



Storage Performance Matters


Benchmarking PostgreSQL on Kubernetes

Jonathan Battiato
Cloud Native PG contributor
DoK Community Ambassador
Napoli, 25/09/2025



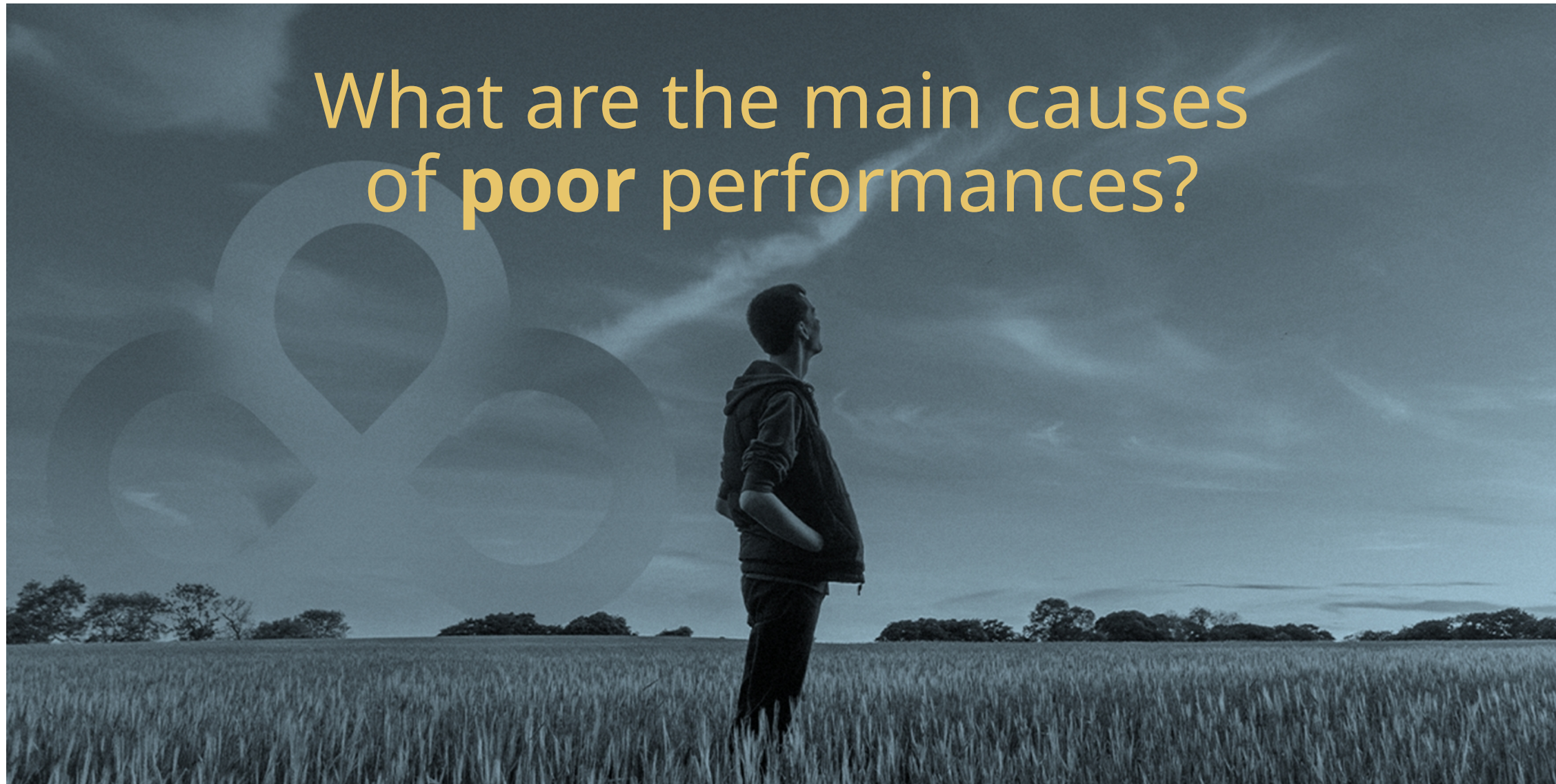


How many of you run PostgreSQL in **production**?

A person in a dark hoodie stands in a field of tall grass, looking up at a dramatic, cloudy sky. A large, faint Kubernetes logo is visible in the background. The text "How many of you run it in **Kubernetes**?" is overlaid in yellow.

How many of you run it
in **Kubernetes**?

What are the main causes of **poor** performances?



Missing
Indexes

Non-optimized
Queries

Wrong Hardware
Resources

Bad PostgreSQL
Settings

Faulty
Hardware

Let's suppose that you have already:

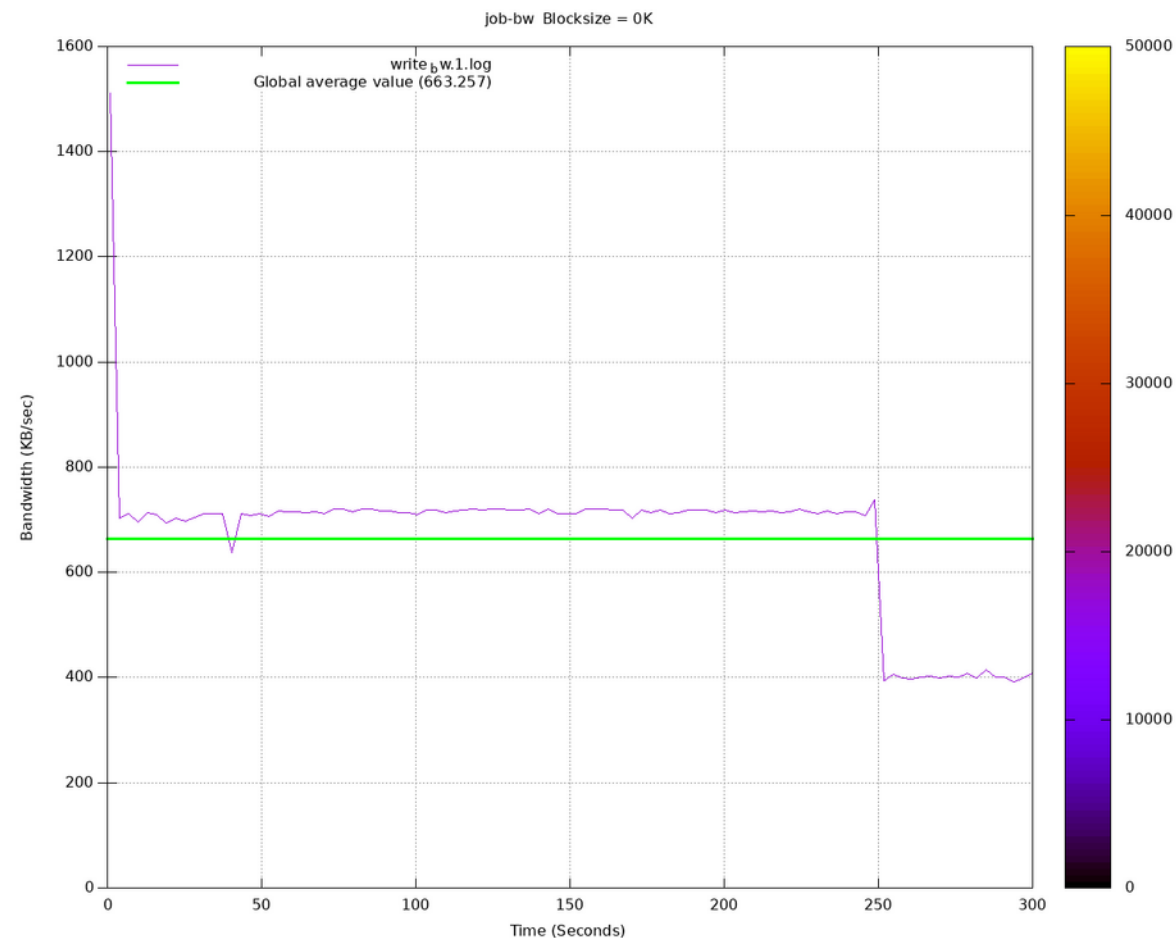
- Created the required indexes
- Improved the PostgreSQL memory settings
- Optimized your queries
- Given the right amount of hardware resources to the server



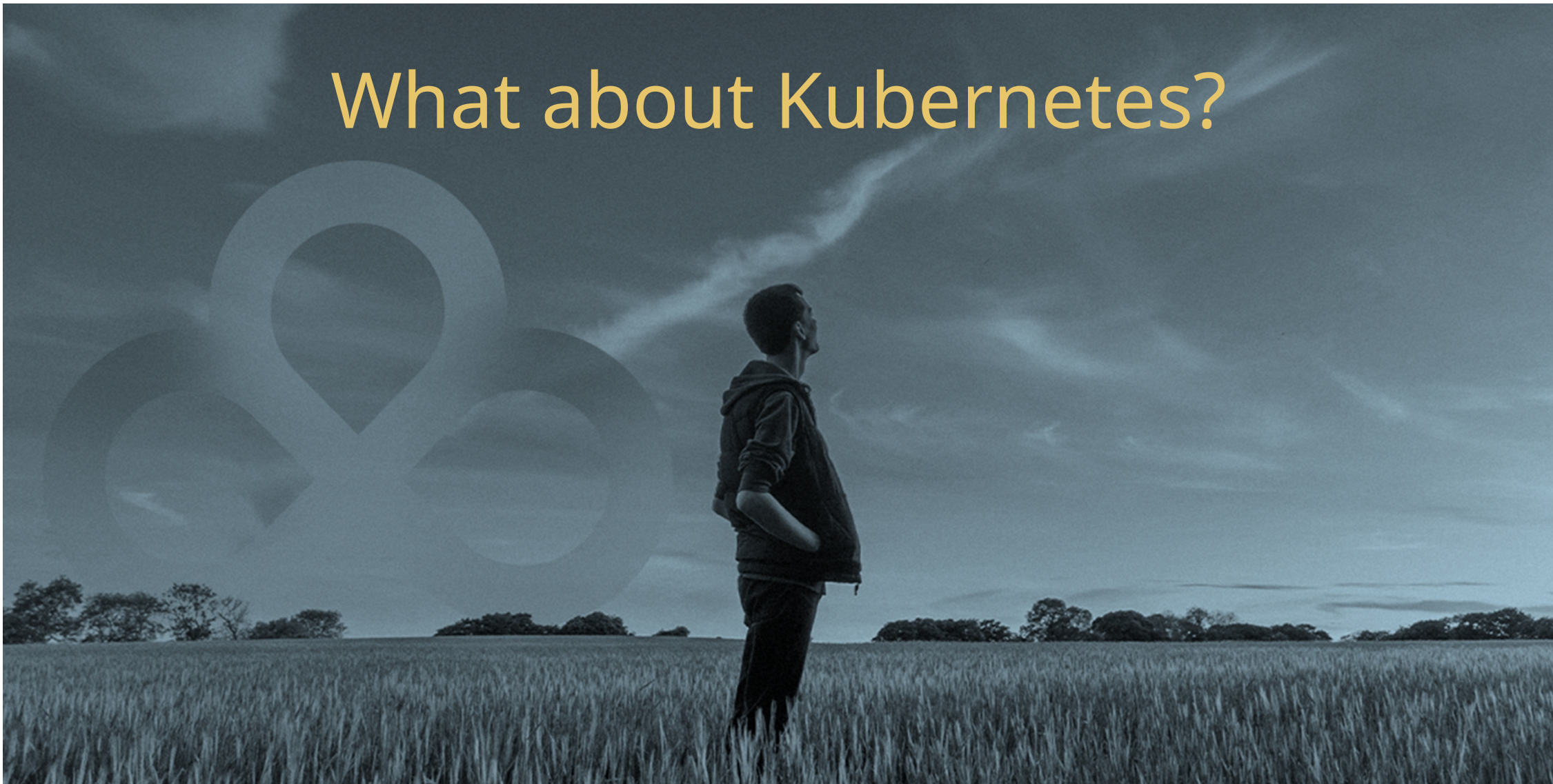
How do you check for **faulty** hardware or wrong hardware **configuration**?

Storage Benchmark Tests

1. Collect **baselines**
2. Validate hardware **configuration**
3. Discover **faulty** hardware



What about Kubernetes?

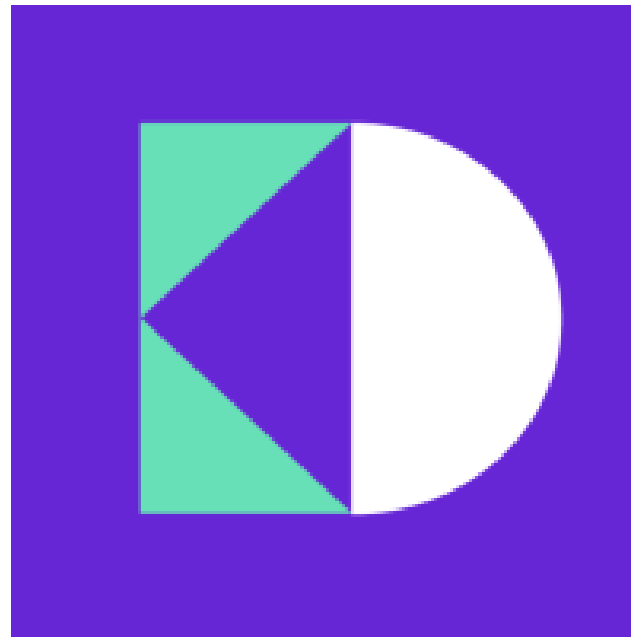


PostgreSQL doesn't care if it's running
on bare metal, virtual machines,
or Kubernetes:
if your storage is slow,
your database is slow.

The approach should be the same as
bare metal and VM

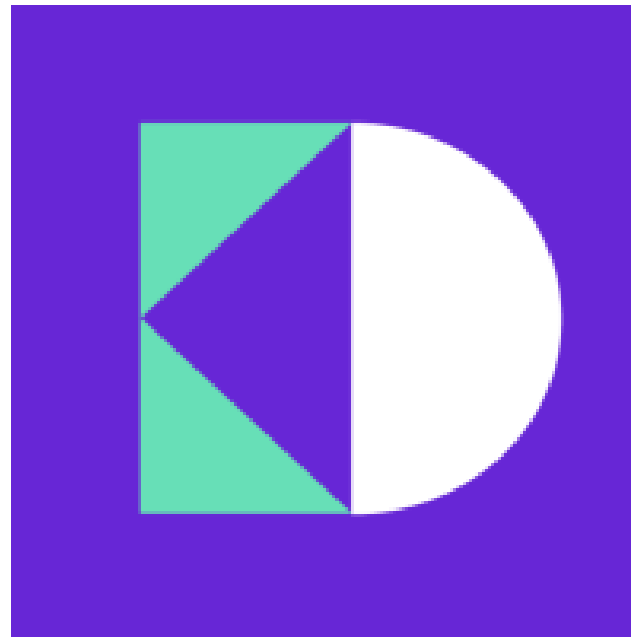
DoK Best Practice Guide

1. FIO
2. pgbench
3. pg_test_fsync



Follow the guide, collect results to:

1. gather baselines
2. get max throughput
3. compare with providers measurements





Monitoring

Implement monitoring system

1. Prometheus
2. Grafana

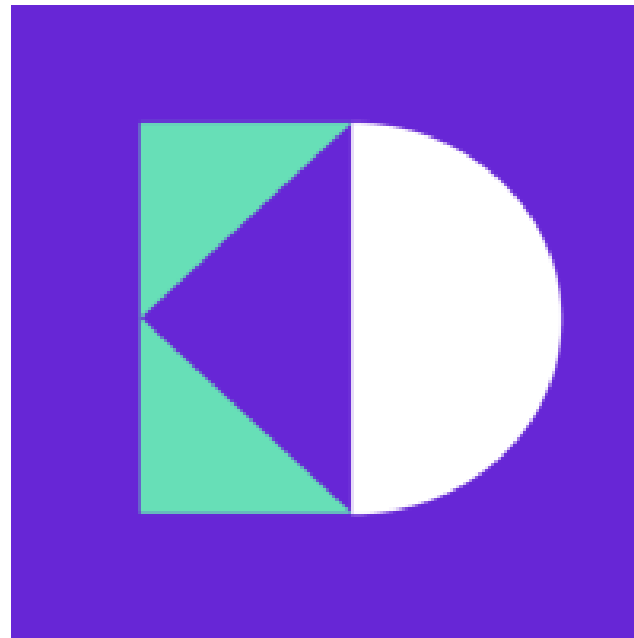
Collect metrics during benchmarks

- **History** of performances
- **Compare MB/s** instead of TPS

Call to action!

Don't wait for **production** to reveal the
bottleneck.

Benchmark **now**,
keep the results,
and use them to make **better decisions.**



Thank you!

