Rust for Morello: Always-On Memory Safety, Even in Unsafe Code (Artifact)

Sarah Harris
University of Kent, Canterbury, UK

Simon Cooksey
University of Kent, Canterbury, UK

Michael Vollmer
University of Kent, Canterbury UK

Mark Batty
University of Kent, Canterbury, UK

Abstract

2012 ACM Subject Classification Software and its engineering → Compilers; Software and its engineering → Software safety; Software and its engineering → Object oriented languages

Keywords and phrases Compilers, Rust, Memory Safety, CHERI

Digital Object Identifier 10.4230/DARTS.9.2.25

Funding This work has been supported by the EPSRC under the DSbD Software Ecosystem grant programme EP/X021173/1.

Acknowledgements This paper was greatly improved thanks to the responses of anonymous reviewers. We extend our thanks to Jessica Clarke for her invaluable help with CHERI LLVM.

Related Article Sarah Harris, Simon Cooksey, Michael Vollmer, and Mark Batty, "Rust for Morello: Always-On Memory Safety, Even in Unsafe Code", in 37th European Conference on Object-Oriented Programming (ECOOP 2023), LIPIcs, Vol. 263, pp. 39:1–39:27, 2023.
`https://doi.org/10.4230/LIPIcs.ECOOP.2023.39`

Related Conference 37th European Conference on Object-Oriented Programming (ECOOP 2023), July 17–21, 2023, Seattle, Washington, United States

Evaluation Policy The artifact has been evaluated as described in the ECOOP 2023 Call for Artifacts and the ACM Artifact Review and Badging Policy.

1 Scope

The artefact is a fork of the Rust compiler [3, 2], modified to target Morello [1].

2 Content

The artefact is packaged as a `.zip` of the source code.

3 Getting the artefact

The artefact endorsed by the Artifact Evaluation Committee is available free of charge on the Dagstuhl Research Online Publication Server (DROPS). In addition, the artefact is also available at: `https://github.com/kent-weak-memory/rust`.

4 Tested platforms

This compiler targets the Morello machine, and can be compiled on an X86 or ARM-based computer. We have tested on X86 Linux, and Aarch64 Darwin (M-series macOS).

5 License

The artefact is dual licensed as MIT and APACHE, in-line with the upstream Rust project.

6 MD5 sum of the artefact

7a2b2d938225fe7f7dbc9460629004cd

7 Size of the artefact

44 M

References

1 Arm. *Arm® Architecture Reference Manual Supplement Morello for A-profile Architecture*. Arm, 2020.

2 Steve Klabnik, Carol Nichols, et al. *The Rust Programming Language*. The Rust Project Developers, 2021. URL: `https://doc.rust-lang.org/1.55.0/book/`.

3 Nicholas D. Matsakis and Felix S. Klock. The rust language. *Ada Lett.*, 34(3):103–104, October 2014. `doi:10.1145/2692956.2663188`.