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.

Diese Version behebt eine Wettlaufsituation bei den Dateisperren. Einem Befehl konnte gemeldet werden, die Plugin-Registry, der Zustand eines Publishers oder ein Diskussionskatalog sei belegt, obwohl kein Befehl die Sperre hielt.

Unter Unix gehört eine Dateisperre zur geöffneten Datei, nicht zu dem Programm, das sie genommen hat. Ein Kindprozess hält von seiner Erzeugung an, bis er sein Programm startet, eine Kopie jeder geöffneten Datei. Ein Befehl, der seine Sperrdatei nur schloss, hatte die Sperre daher nicht freigegeben, solange eine solche Kopie existierte. Das zeigte sich, wenn ein Thread die Plugin-Registry änderte, während ein anderer ein Plugin-Programm startete. Ein Halter entsperrt jetzt, bevor er schließt; das gibt die Sperre für alle Kopien zugleich frei.

Eine einzige Sperre dient jetzt allen Befehlen: der Plugin-Registry, dem Pages-Publisher, den Diskussionskatalogen und beiden Sperren einer statischen Website. Wer die Bibliothek nutzt, findet sie als `gd_plugins::files::lock_exclusive`. Problemcodes, Meldungen, Exit-Status und die Namen der Sperrdateien sind unverändert. Ein Diskussionsbefehl, der auf eine gehaltene Sperre trifft, meldet sie sofort; er versucht es nicht mehr 200 ms lang erneut.

GDS 0019 modelliert die Sperre, die Prozesse, die sie nehmen, Kindprozesse und Abstürze. Es beweist den gegenseitigen Ausschluss, dass eine freie Sperre nie verweigert wird, die Wiederherstellung nach einem Absturz und dass Leser ganze Dateien sehen und keine Aktualisierung verloren geht. Lean 4.19 prüft die Beweise allein mit seiner Kernbibliothek, und `cargo xtask proofs` führt die Prüfung aus.

GDS 0019 nennt das Modell, die Sätze und die Belege. Die Plugin-Interna zeigen, wie ein Publisher die Sperre verwendet.