As the Kernel Turns: Rust in Linux saga reaches the “Linus in all-caps” phase

“Put another way: the ‘nobody is forced to deal with Rust’ does not imply ‘everybody is allowed to veto any Rust code.'” Maintainers might also find space in the middle, being aware of Rust bindings and working with Rust developers, but not actively involved, Torvalds writes.

“Why wouldn’t we do this?”

In an earlier response to the “Rust kernel policy” topic, Kroah-Hartman suggests that, “As someone who has seen almost EVERY kernel bugfix and security issue for the past 15+ years … I think I can speak on this topic.”

As the majority of bugs are due to “stupid little corner cases in C that are totally gone in Rust,” Koah-Hartman is “wanting to see Rust get into the kernel,” so focus can shift to more important bugs. While there are “30 million lines of C code that isn’t going anywhere any year soon,” new code and drivers written in Rust are “a win for all of us, why wouldn’t we do this?” After casting doubt on C++ as a viable long-term codebase, Kroah-Hartman clarifies the obvious point that Rust, while not a “silver bullet,” does a lot of things right, especially for developers trying to deal with the kernel’s tricky APIs.

“Yes, mixed language codebases are rough, and hard to maintain, but we are kernel developers dammit, we’ve been maintaining and strengthening Linux for longer than anyone

ever thought was going to be possible," Kroah-Hartman writes. "We've turned our development model into a well-oiled engineering marvel creating something that no one else has ever been able to accomplish. Adding another language really shouldn't be a problem, we've handled much worse things in the past and we shouldn't give up now on wanting to ensure that our project succeeds for the next 20+ years."

Rust may or may not become an ascendant language in the kernel. But maintaining C as the dominant language, to the point of actively tamping down even non-direct interaction with any C code, did not seem like a viable long-term strategy. Many discussions on the topic have noted the existence of Redox, a Rust-centered microkernel, or the theoretical but technically possible forking of Linux into a C-only project. But they are both just a smidge dismissive of how important the active development of Linux, the dominant infrastructure OS, is to the world.

https://arstechnica.com/gadgets/2025/02/linux-leaders-pave-a-path-for-rust-in-kernel-while-supporting-c-veterans/