

Other languages: [繁體中文](#)

## Causal profiling with Coz

With Coz you can profile two things:

- throughput
- latency

If you want to profile throughput of your application, you should switch on the `COZ_PROFILING` cmake option and include debug information in your executable with `Debug` or `RelWithDebInfo` release modes in cmake. Doing so will include coz progress points when serving a request. Profiling latency is currently not supported in whole application scope, but can still be done in user code.

When you're done compiling you application with progress points included. You need to run the executable with the coz profiler, for example `coz run --- [path to your executable]`.

Lastly, the application needs to be stressed, for best results you need to stress all code paths and run the profile for a good amount of time, 15+ min.

The final profile will be a `profile.coz` file created in the current working directory. To view results, open the profile in the official [viewer](#), or you could run a local copy from the official [git repo](#).

Coz also supports scoping source files included for the profile with `--source-scope <pattern>` or `-s <pattern>` among other things, that should prove useful.

For more information checkout:

- `coz run --help`
- [Git repo](#)
- [Coz whitepaper](#)

## Next: [Brotli compression](#)

---