Using capabilities for strict runtime invariant checking

Isaac Oscar Gariano[1][0000−0002−4881−0999],
Marco Servetto[1][0000−0003−1458−2868], and Alex Potanin[2][0000−0002−4242−2725]

[1] Victoria University of Wellington, Kelburn, 6012, Wellington, New Zealand
{isaac, marco.servetto}@ecs.vuw.ac.nz
[2] The Australian National University, Canberra, 2600, ACT, Australia
alex.potanin@anu.edu.au

Abstract. In this paper we use pre-existing language support for both reference and object capabilities to enable sound runtime verification of representation invariants. Our invariant protocol is stricter than the other protocols, since it guarantees that invariants hold for all objects involved in execution. Any language already offering appropriate support for reference and object capabilities can support our invariant protocol with minimal added complexity. In our protocol, invariants are simply specified as methods whose execution is statically guaranteed to be deterministic and to not access any externally mutable state. We formalise our approach and prove that our protocol is sound, in the context of a language supporting mutation, dynamic dispatch, exceptions, and non-deterministic I/O. We present case studies showing that our system requires a lighter annotation burden compared to Spec#, and performs orders of magnitude less runtime invariant checks compared to the 'visible state semantics' protocols of D and Eiffel.

Keywords: reference capabilities· object capabilities· runtime verification· class invariants.

1 Journal first submission of Using capabilities for strict runtime invariant checking

Our paper has been published in Science of Computer Programming Volume 224, 1 December 2022, 102878, and it can be easily retreived in open access on the following url and DOI:

https://www.sciencedirect.com/science/article/pii/S0167642322001113
https://doi.org/10.1016/j.scico.2022.102878.

The authors of our work are Isaac Oscar Gariano, Marco Servetto and Alex Potanin.

On Fri, 19 May 2023, we received an email titled *"Invitation to QUATIC 2023 Journal-First Track"* from *"Dr. Andrea Janes"*. The email stats that *"The chairs of the tracks Verification, Validation, and Testing, Quality Requirements; ICT*

Process Improvement; Organization, and Governance, Quality Aspects of Digital Transformation; and Quality Aspects of Human-Factors in Software Engineering suggested your paper Using capabilities for strict runtime invariant checking for the journal-first track."

We think this shows that our work is relevant for QUATIC 2023. As you can see from our abstract, our work is clearly well suited for the track *"Verification, Validation, and Testing"*. We consider our work a multidisciplinary approach that merges aspects of Runtime Verification, Static verification and type systems. We think that our work could facilitate important discussions between the members of those three similar but distinct research areas.