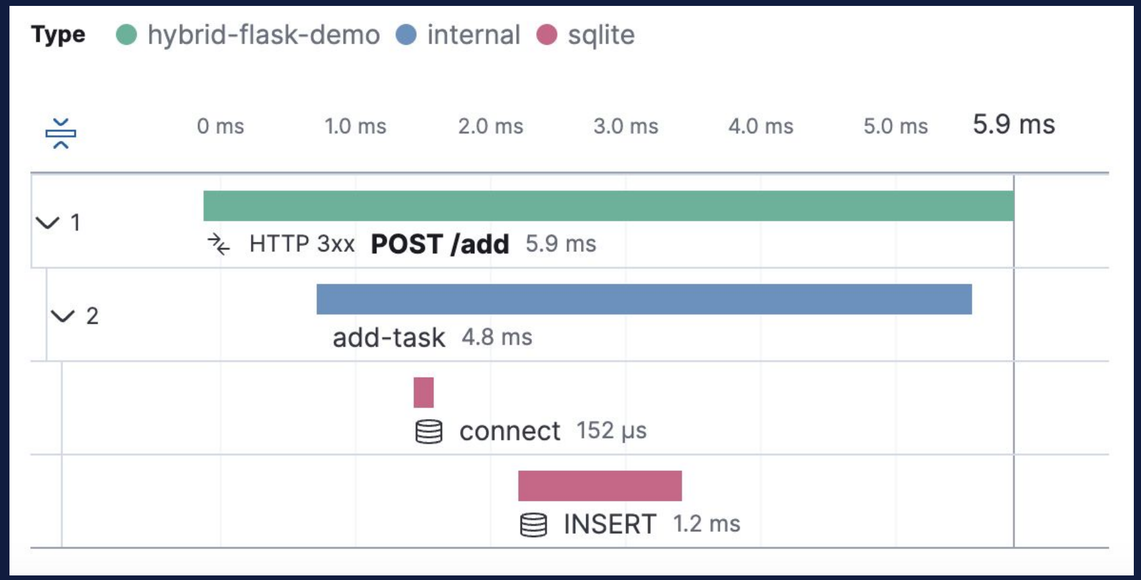
Hybrid Instrumentation - The “sweet spot”

Start with automatic instrumentation, then refine using manual spans.

Use OpenTelemetry config options to dial back and reduce noise:

- Exclude unnecessary libraries
- Adjust overall sampling rates
- Use span processors for filtering

Let **automation** handle the heavy lifting, then use **manual** spans to enhance critical workflows.



labels

labels.added_to_db	true
labels.process_runtime_description	3.12.3 (main, Jul 31 2024, 20:00:15) [Clang 15.0.0 (clang-1500.3.9.4)]
labels.task_description	Give away more swag so you don't have to carry it all home
labels.telemetry_auto_version	0.46b0

elastic

What just happened?!

Observability with Open Telemetry isn't about *more* data—it’s about the *right* data.

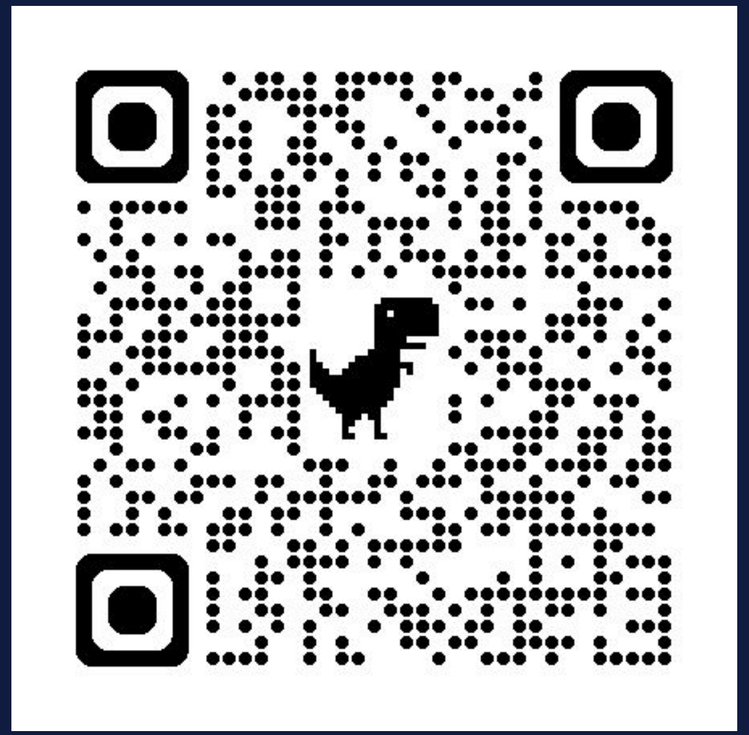
Automatic instrumentation gets you started fast, but **manual** spans add clarity.

Elastic and many other Data Platforms provide powerful tools to visualize and refine your telemetry.

Check out Yash Verma’s PyCascades talk on Open Telemetry

Start simple, find your data, pull back the data pressure.

Take a look at my sample Flask demos for all three scenarios!



elastic