Michael Goulet

✉ michael@errs.io — 🔗 linkedin.com/in/compiler-errors — ⚙ github.com/compiler-errors

Summary —

I am a Rust Compiler Engineer at Amazon Web Services and am a member of the Compiler and Types teams in the Rust project. I am a generalist in the Rust compiler codebase and split my time between feature development and compiler maintenance primarily in the front-end of the compiler (parsing, analysis, and the type system).

This résumé is a non-exhaustive list of my contributions, focusing primarily on my contributions to Rust that I think are noteworthy.

Experience

Amazon Web Services

Senior Rust Compiler Engineer   *July 2024 – present*
Rust Compiler Engineer   *October 2022 – July 2024*

– Hired to contribute full-time to maintenance and feature development for the Rust compiler, work reflected below.

Rust Project

Compiler Team   *March 2022 - present*
Types Team   *May 2022 - present*

– More than 2000 PRs reflecting widespread involvement in compiler maintenance and feature development.
– Reviewed and approved over 2600 PR to the compiler.

  User-facing feature implementation

– Async closures - enable users to write more expressive code with closures in async environments.
– Async functions in traits - allow users to finally write first-class async traits.
– Precise capturing - allowed us to migrate to edition 2024 `impl Trait` capturing rules (RFC 3498).
– Associated type bounds and Return type notation - allow users to express complex `where` clause bounds in more intuitive ways.
– Return type notation - allow users to express new API patterns in generic async code (ongoing stabilization).

  Ongoing project work

– Significantly contributed to the new trait solver refactoring initiative.
– Coordinated major refactorings of the type system internals into a library (type system librarification).

  Compiler sustaining and refactoring

– Fixed various type system unsoundness bugs including those involving `dyn Trait` types, supertraits, implied bounds, never types among others.
– Completely re-implemented several features in the compiler including negative coherence and const traits.
– Authored significant improvements for Rust error messages (including major overhauls for behavior and presentation).
– Fixed hundreds of compiler crashes (Internal Compiler Errors) and authored refactors to robustify the compiler.

Meta/Facebook

Production Engineer   *October 2019 – October 2022*

– *Video Infrastructure Team* - Coordinated vendoring and version management of `FFmpeg` in Facebook monorepo.
– *Storage Hardware Team* - Led testing and integration of new storage servers and participated in incident management for hard-drive related issues in the storage hardware fleet.
– *Internship 2018: Video Infrastructure* - Worked on testbed for visual match algorithm, used for for unsafe content detection.
– *Internship 2017: POSIX Storage* - Worked on improvements to GlusterFS, focusing on performance bottlenecking in multi-tenant shared Gluster volumes.

Education

Caltech

Bachelor of Science in Computer Science and English   *June 2019*

– Teaching assistant for "Introduction to Systems Programming", "Introduction to Operating Systems", "Introduction to Database Implementation", "Database Projects", and an interpreters course.
– Authored a English thesis titled *Caught in the Middle: Homosexual Guilt, Liminality, and the role of the 'Novel of Identification' in Post-World War, Pre-Stonewall America* as part of the requirements of the English double major.