releases / rust / plugins / publishing

Guidedog 1.0.0-beta.15

Guidedog contributors

2026-11-23

Abstract

Fix a race that made a file lock look held when no command held it, with one lock for every command and proofs checked by Lean.

This release fixes a race in the file locks. A command could be told that the plugin registry, a publisher's state or a discussion catalog was busy when no command held its lock.

On Unix a file lock belongs to the open file, not to the program that took it. A child process holds a copy of every open file from its creation until it starts its program. A command that only closed its lock file had therefore not released the lock while such a copy lived. This showed when one thread changed the plugin registry while another started a plugin program. A holder now unlocks before it closes, which releases the lock for every copy at once.

One lock now serves every command: the plugin registry, the Pages publisher, discussion catalogs and both locks of a static site. Library users find it as `gd_plugins::files::lock_exclusive`. Problem codes, messages, exit statuses and lock file names are unchanged. A discussion command that meets a held lock reports it at once; it no longer asks again for 200 ms.

GDS 0019 models the lock, the takers, child processes and crashes. It proves mutual exclusion, that a free lock is never refused, recovery after a crash, and that readers see whole files and no update is lost. Lean 4.19 checks the proofs with its core library alone, and `cargo xtask proofs` runs the check. GDS 0019 states the model, the theorems and the evidence. The plugin internals show how a publisher uses the lock.