

OCaml Rust

September 21st, 2020 Gabriel Scherer

Principal Investigators:

- Frédéric Bour
- Jacques-Henri Jourdan
- Guillaume Munch-Maccagnoni

How to write hybrid programs in OCaml and Rust?

OCaml FFI today

Low-level

OCaml exports a low-level FFI to C (and all languages that talk to C)

Calls low-level functions of the runtime:

- allocate and set all forms of OCaml values
- declare roots, maintain the GC invariants

Difficult

Basically you are doing the same work the compiler does for OCaml programs.
... with no static or dynamic checking of code correctness.

Efficient, flexible, but very easy to get wrong. Hard to debug segfaults.

Automated?

ocaml-ctypes: automated stub generation from OCaml-side description of low-level C types. (not in this talk)

for Rust

OCaml/C + C/Rust, or

ocaml-rs: Rust library exporting the C low-level API. (+ convenience layers for some boilerplate code generation, using traits)

Our proposal

1. Improve the OCaml runtime for safe operability with RAII systems ("result" reification)
2. Put CAMLroot in production. (a safer C library with runtime-time checking of many invariants)

3. Use the Rust type system to encode GC invariants (Stephen Dolan's caml-oxyde, Alan Jeffrey's Josephine)
4. Study language features for resource-safety in OCaml linear pointers/capabilites, linear types, asynchronous exceptions

Difficulties

(runtime, CAMLroot, caml-oxyde) would be good tasks for a research engineer.

But: nearly impossible to hire a research engineer on an INRIA contract. (2 years CDD with average pay, compared to well-paid CDI in the OCaml industry)