
A
Practical Perfetto Introduction
for AOSP and Linux Developers
INNOVATE. INTEGRATE. EXCEED.
inovex

stefan.lengfeld@inovex.de
stefan@lengfeld.xyz
@stefan-lengfeld
@lengfeld
stefan.lengfeld.xyz

Hello,

I’m Stefan Lengfeld

Embedded Software Developer, Germany

- Embedded Systems (Linux and Android)
- Linux Kernel
- Build systems
- Linux and Android Graphics Stack
- Performance and Tracing

inovex

This is my third presentation
about tracing with Perfetto (and systrace).

I’m a
Perfetto Enthusiast by choice.

https://perfetto.dev/docs/contributing/getting-started#community

Perfetto is so cool. Try it out and get help!

Support by Perfetto devs:

- Mailinglist/Google Group
- Discord Server

Perfetto

Support by me:

- See my contact details
- Or join the discord on aosp-devs.org for AOSP/Android system tracing

inovex

AGENDA

- Introduction
  a. Android
  b. Embedded Linux
- Advanced usage
  (Tips & Tricks)
- Outlook

inovex

Introduction

https://commons.wikimedia.org/wiki/Category:G%C3%B6del,_Escher,_Bach

Why Tracing?

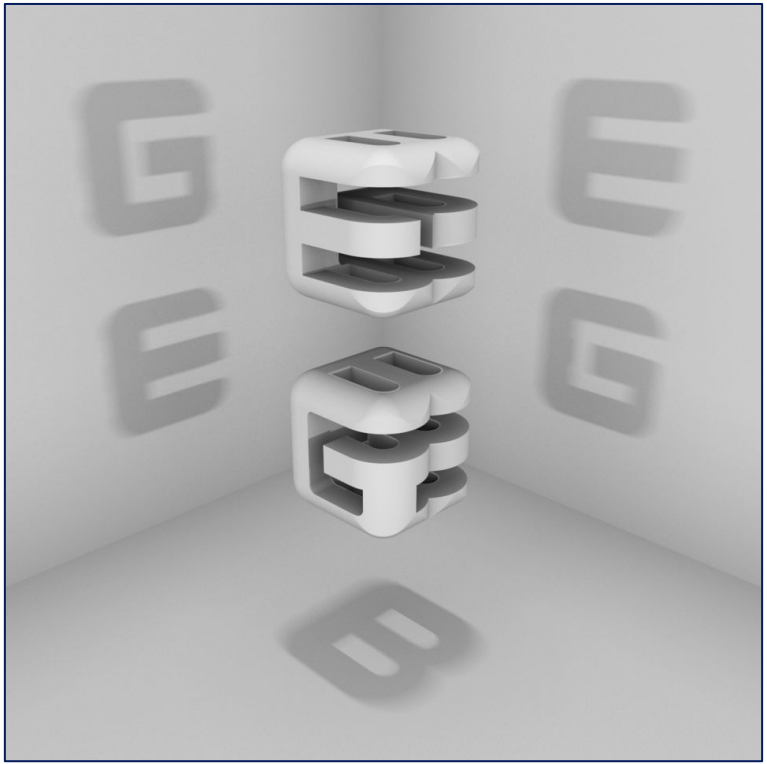
When you only look at the source code, you just see **G**.

But when you trace, you also see

- **E** (the runtime behavior of your code – in-app tracing) and
- **B** (the runtime behavior of the other parts of the system – system-wide tracing)

It this avoids:

> *[...] premature optimization is the root of all evil.* Donald Ervin Knuth



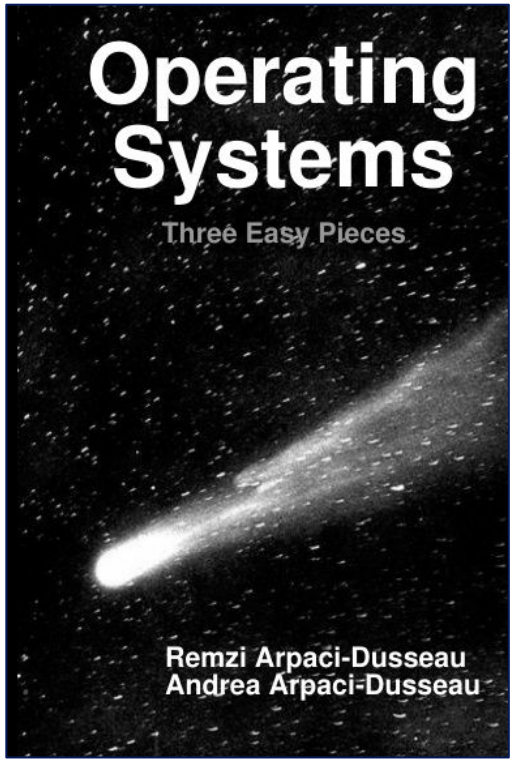
Operating Systems: Three Easy Pieces and Modern Operating Systems

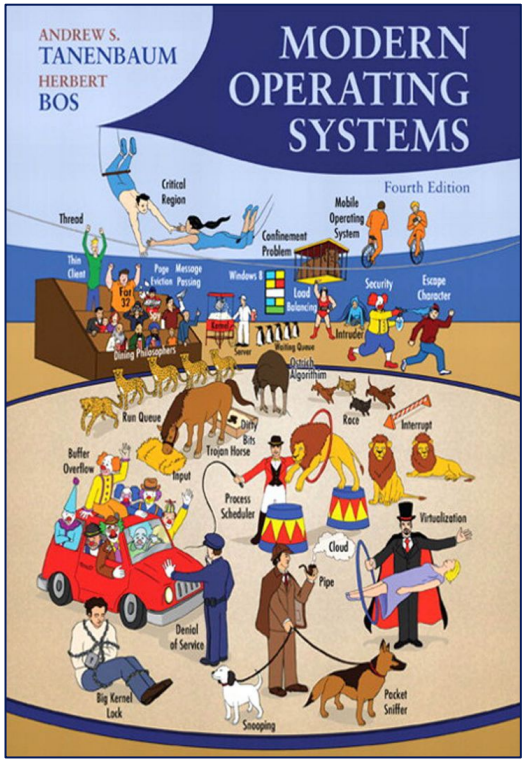
Prerequisites

You need to know some OS fundamentals:

- processes, threads
- thread states
- system calls
- scheduling
- (virtual) memory
- (call) stacks

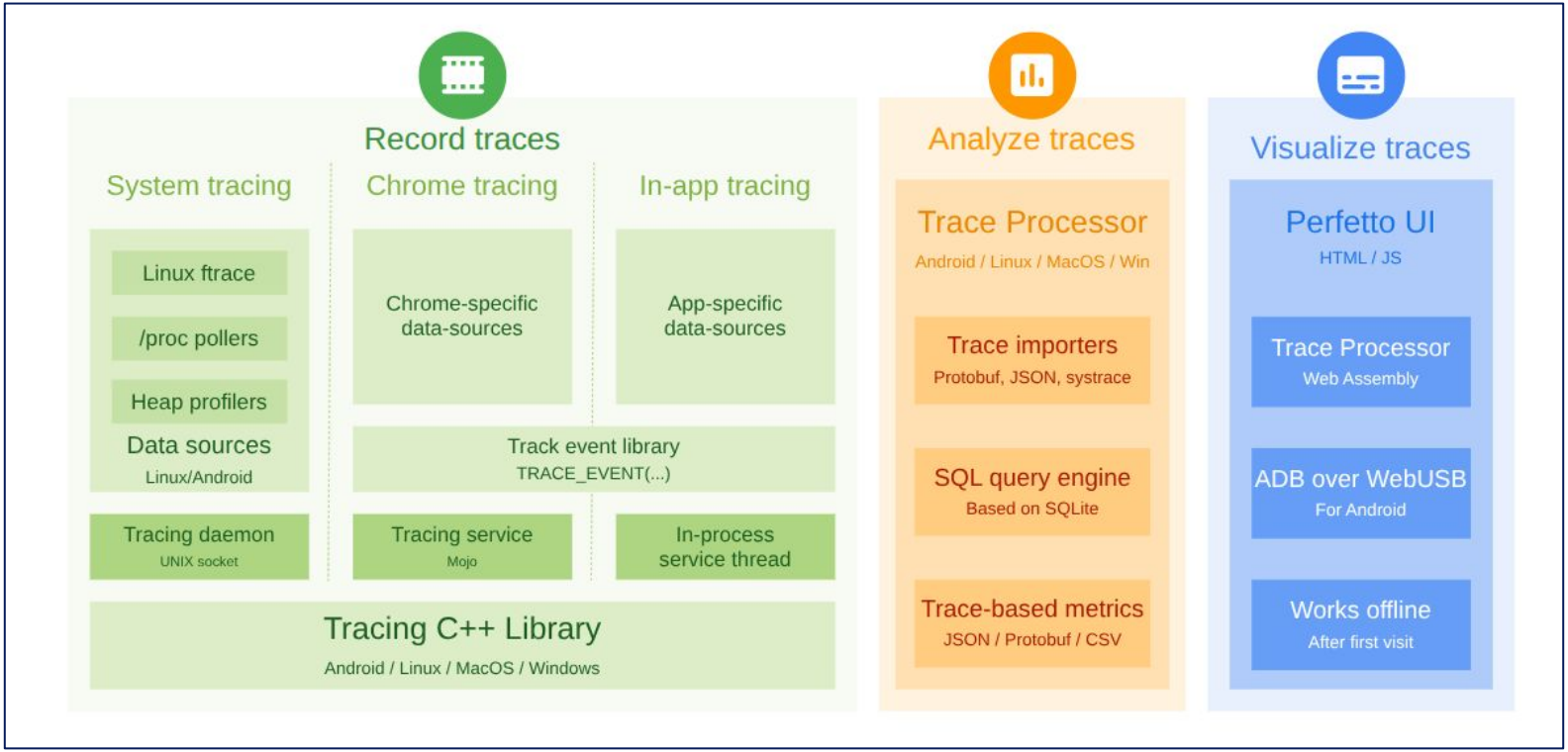
At least a bit!





https://perfetto.dev/docs/

Perfetto Overview



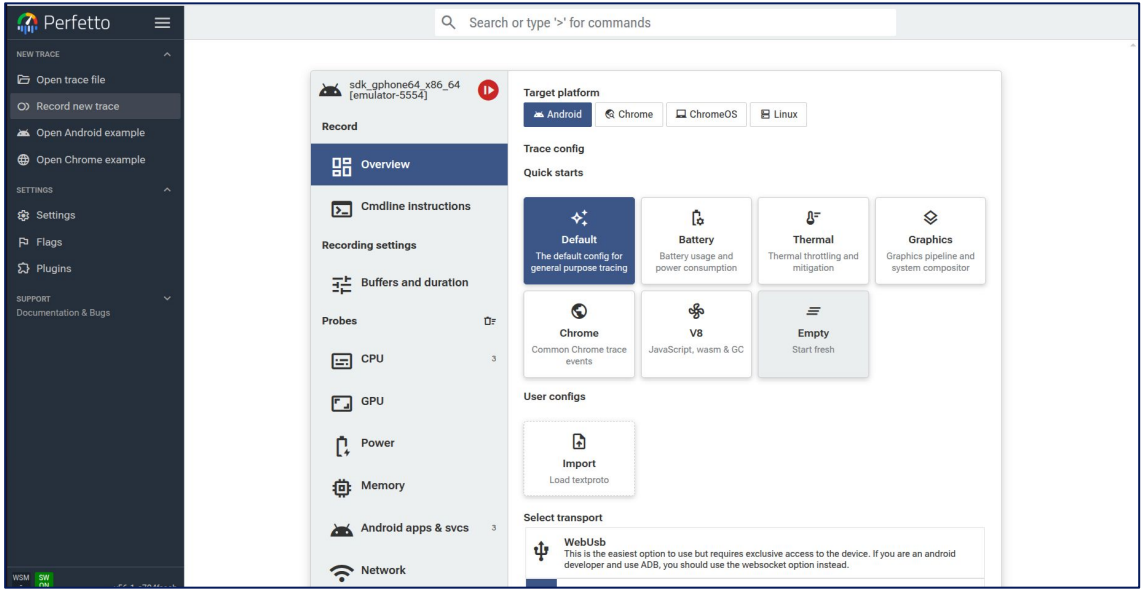
inovex

https://perfetto.dev/docs/getting-started/system-tracing

Tracing on Android

There is **plenty** of documentation and it's **well** integrated.

1) Start your Android device and connect via adb
2) Go to ui.perfetto.dev and select "Record a new Trace"
3) Use WebUSB or start "`./tracebox websocket_bridge`"



See also

- Performance: Perfetto Traceviewer - MAD Skills
- Debug Android Performance with Tracing & Perfetto

inovex

Tracing on Embedded Linux (e.g. Yocto)

There is **not plenty** of docu and it's **not well** integrated.

Therefore I have created an example and tutorial:

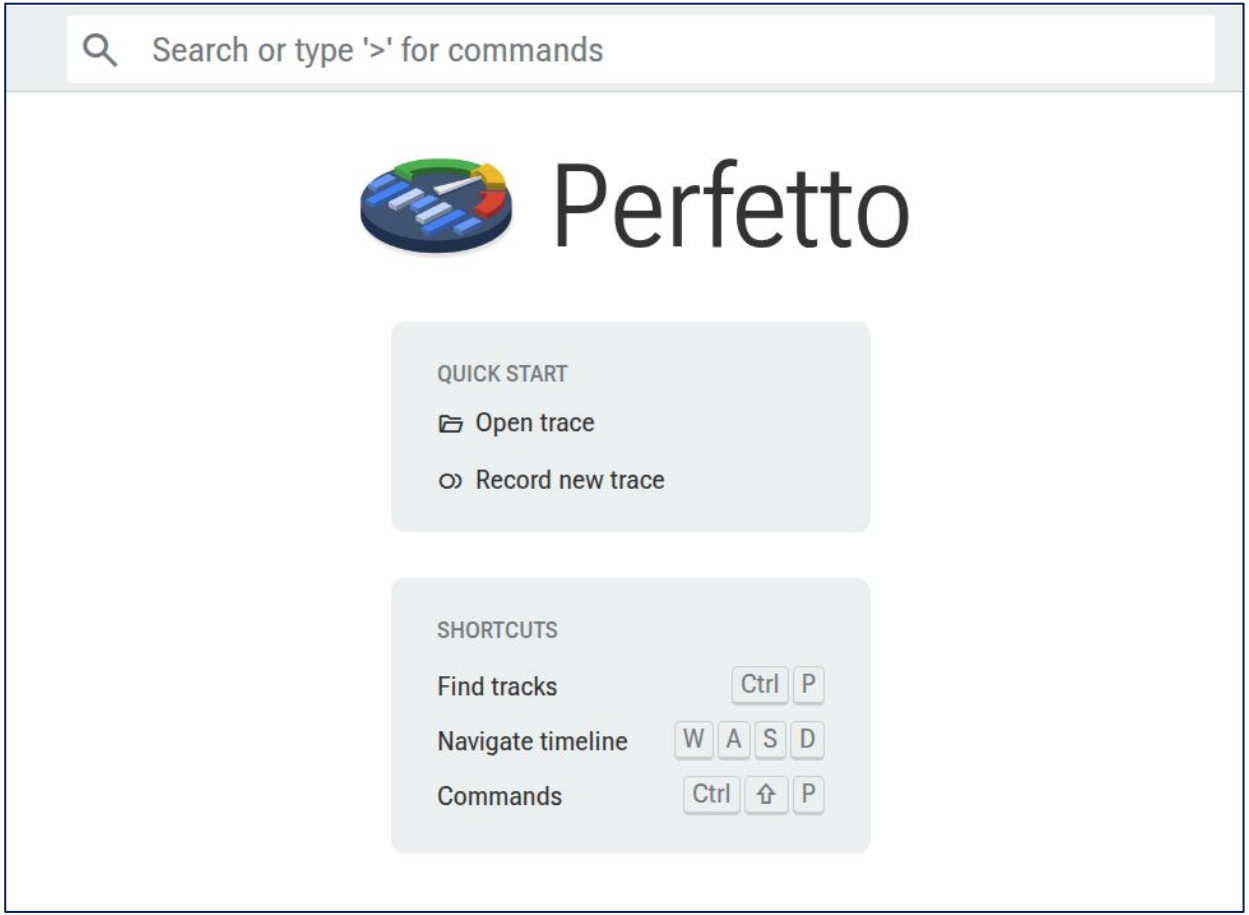
https://codeberg.org/lengfeld/perfetto-yocto-tutorial

inovex

Advanced usage (Tips & Tricks)

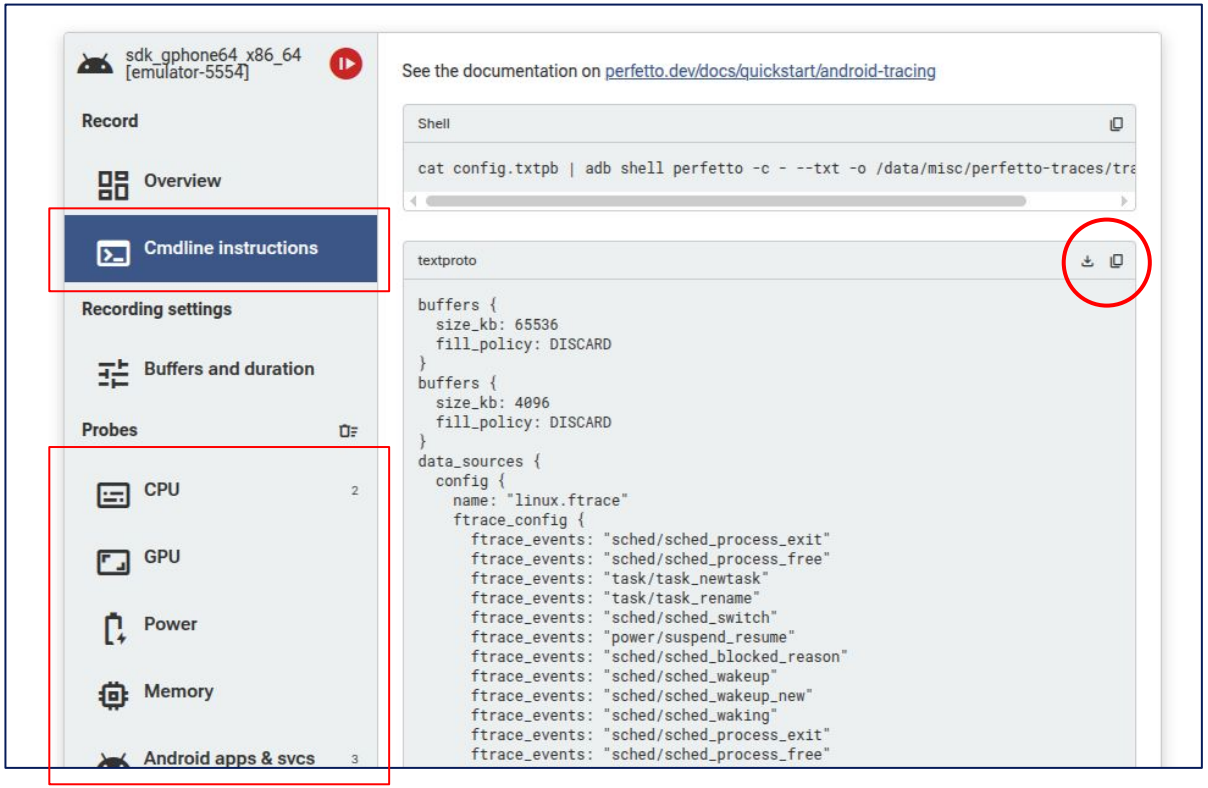
Basic Tips

- Gamer keys
- Search for
  - tracks
  - slices
  - logs



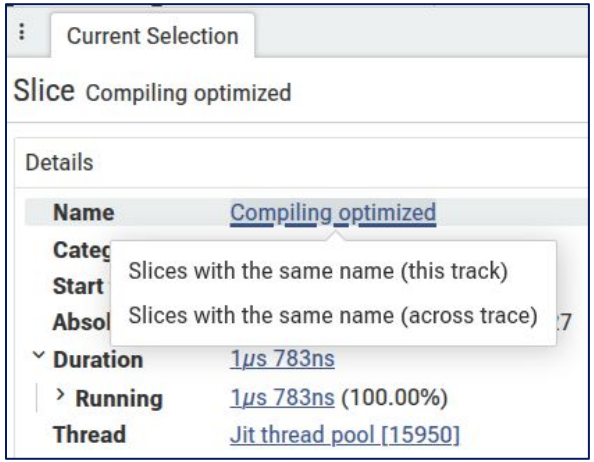
Use the TraceViewer as your config config manager

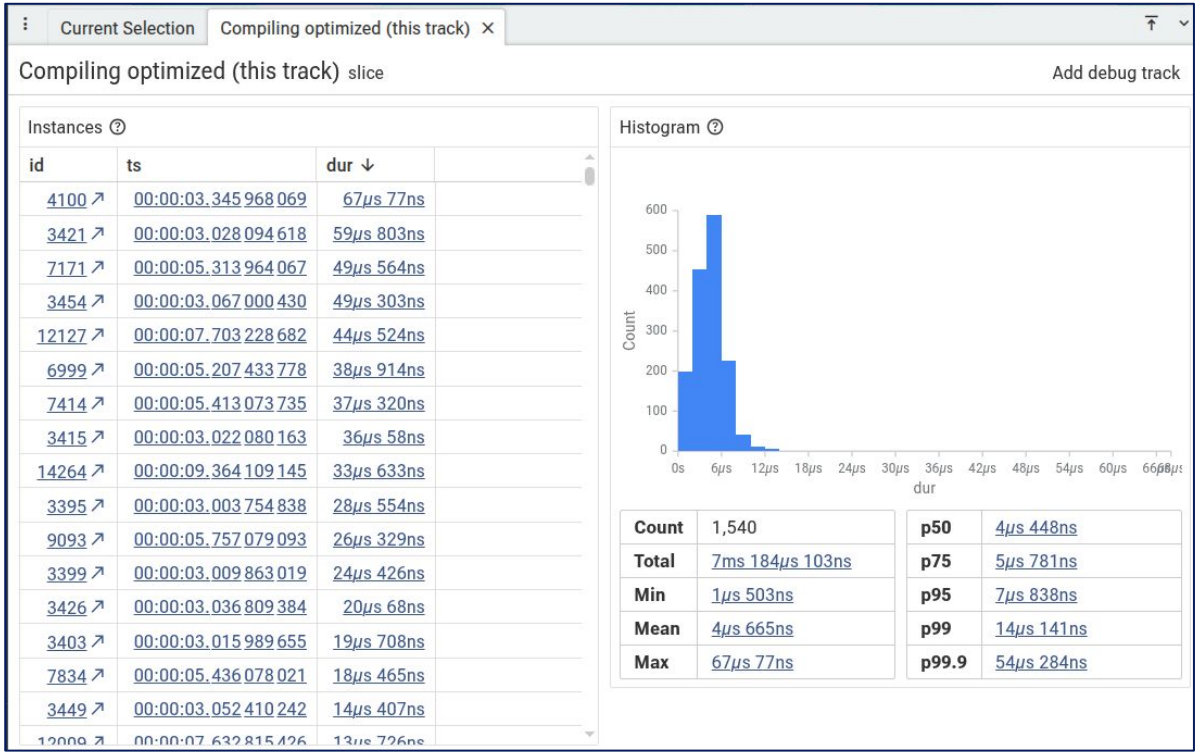
1) Explore the data sources in the Trace Viewer
2) Copy into a file for manual tracing



Built-in Histogram and statistics

1. Click on Slice
2. Then click on slice name
3. Select “with same name”

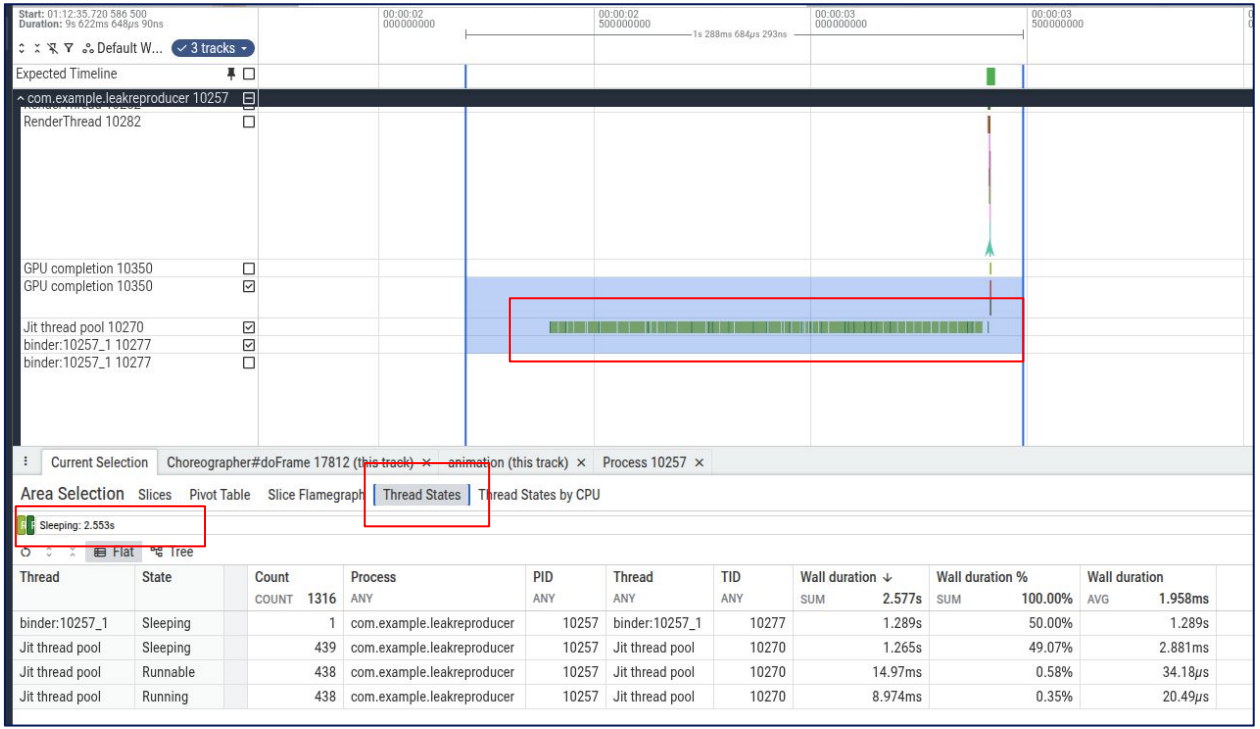




Thread states

1) Use rectangle select
2) Click on “Thread States”
3) Shows percentage for states (sleeping, runnable, running, …)

General advance:
Rectangle select does a lot of useful things!



CPU tracks

And rectangle select on CPU tracks

1) Select a region, e.g. all CPU tracks for a given time
2) Select “CPU by process” and sort

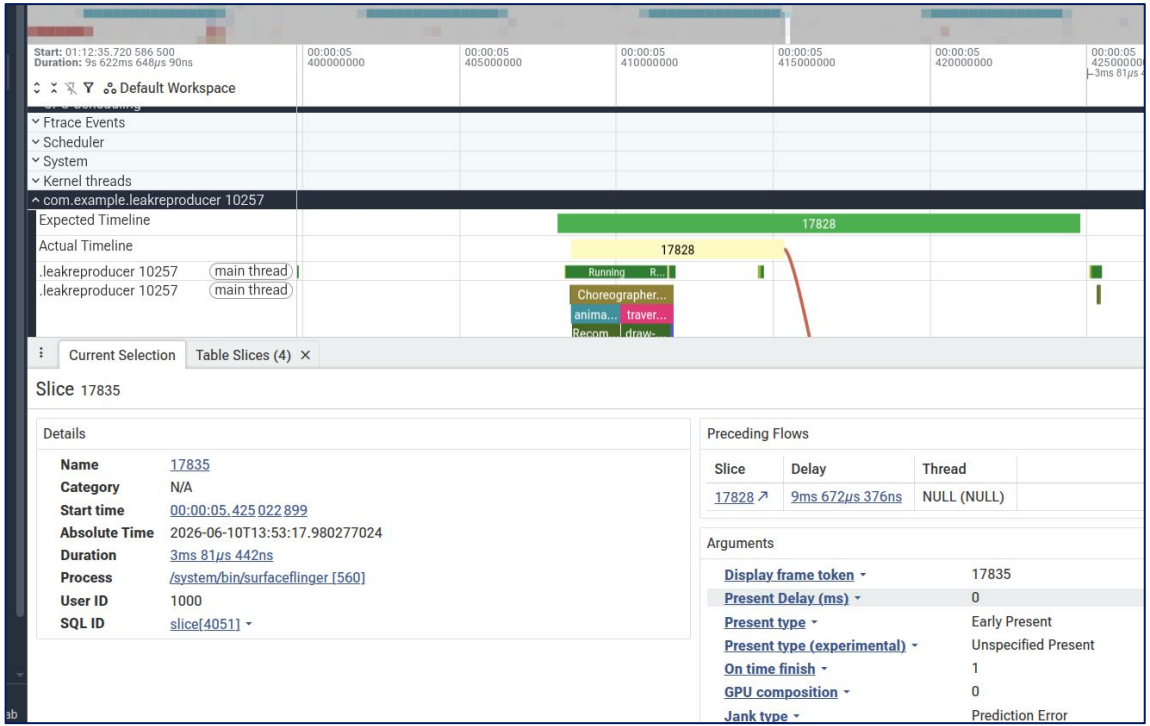
Android’s Frame Timeline

There is the

- Expected Timeline
- Actual Timeline

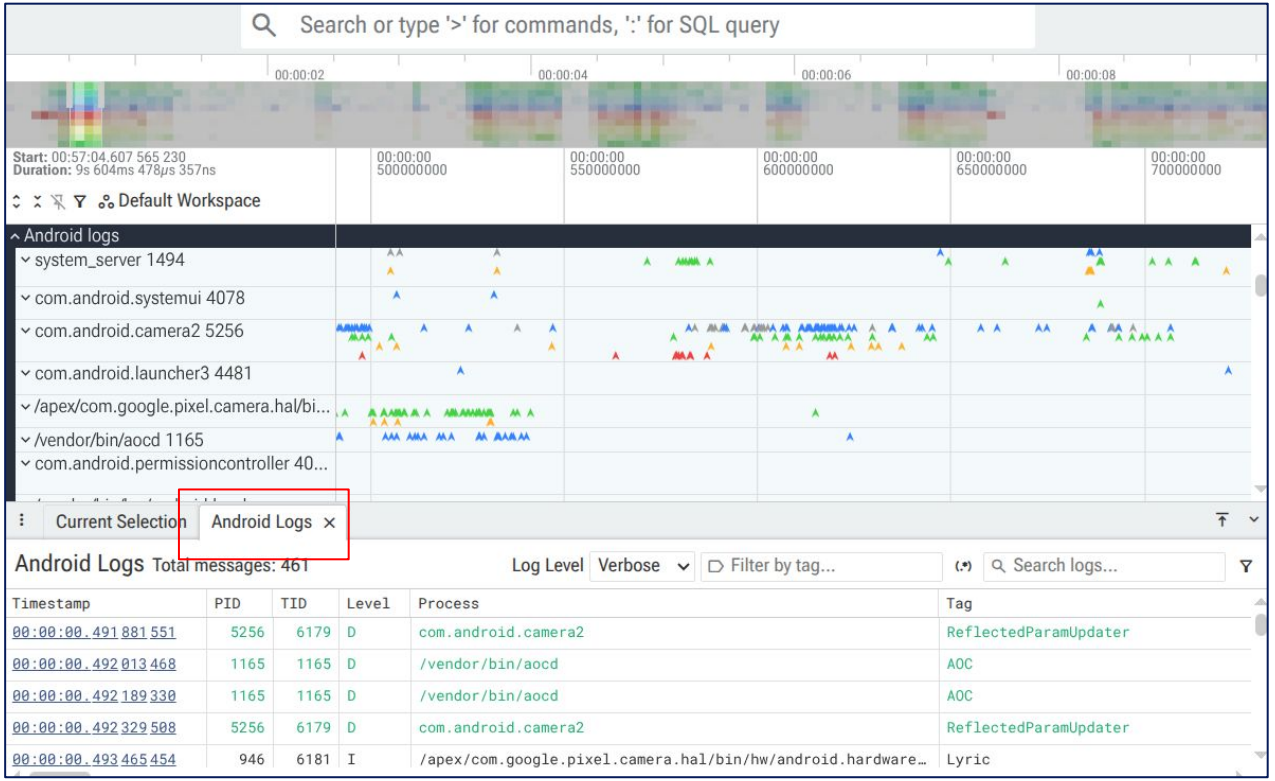
Every frame has additional information:

- jank type
- …



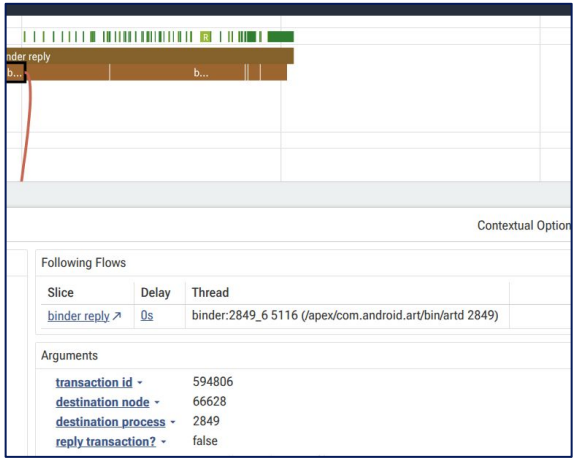
Android’s logcat

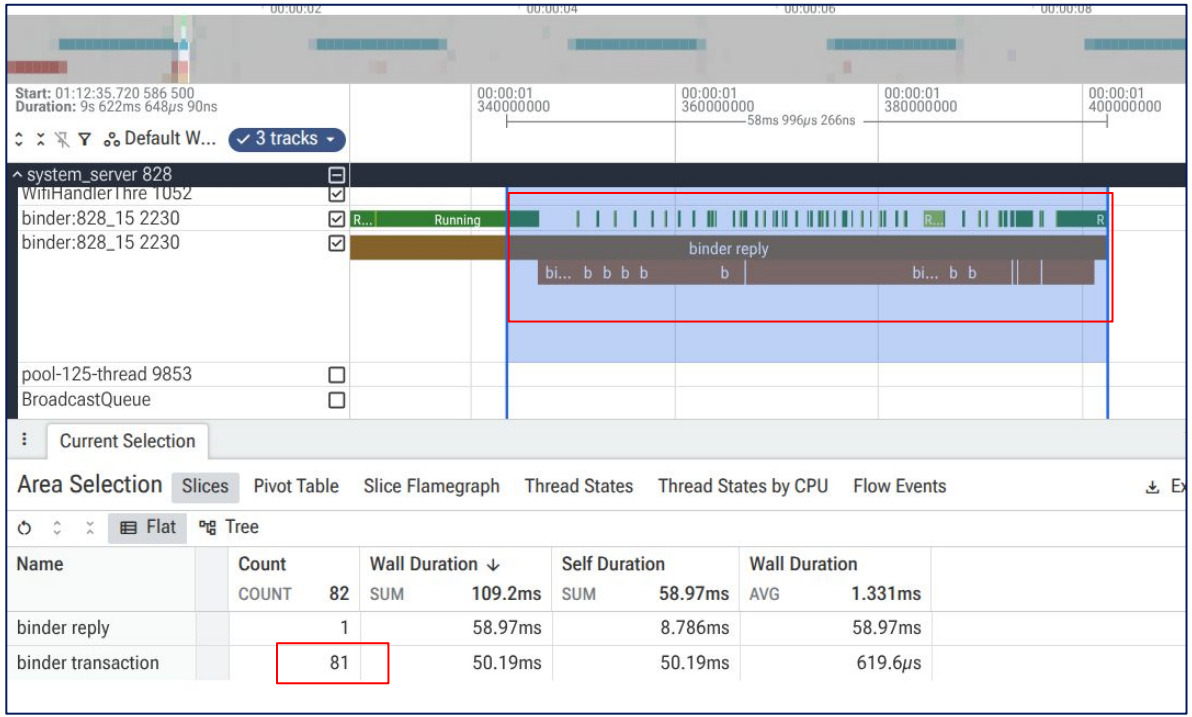
- Works only with “userdebug builds
- Logs shown in the tab are based on the current view and adapt when moving and zooming



Binder (and Flows)

- Click on slice for Flow information

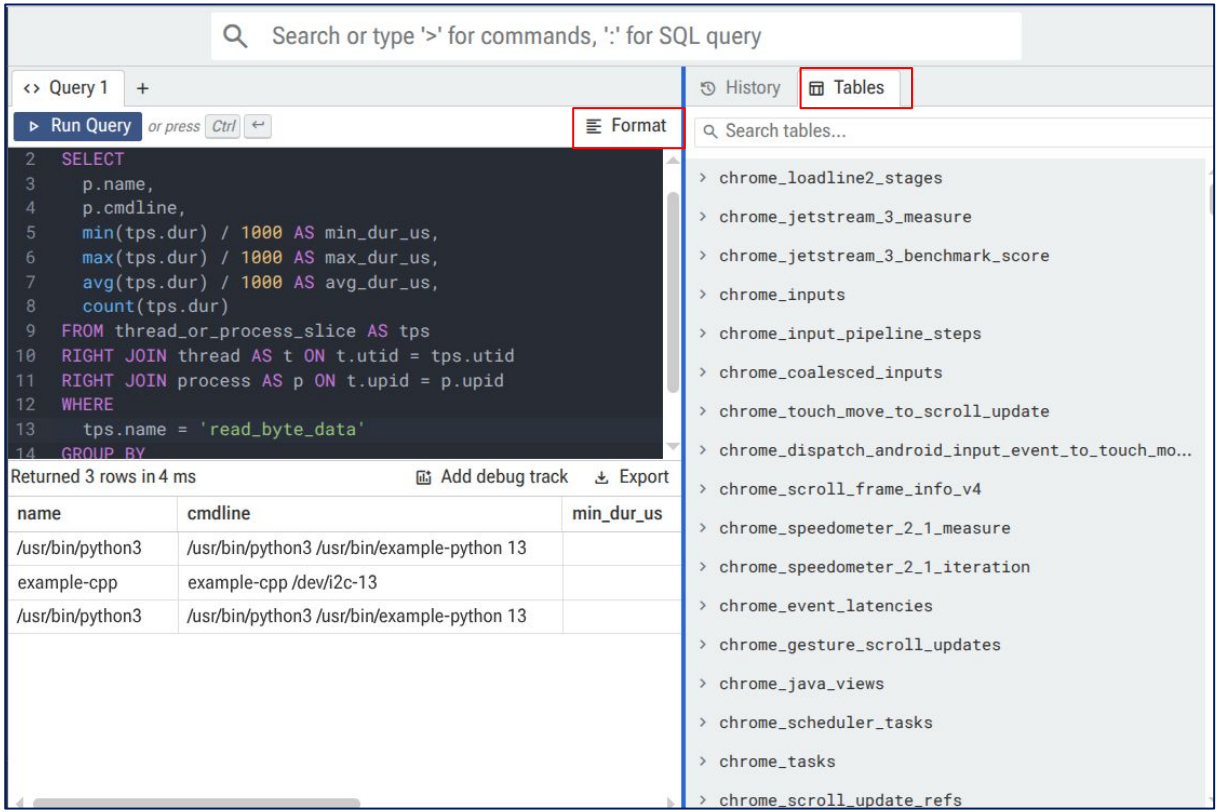




- Rectangle select to see statistics

Perfetto SQL

- Look out for “Copy SQL Query” in the context menus
- Use Tables tab as Quick Reference
- Use Format button

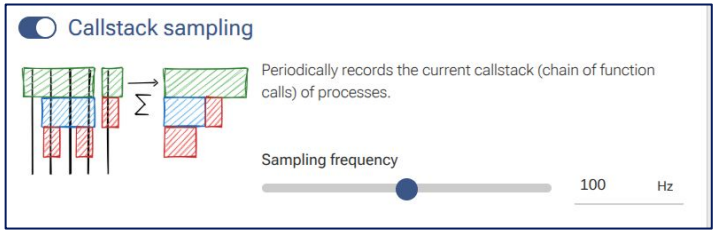


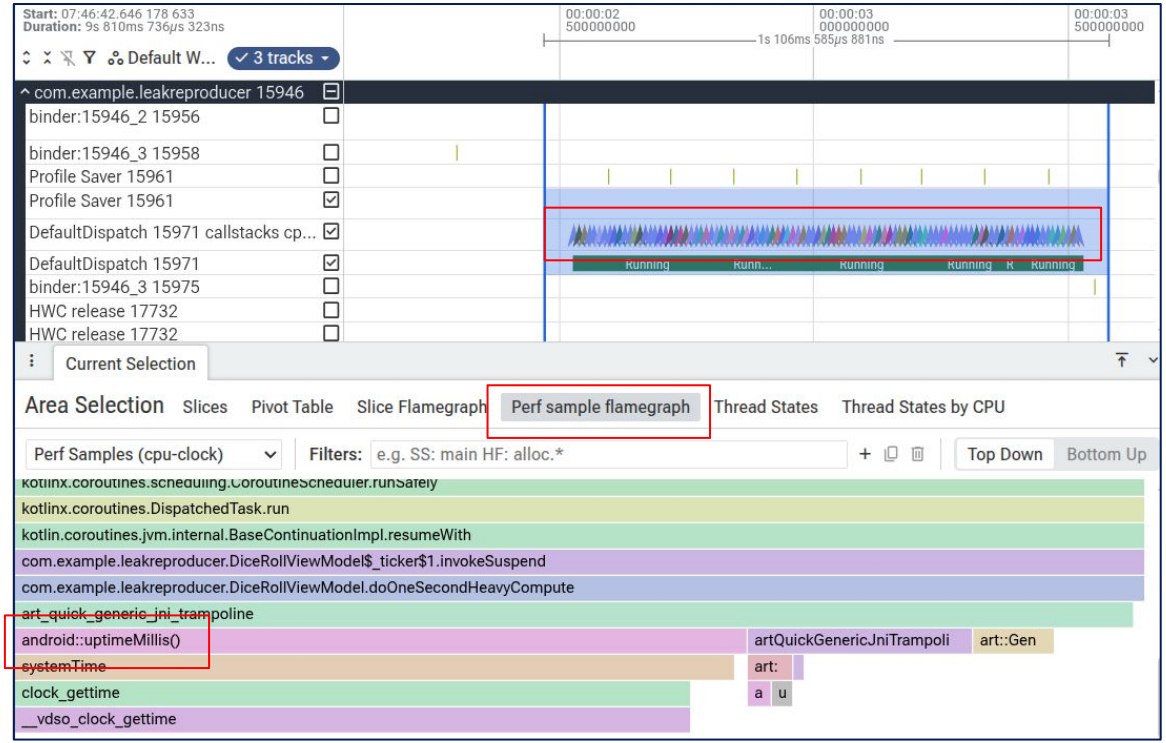
inovex

Gold

Stack Sampling

1) Enable stack sample in Trace Viewer UI
2) Either click on one trace sample or use rectangle select
3) See the flame graph at the bottom to pinpoint expensive functions





Release Notes (on Github and Google groups)

last week

perfetto-automation

v56.0

62048c1

Compare

Perfetto v56.0

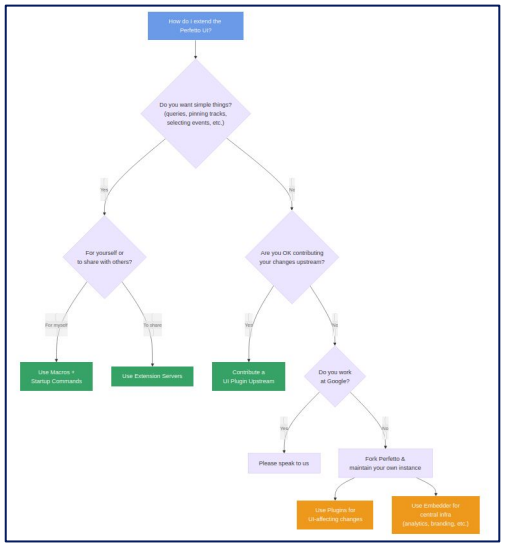
Perfetto v56.0 is out! This release adds more to the Heap Dump Explorer, a round of UI improvements, easier-to-read multi-machine traces, kernel-assisted stack sampling in `traced_perf`, and a few new SDK knobs.

🔥 Heap Dump Explorer: flamegraphs & bitmaps

The Heap Dump Explorer now has a built-in **flamegraph viewer**, so you can jump between the class/object/dominator views and a flamegraph without leaving the page. There's also a new **bitmap metadata page** for bitmaps captured in Android heap dumps, and a feature flag to control whether the explorer **auto-opens** on traces with heap-graph data.

Extending Perfetto

Extending the UI



Custom data sources

(when track events are not enough)

```
traceconv text example_custom_data_source.perfetto-trace
...
packet {
  trusted_uid: 0
  timestamp: 42
  trusted_packet_sequence_id: 2
  previous_packet_dropped: true
  for_testing {
    str: "Hello world!"
  }
}
...
```

inovex

Outlook

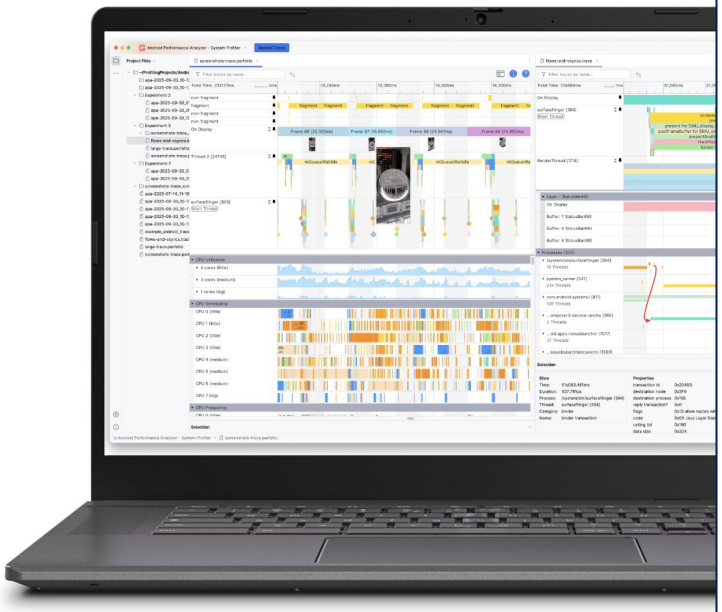
https://android-developers.googleblog.com/2026/05/introducing-android-performance-analyzer.html

New: Android Performance Analyzer (APA)

Android Performance Analyzer

System wide performance

- GPU, CPU, memory, battery
- Works for all major GPU's
- Built for Vulkan profiling
- Project, folders & tabs based workflow
- Native desktop app for all major OS
- Available as standalone app & in Android Studio



inovex

Nearly the end of my presentation

inovex

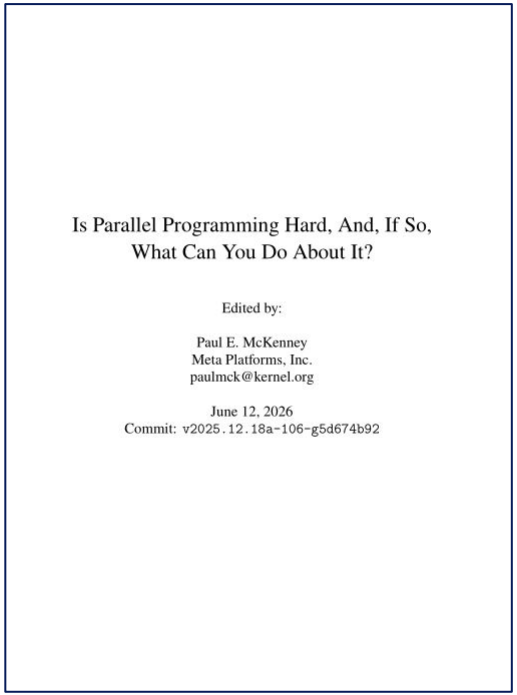
Thank You! Additional Reading.

And thanks to

- the Perfetto-Team at Google for building Perfetto,
- and all the people working on the Linux kernel performance framework!

Is Parallel Programming Hard, And, If So, What Can You Do About It?
(The perfbook, ~700 pages)

by Paul E. McKenney


Is Parallel Programming Hard, And, If So,
What Can You Do About It?

Edited by:

Paul E. McKenney
Meta Platforms, Inc.
paulmck@kernel.org

June 12, 2026
Commit: v2025.12.18a-106-g5d674b92

https://perfetto.dev/docs/contributing/getting-started#community

Perfetto is so cool. Try it out and get help!

Support by Perfetto devs:

- Mailinglist/Google Group
- Discord Server

Perfetto

Support by me:

- See my contact details
- Or join the discord on aosp-devs.org for AOSP/Android system tracing

Thank you!

Stefan Lengfeld

Perfetto Enthusiast by choice

stefan.lengfeld@inovex.de