

39th European Conference on Object-Oriented Programming

ECOOP 2025, June 30–July 2, 2025, Bergen, Norway

Edited by

Jonathan Aldrich

Alexandra Silva

Part 1 of 2



Editors

Jonathan Aldrich 

Carnegie Mellon University, Pittsburgh, PA, USA
jonathan.aldrich@cs.cmu.edu

Alexandra Silva 

Cornell University, Ithaca, NY, USA
alexandra.silva@cornell.edu

ACM Classification 2012

Software and its engineering → General programming languages

ISBN 978-3-95977-373-7

PRINT ISBN: 979-8-3313-2289-2

Published online and open access by

Schloss Dagstuhl – Leibniz-Zentrum für Informatik GmbH, Dagstuhl Publishing, Saarbrücken/Wadern, Germany. Online available at <https://www.dagstuhl.de/dagpub/978-3-95977-373-7>.

Publication date

June, 2025

Bibliographic information published by the Deutsche Nationalbibliothek

The Deutsche Nationalbibliothek lists this publication in the Deutsche Nationalbibliografie; detailed bibliographic data are available in the Internet at <https://portal.dnb.de>.

License

This work is licensed under a Creative Commons Attribution 4.0 International license (CC-BY 4.0):
<https://creativecommons.org/licenses/by/4.0/legalcode>.



In brief, this license authorizes each and everybody to share (to copy, distribute and transmit) the work under the following conditions, without impairing or restricting the authors' moral rights:

- Attribution: The work must be attributed to its authors.

The copyright is retained by the corresponding authors.

Digital Object Identifier: 10.4230/LIPIcs.ECOOP.2025.0

ISBN 978-3-95977-373-7

ISSN 1868-8969

<https://www.dagstuhl.de/lipics>

■ Contents

Preface	
<i>Jonathan Aldrich and Alexandra Silva</i>	0:ix–0:x
Message from the Artifact Evaluation Chairs	
<i>Karine Even-Mendoza and Raphaël Monat</i>	0:xi
Message from the AITO President	
<i>Davide Ancona</i>	0:xiii
List of Authors	
.....	0:xv–0:xviii

Regular Papers

A Theory of (Linear-Time) Timed Monitors	
<i>Mouloud Amara, Giovanni Bernardi, Mohammed Aristide Foughali, and Adrian Francalanza</i>	1:1–1:30
The Algebra of Patterns	
<i>David Binder and Lean Ermantraut</i>	2:1–2:28
Pydrofoil: Accelerating Sail-Based Instruction Set Simulators	
<i>Carl Friedrich Bolz-Tereick, Luke Panayi, Ferdia McKeogh, Tom Spink, and Martin Berger</i>	3:1–3:31
Ensuring Convergence and Invariants Without Coordination	
<i>Dina Borrego, Nuno Preguiça, Elisa Gonzalez Boix, and Carla Ferreira</i>	4:1–4:29
Compositional Bug Detection for Internally Unsafe Libraries: A Logical Approach to Type Unsoundness	
<i>Pedro Carrott, Sacha-Élie Ayoun, and Azalea Raad</i>	5:1–5:28
Detecting Functionality-Specific Vulnerabilities via Retrieving Individual Functionality-Equivalent APIs in Open-Source Repositories	
<i>Tianyu Chen, Zeyu Wang, Lin Li, Ding Li, Zongyang Li, Xiaoning Chang, Pan Bian, Guangtai Liang, Qianxiang Wang, and Tao Xie</i>	6:1–6:27
Monadic Type-And-Effect Soundness	
<i>Francesco Dagnino, Paola Giannini, and Elena Zucca</i>	7:1–7:31
An Effectful Object Calculus	
<i>Francesco Dagnino, Paola Giannini, and Elena Zucca</i>	8:1–8:30
Event Race Detection for Node.js Using Delay Injections	
<i>Andre Takeshi Endo and Anders Møller</i>	9:1–9:28
Bottom-Up Synthesis of Memory Mutations with Separation Logic	
<i>Kasra Ferdowsi and Hila Peleg</i>	10:1–10:32
Contract Usage and Evolution in Android Mobile Applications	
<i>David R. Ferreira, Alexandra Mendes, João F. Ferreira, and Carolina Carreira</i> ..	11:1–11:30

Automatic Goal Clone Detection in Rocq <i>Ali Ghanbari</i>	12:1–12:19
FuzzFlesh: Randomised Testing of Decompilers via Control Flow Graph-Based Program Generation <i>Amber Gorzynski and Alastair F. Donaldson</i>	13:1–13:26
IsaBIL: A Framework for Verifying (In)correctness of Binaries in Isabelle/HOL <i>Matt Griffin, Brijesh Dongol, and Azalea Raad</i>	14:1–14:30
Spejion: Implicit and Non-Lexical Regions with Sized Allocations <i>Jack Hughes, Michael Vollmer, and Mark Batty</i>	15:1–15:26
A Lightweight Method for Generating Multi-Tier JIT Compilation Virtual Machine in a Meta-Tracing Compiler Framework <i>Yusuke Izawa, Hidehiko Masuhara, and Carl Friedrich Bolz-Tereick</i>	16:1–16:29
Contrasting Deadlock-Free Session Processes <i>Juan C. Jaramillo and Jorge A. Pérez</i>	17:1–17:29
Practical Type-Based Taint Checking and Inference <i>Nima Karimipour, Kanak Das, Manu Sridharan, and Behnaz Hassanshahi</i>	18:1–18:25
Profile-Guided Field Externalization in an Ahead-Of-Time Compiler <i>Sebastian Kloibhofer, Lukas Makor[1], Peter Hofer, David Leopoldseder, and Hanspeter Mössenböck</i>	19:1–19:32
Incremental Computing by Differential Execution <i>Prashant Kumar, André Pacak, and Sebastian Erdweg</i>	20:1–20:24
GSOHC: Global Synchronization Optimization in Heterogeneous Computing <i>Soumik Kumar Basu and Jyothi Vedurada</i>	21:1–21:30
Quantifying Cache Side-Channel Leakage by Refining Set-Based Abstractions <i>Jacqueline L. Mitchell and Chao Wang</i>	22:1–22:28
Wastrumentation: Portable WebAssembly Dynamic Analysis with Support for Intercession <i>Ääron Munsters, Angel Luis Scull Pupo, and Elisa Gonzalez Boix</i>	23:1–23:29
Fair Termination of Asynchronous Binary Sessions <i>Luca Padovani and Gianluigi Zavattaro</i>	24:1–24:29
Reusing Highly Optimized IR in Dynamic Compilation <i>Andrej Pečimúth, David Leopoldseder, and Petr Tůma</i>	25:1–25:25
Lightweight Diagramming for Lightweight Formal Methods: A Grounded Language Design <i>Siddhartha Prasad, Ben Greenman, Tim Nelson, and Shriram Krishnamurthi</i>	26:1–26:29
PoTo: A Hybrid Andersen’s Points-To Analysis for Python <i>Ingkarat Rak-amnouykit, Ana Milanova, Guillaume Baudart, Martin Hürzel, and Julian Dolby</i>	27:1–27:29
Reusing Caches and Invariants for Efficient and Sound Incremental Static Analysis <i>Mamy Razafintsialonina, David Bühler, Antoine Miné, Valentin Perrelle, and Julien Signoles</i>	28:1–28:29

Declarative Dynamic Object Reclassification <i>Riccardo Sieve, Eduard Kamburjan, Ferruccio Damiani, and Einar Broch Johnsen</i>	29:1–29:31
In-Memory Object Graph Stores <i>Aditya Thimmaiah, Zijian Yi, Joseph Kenis, Christopher J Rossbach, and Milos Gligoric</i>	30:1–30:30
Multiparty Asynchronous Session Types: A Mechanised Proof of Subject Reduction <i>Dawit Tirore, Jesper Bengtson, and Marco Carbone</i>	31:1–31:30
Compositional Static Value Analysis for Higher-Order Numerical Programs <i>Milla Valmet, Raphaël Monat, and Antoine Miné</i>	32:1–32:29
Mono Types – First-Class Containers for Datalog <i>Runqing Xu, David Klopp, and Sebastian Erdweg</i>	33:1–33:26
Taming and Dissecting Recursions Through Interprocedural Weak Topological Ordering <i>Jiawei Yang, Xiao Cheng, Bor-Yuh Evan Chang, Xiapu Luo, and Yulei Sui</i>	34:1–34:31
Chain of Grounded Objectives: Concise Goal-Oriented Prompting for Code Generation <i>Sangyeop Yeo, Seung-Won Hwang, and Yu-Seung Ma</i>	35:1–35:25
Efficient Neural Network Verification via Order Leading Exploration of Branch-and-Bound Trees <i>Guangqin Zhang, Kota Fukuda, Zhenya Zhang, H.M.N. Dilum Bandara, Shiping Chen, Jianjun Zhao, and Yulei Sui</i>	36:1–36:29

Experience Papers

RacerF: Lightweight Static Data Race Detection for C Code <i>Tomáš Dacík and Tomáš Vojnar</i>	37:1–37:19
Type-Safe and Portable Support for Packed Data <i>Arthur Jamet and Michael Vollmer</i>	38:1–38:19
WebGlitch: A Randomised Testing Tool for the WebGPU API <i>Matthew K. L. Wong and Alastair F. Donaldson</i>	39:1–39:26

Replication Paper

Scaling Up: Revisiting Mining Android Sandboxes at Scale for Malware Classification <i>Francisco Handrick Tomaz da Costa, Ismael Medeiros, Leandro Oliveira, João Calássio, Rodrigo Bonifácio, Krishna Narasimhan, Mira Mezini, and Márcio Ribeiro</i>	40:1–40:26
---	------------

Pearl/Brave New Ideas

Shouting at Memory: Where Did My Write Go? <i>Vasileios Klimis</i>	41:1–41:26
Contract Systems Need Domain-Specific Notations <i>Cameron Moy, Ryan Jung, and Matthias Felleisen</i>	42:1–42:24