releases / rust / publishing / performance

Guidedog 1.0.0-beta.16

Guidedog contributors

2026-11-27

Abstract

Build the editions of a site at the same time, as many as fit in memory, with the same site and the same reports as before, and proofs checked by Lean.

This release builds the editions of a site at the same time. The Guidedog website, 23 editions with seven PDF books, builds in 13 seconds instead of about a minute, and publishes the same files.

`guidedog ssg build` used to build every mount in every language one after another, and most processors stayed idle. Now each edition is built by a `guidedog build` child process, as many at once as fit in nine tenths of the memory available and as the processors allow. An edition reserves the memory it used at its last build, with a quarter more; `_build/ssg-projects/memory.toml` keeps these peaks. An edition heavier than the room is built alone, as before, so a build never needs more memory than the sequential build needed. Linux, macOS and Windows report the memory available and each build's peak.

The site, the order of the reports and the first failure reported are those of the sequential build. `--jobs=N` limits the editions built at once, and `--jobs=1` builds them one after another in the `guidedog` process.

Every project build now locks its records folder, so two builds never write one output at once. A second build of the same output stops with `build.busy`.

GDS 0020 models the schedule and proves in Lean that the running editions stay within the room unless one runs alone, that every run ends, that the outcome is the sequential one, that two editions never write the same file, and that a catalog several editions write is read whole.

GDS 0020 states the model, the theorems and the measurements. The command reference describes `--jobs` and `--peak-memory-file`.