Alex Potanin

School of Computing
The Australian National University
Canberra, ACT 2601
alex.potanin@anu.edu.au
comp.anu.edu.au/people/alex-potanin

Introduction

A/Prof Potanin has worked in the area of programming language design, type systems, and security for 20 years. The Rust programming language was inspired by the decades of work on ownership and immutability that was the focus of Potanin's research for the first 10 years of his career. More modern approaches work on securing software components using capabilities both at hardware and software level - this has been Potanin's research for the last 10 years of his career. The Wyvern programming language developed by Potanin after spending a sabbatical at Carnegie Mellon University in 2013 lays the foundation for usable yet secure programming language designs.

Having been working extensively in the programming language community (and being the General Chair in 2022 for SPLASH/OOPSLA and the Review Committee Chair in 2024 for OOPSLA) he is ideally placed to put together a team of both Australian-based and international leading researchers to address the challenges around the suitable programming language development.

Employment History

since 2023 **Associate Director HDR**, *School of Computing*, The Australian National University

since 2022 **Associate Professor**, *School of Computing*, The Australian National University

2022 **Deputy Head of School**, *School of Engineering and Computer Science*, Victoria University of Wellington, New Zealand

2021–2022 **Associate Dean (Students)**, *Faculty of Engineering*, Victoria University of Wellington, New Zealand

2019–2020 **Visiting Associate Professor**, *Kyoto University*, Japan

2018–2022 **Associate Professorl**, *School of Engineering and Computer Science*, Victoria University of Wellington, New Zealand

2013 **Visiting Research Scholar**, *Carnegie Mellon University*, USA

2010–2017 **Senior Lecturer**, *School of Engineering and Computer Science*, Victoria University of Wellington, New Zealand

2006–2009 **Lecturer**, *School of Engineering and Computer Science*, Victoria University of Wellington, New Zealand

2005–2006 **Software Engineer**, *Innaworks Limited*, Wellington, New Zealand

2004 **Visiting Researcher**, *Purdue University*, USA

2001–2003 **Software Engineer**, *Catalyst Systems Limited*, Wellington, New Zealand

Education

2003–2006 **PhD**, *Software Engineering*, Victoria University of Wellington, New Zealand

2002 **BSc (Hons)**, *Computer Science*, Victoria University of Wellington, New Zealand

1999–2001 **BSc**, *Mathematics and Computer Science*, Victoria University of Wellington, New Zealand

Research Interests

Type Systems, Verification and Program Synthesis, Object Capability Security, Secure Programming Language Design, Memory Management, and Full Stack Development

Professional Memberships

2024—2027 **Elected Member at Large**, *ACM SIGPLAN Executive Committee*

since 2023 **Member of Engineers Australia**

since 2023 **Permanent Member of the IFIP 2.4 Working Group on Software Implementation Technology**

since 2002 **Member of the ACM (currently: Senior Member of the ACM)**
2018-–2022 **Member of Engineering New Zealand**
2010-–2014 **Member of CORE (CS Departments of Australia/NZ) Executive**

Selected Prizes and Awards

2022 🏆 **FORTE 2022 Best Paper Award**
2020-–2021 **Robonomics.Network Research Grant ($72,000 NZD)**
2021—2023 **VUW SHEADI Faculty Strategic Initiative PhD Scholarship ($100,000 NZD)**
2017-–2018 **Oracle Corporation Research Grant ($70,000 NZD)**
2017 🏆 **ECOOP 2017 Distinguished Artefact Award.**
2014 🏆 **ECOOP 2014 Distinguished Paper Award.**
2009-–2011 **RSNZ Fast Start Marsden Grant ($300,000 NZD)**
2012 **Mozilla Foundation Research Grant ($15,000 NZD)**
2016, 2008, 2007 **VUW URF Grant ($18,000 NZD '16, $12,000 NZD '08, $17,000 NZD '07)**
2009 **RSNZ ISAT Grant ($5,420 NZD)**
2007 🏆 **ESEC/FSE ACM SIGSOFT Distinguished Paper Award**
2003–2005 **2nd Prize and the Judges Prize at the ICFP Programming Contest 2005 and 2003**
2003 **2nd Place in the ACM International Research Competition (via OOPSLA2002 SRC)**

Program or Review Committee Chair

2024 **Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA) – with Bur-Yuh Chang**
2016 **Asia-Pacific Software Engineering Conference (APSEC) – with Gail Murphy**
2010–2011 **Computing: The Australasian Theory Symposium (CATS) – with Taso Viglas**

Representative Publications

1. A. Potanin, J. Östlund, Y. Zibin, M. Ernst, *Immutability* in *Aliasing in Object-Oriented Programming. Types, Analysis and Verification*, Vol. 7850 of *Lecture Notes in Computer Science*, D. Clarke, J. Noble, T. Wrigstad (Eds.), Springer Berlin Heidelberg, **2013**, pp. 233–269
2. T. Runge, A. Kittelmann, M. Servetto, A. Potanin, I. Schaefer, *Information Flow Control-by-Construction for an Object-Oriented Language* in *Software Engineering and Formal Methods: 20th International Conference, SEFM 2022, Berlin, Germany, September 26–30, 2022, Proceedings*, Springer-Verlag, Berlin, Heidelberg, p. 209–226. `https://doi.org/10.1007/978-3-031-17108-6_13`
3. I. O. Gariano, M. Servetto, A. Potanin. Using capabilities for strict runtime invariant checking. *Sci. Comput. Program.* **2022**, *224*, 102878
4. J. Dietrich, K. Jezek, S. Rasheed, A. Tahir, A. Potanin, *Evil Pickles: DoS Attacks Based on Object-Graph Engineering* in *31st European Conference on Object-Oriented Programming, ECOOP 2017, June 19-23, 2017, Barcelona, Spain*, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, Vol. 74 of *LIPIcs*, pp. 10:1–10:32. `https://doi.org/10.4230/LIPIcs.ECOOP.2017.10`
5. D. Melicher, Y. Shi, A. Potanin, J. Aldrich, *A Capability-Based Module System for Authority Control* in *31st European Conference on Object-Oriented Programming (ECOOP 2017)*, Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, Vol. 74 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pp. 20:1–20:27. `http://drops.dagstuhl.de/opus/volltexte/2017/7270`
6. C. Omar, D. Kurilova, L. Nistor, B. Chung, A. Potanin, J. Aldrich, *Safely Composable Type-Specific Languages* in *ECOOP 2014 – Object-Oriented Programming*, Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 105–130
7. M. Servetto, J. Mackay, A. Potanin, J. Noble, *The Billion-Dollar Fix* in *ECOOP 2013 – Object-Oriented Programming*, Vol. 7920 of *Lecture Notes in Computer Science*, G. Castagna (Ed.), Springer Berlin Heidelberg, **2013**, pp. 205–229
8. A. Potanin, M. Damitio, J. Noble, *Are Your Incoming Aliases Really Necessary? Counting the Cost of Object Ownership* in *Proceedings of the 2013 International Conference on Software Engineering*, IEEE Press, Piscataway, NJ, USA, of *ICSE '13*, pp. 742–751. `http://dl.acm.org/citation.cfm?id=2486788.2486886`
9. 🏆 Y. Zibin, A. Potanin, M. Ali, S. Artzi, A. Kie—un, M. D. Ernst, *Object and reference immutability using Java generics* in *Proceedings of the the 6th Joint Meeting of the European Software Engineering Conference and the ACM SIGSOFT Symposium on The Foundations of Software Engineering*, Association for Computing Machinery, New York, NY, USA, of *ESEC-FSE '07*, p. 75–84. `https://doi.org/10.1145/1287624.1287637`
10. A. Potanin, J. Noble, M. Frean, R. Biddle. Scale-free Geometry in Object-Oriented Programs. *Communications of the ACM* **2005**