Reto **Achermann**

Systems · Researcher

University of British Columbia

achreto@gmail.com | retoachermann.ch | achreto | achreto

Summary

I am an operating system researcher with three years of post-graduate research and ten years of teaching experience. I make operating systems more secure, faster, and easier to develop by combining programming languages, formal methods, software synthesis, and hardware models.

Education

Doctor of Science, ETH Zurich

November 2014 - February 2020

Zurich, Switzerland

- Advisor: Prof. Timothy Roscoe
- Thesis Title: On Memory Addressing [pdf]

Master of Science in Computer Science, ETH Zurich

September 2013 - October 2014

Zurich, Switzerland

- Advisor: Prof. Timothy Roscoe
- Specialization: Distributed Systems. Grade: 5.8 / 6.0 (with distinction)
- Thesis Title: Message passing and bulk transport on heterogeneous multiprocessors [pdf]

Bachelor of Science in Computer Science, ETH Zurich

September 2009 - September 2013

Zurich, Switzerland

- Thesis Title: Barrelfish USB Subsystem [pdf]

Officer School Swiss Armed Forces, Swiss Government

June 2008 - September 2009

Switzerland

- Leadership education, management, and conflict management.

Professional Experience

Postdoctoral Research Fellow

December 2020 - present

University of British Columbia – Systopia Lab

Vancouver (BC), Canada

- Research in the intersection of operating systems, verification and software synthesis.
- Mentored several students on their research projects and co-advised two students on their Honours thesis.
- Teaching: Lecturer for CS508 - Graduate Operating Systems and CS436A - Operating Systems Design and Implementation

Research Assistant

November 2014 - November 2020

ETH Zurich – Systems Group

Zurich, Switzerland

- Research in operating systems and hardware specification
- Mentoring multiple students in their Master's and Bachelor's thesis projects.
- Teaching: Leading tutorials / labs as a teaching assistant for multiple courses.

Intern VMware Research Group

June 2018 - September 2018

VMware, Inc. – VMware Research Group

Palo Alto (CA), United States

- Project: Mitosis: Transparently Self-Replicating Page-Tables for Large-Memory Machines.
- Design and implementation of page-table replication in the Linux kernel.

Intern Systems Software

September 2015 - December 2015

Hewlett-Packard Labs – Systems Software Research Group

Palo Alto (CA), United States

- Project: Consensus protocols and capabilities for the Machine.

Staff Officer (Captain)

June 2009 - November 2020

Swiss Armed Forces – Radio transmission and IT services.

Switzerland

- Planning and deployment of heterogeneous communication systems, IT systems and networks.
- Leading education modules and training exercises with up to 350 people.

Teaching Experience

University of British Columbia

Lecturer	Graduate Operating Systems (CPSC 508)	*Winter Term 1, 2023*
Lecturer	Operating Systems Design and Implementation (CPSC436A)	*Winter Term 1, 2022*
Lecturer	Operating Systems Design and Implementation (CPSC436A/CPSC538A)	*Winter Term 2, 2021*
Lecturer	Graduate Operating Systems (CPSC 508)	*Winter Term 2, 2020*

ETH Zurich

Teaching Assistant	Advanced Operating Systems (263-3800-00L)	*2020 , 2019 , 2017*
	Systems Programming and Computer Architecture (252-0061-00L)	*2019 , 2018 , 2017 , 2016 , 2014 , 2013*
	Application-Oriented Programming (252-0840-02L)	*2019 , 2018*
	Introductory Programming in MATLAB (252-0840-01L)	*2017 , 2016*
	Parallel Programming (252-0024-00L)	*2015*
	Data Modelling and Databases (252-0063-00L)	*2014*
	Operating Systems and Networks (252-0062-00L)	*2013*

Peer Reviewed Publications

Synthesizing Device Drivers with Ghost Writer — *PLOS '23*

Bingyao Wang, Sepehr Noorafshan, **Reto Achermann** and Margo Seltzer — *2023*

Sharding the State Machine: Automated Modular Reasoning for Complex Concurrent Systems — *OSDI '23*

Travis Hance, Andrea Lattuada, **Reto Achermann**, Alex Conway, Ryan Stutsman, Gerd Zellweger, Chris Hawblitzel, Jon Howell and Bryan Parno — *2023*

Why write address translation OS code yourself when you can synthesize it? — *HotOS '23*

Reto Achermann, Ilias Karimalis and Margo Seltzer — *2023*

Beyond isolation: OS verification as a foundation for correct applications — *HotOS '23*

Matthias Brun, **Reto Achermann**, Tej Chajed, Jon Howell, Gerd Zellweger and Andrea Lattuada — *2023*

Cache-Coherent Accelerators for Persistent Memory Crash Consistency — *HotStorage '22*

Ankit Bhardwaj, Todd Thornley, Vinita Pawar, **Reto Achermann**, Gerd Zellweger and Ryan Stutsman — *2022*

Enzian: An Open, General, CPU/FPGA Platform for Systems Software Research — *ASPLOS '22*

David Cock, Abishek Ramdas, Daniel Schwyn, Michael Giardino, Adam Turowski, Zhenhao He, Nora Hossle, Dario Korolija, Melissa Licciardello, Kristina Martsenko, **Reto Achermann**, Gustavo Alonso and Timothy Roscoe — *2022*

Fast Sparse Decision Tree Optimization via Reference Ensembles — *AAAI '22*

Hayden McTavish, Chudi Zhong, **Reto Achermann**, Ilias Karimalis, Jacques Chen, Cynthia Rudin and Margo Seltzer — *2022*

Generating Correct Initial Page Tables from Formal Hardware Descriptions — *PLOS '21*

Reto Achermann, David Cock, Roni Haecki, Nora Hossle, Lukas Humbel, Timothy Roscoe and Daniel Schwyn — *2021*

Declarative Power Sequencing — *ACM TECS Vol 20-5s*

Jasmin Schult, Daniel Schwyn, Michael Giardino, David Cock, **Reto Achermann** and Timothy Roscoe — *2021*

NrOS: Effective Replication and Sharing in an Operating System — *OSDI '21*

Ankit Bhardwaj, Chinmay Kulkarni, **Reto Achermann**, Irina Calciu, Sanidhya Kashyap, Ryan Stutsman, Amy Tai and Gerd Zellweger — *2021*

Mmapx: Uniform Memory Protection in a Heterogeneous World — *HotOS '21*

Reto Achermann, David Cock, Roni Haecki, Nora Hossle, Lukas Humbel, Timothy Roscoe and Daniel Schwyn — *2021*

Fast Local Page-Tables for Virtualized NUMA Servers with vMitosis — *ASPLOS '21*

Ashish Panwar, **Reto Achermann**, Arkaprava Basu, Abhishek Bhattacharjee, K. Gopinath and Jayneel Gandhi — *2021*

Mitosis: Transparently Self-Replicating Page-Tables for Large-Memory Machines — *ASPLOS '20*

Reto Achermann, Ashish Panwar, Abhishek Bhattacharjee, Timothy Roscoe and Jayneel Gandhi — *2020*

Memory-Side Protection With a Capability Enforcement Co-Processor — *ACM TACO Vol 16-1*

Leonid Azriel, Lukas Humbel, **Reto Achermann**, Alex Richardson, Moritz Hoffmann, Avi Mendelson, Timothy Roscoe, Robert N. M. Watson, Paolo Faraboschi and Dejan Milojicic — *2019*

Physical Addressing on Real Hardware in Isabelle/HOL — *ITP'18*

Reto Achermann, Lukas Humbel, David Cock and Timothy Roscoe — *2018*

Formalizing Memory Accesses and Interrupts — *MARS'17*

Reto Achermann, Lukas Humbel, David Cock and Timothy Roscoe — *2017*

Towards Correct-by-Construction Interrupt Routing on Real Hardware — *PLOS'17*

Lukas Humbel, **Reto Achermann**, David Cock and Timothy Roscoe — *2017*

Separating Translation from Protection in Address Spaces with Dynamic Remapping — *HotOS '17*

Reto Achermann, Chris Dalton, Paolo Faraboschi, Moritz Hoffmann, Dejan Milojicic, Geoffrey Ndu, Alexander Richardson, Timothy Roscoe, Adrian L. Shaw and Robert N. M. Watson — *2017*

Machine-aware Atomic Broadcast Trees for Multicores — *OSDI'16*

Stefan Kaestle, **Reto Achermann**, Roni Haecki, Moritz Hoffmann, Sabela Ramos and Timothy Roscoe — *2016*

SpaceJMP: Programming with Multiple Virtual Address Spaces — *ASPLOS '16*

Izzat El Hajj, Alexander Merritt, Gerd Zellweger, Dejan Milojicic, **Reto Achermann**, Paolo Faraboschi, Wen-mei Hwu, Timothy Roscoe and Karsten Schwan — *2016*

Shoal: Smart Allocation and Replication of Memory for Parallel Programs — *USENIX ATC '15*

Stefan Kaestle, **Reto Achermann**, Timothy Roscoe and Tim Harris — *2015*

Not Your Parents' Physical Address Space — *HotOS'15*

Simon Gerber, Gerd Zellweger, **Reto Achermann**, Kornilios Kourtis, Timothy Roscoe and Dejan Milojicic — *2015*

Other Publications

OSmosis: Modeling and Building Flexible OS Isolation Mechanisms (Poster) — *SOSP '23:Poster Sessions*

Sidhartha Agrawal, Shaurya Patel, **Reto Achermann** and Margo Seltzer — *2023*

OSmosis: No more Déjà vu in OS isolation — *arXiv:2309.09291*

Sidhartha Agrawal, **Reto Achermann** and Margo Seltzer — *2023*

Secure Memory Management on Modern Hardware — *arXiv:2009.02737*

Reto Achermann, Nora Hossle, Lukas Humbel, Daniel Schwyn, David Cock and Timothy Roscoe — *2020*

CleanQ: a lightweight, uniform, formally specified interface for intra-machine data transfer — *arXiv:1911.08773*

Roni Haecki, Lukas Humbel, **Reto Achermann**, David Cock, Daniel Schwyn and Timothy Roscoe — *2019*

Cichlid: Explicit physical memory management for large machines — *arXiv:1911.08367*

Simon Gerber, Gerd Zellweger, **Reto Achermann**, Moritz Hoffmann, Kornilios Kourtis, Timothy Roscoe and Dejan Milojicic — *2019*

A Least-Privilege Memory Protection Model for Modern Hardware — *arXiv:1908.08707*

Reto Achermann, Nora Hossle, Lukas Humbel, Daniel Schwyn, David Cock and Timothy Roscoe — *2019*

Mitosis: Transparently Self-Replicating Page-Tables for Large-Memory Machines — *arXiv:1910.05398*

Reto Achermann, Ashish Panwar, Abhishek Bhattacharjee, Timothy Roscoe and Jayneel Gandhi — *2019*

Mitosis: Transparently Self-Replicating Page-Tables for Large-Memory Machines (Poster) *OSDI'18:Poster Sessions*

Reto Achermann, Ashish Panwar, Abhishek Bhattacharjee, Timothy Roscoe and Jayneel Gandhi *2018*

Sockeye: Formally Describing Hardware as Seen by Software (Poster) *OSDI'18:Poster Sessions*

Reto Achermann, Lukas Humbel, Roni Haecki, David Cock and Timothy Roscoe *2018*

Enzian: a research computer for datacenter and rackscale computing (Poster) *OSDI'18:Poster Sessions*

David Cock, David Sidler, Muhsen Owaida, **Reto Achermann**, Tobias Grosser, Zeke Wang, Amit Kulkarni, Alain Denzler, Adam Turowski, Abishek Ramdas, Anastasiia Ruzhanskaia, Timothy Roscoe and Gustavo Alonso *2018*

Formalizing Address Space Interactions (Poster) *OSDI'16:Poster Sessions*

Reto Achermann and Timothy Roscoe *2016*

Patents

Transparent Self-Replicating Page Tables in Computing Systems *US20200117612A1*

Reto Achermann and Jayneel Gandhi *January 2019*

Interoperable capabilities *US20170329526A1*

Reto Achermann, Maurice Bailleu, Dejan S. Milojicic and Gabriel Parmer *January 2016*

Memory management with versioning of objects *WO2017131789A1*

Izzat El Hajj, Alexander Merritt, Gerd Zellweger, Dejan S. Milojicic and **Reto Achermann** *January 2016*

Awards

Intel Best Paper Award	Cache-coherent accelerators for persistent memory crash consistency (HotStorage '22)	*2021*
EuroSys '21 Shadow PC	Honorable Mention Distinguished Shadow PC Reviewer Award	*2021*
HiPEAC Paper Award	SpaceJMP: Programming with Multiple Virtual Address Spaces (ASPLOS'16)	*2016*

Projects

Verus - Verifying Rust Programs *github.com/verus-lang/verus*

Verus is a tool to verify rust programs with the focus on system software using SMT solver.

Velosiraptor - Why program when you can automatically synthesize OS code? *retoachermann.ch*

Velosiraptor provides a specification language to express the behavior of translation hardware. Based on the specification, Velosiraptor automatically synthesizes the operating-system code that interfaces with the hardware and correctly configures the translation hardware.

IronSync *github.com/secure-foundations/iron-sync*

IronSync is an automated verification framework for concurrent code with shared memory. Seagull scales to complex systems by splitting system-wide proofs into isolated concerns such that each can be substantially automated. IronSync uses separation logic to split the state machine into shards to enable sound local reasoning

Node-Replication Operating System *nrkernel.systems/*

Scalability and concurrency are notoriously hard. The node-replication operating system (NrOS) simplifies this by leveraging node-replication that automatically makes sequential data structures scalable and concurrent. NrOS maintains replicas of the data structure on each NUMA node and uses an operation log to ensure consistency among replicas.

CleanQ - Lightweight, uniform and formally specified Queues. *cleanq-project.github.io*

CleanQ is a queue specification for intra-machine data transfers. The specification defines the semantics of ownership transfers in Isabelle/HOL. We prove the correctness through a sequence of refinement steps down to the C implementation of a queue, and demonstrate that the resulting implementation is lightweight and observes good performance.

Sockeye - Formally Specifying Hardware as Seen by Software *sockeye-project.github.io*

Sockeye is a framework for accurately representing hardware configurations. Faithful hardware abstractions used by operating systems to represent memory, interrupt, power and clock domain configurations of heterogeneous computer systems and formally define the semantics thereof. With Sockeye, platform-specific operating systems code can be generated (e.g. correct-by-construction page-tables). Sockeye is integrated into the Barrelfish OS.

Mitosis - Page-Table Replication for Big Memory Workloads *github.com/mitosis-project*

Big-memory workloads spend a significant fraction of their runtime serving TLB misses. Walking page tables require up to 24 memory accesses and experience NUMA effects. Mitosis transparently replicates page-tables across sockets to eliminate cross-socket page-table walks. Implementation of Linux kernel extensions and application runtime.

Enzian - A Research Computer *enzian.systems*

Enzian is a research computer combining a big server-class ARM CPU with a large FPGA connected through the Enzian Coherency Interface (ECI). My work targeted the understanding of the processor's native coherence protocol driving the development of ECI.

Barrelfish Operating System *barrelfish.org*

Barrelfish is a Multikernel-based research operating system developed at ETH Zurich. Part of this project, my work consists of architectural support (Xeon Phi, ARMv8), device drivers (Xeon Phi co-processor, USB, DMA drivers), runtimes (bulk-transport subsystem, OpenMP, multiple-virtual address spaces, Shoal runtime).

Smelt - Machine Aware Message-Passing Primitives *github.com/libsmelt*

Machine-optimized construction of broadcast and reduction trees as message-passing primitives. Smelt targets multi-core systems. The project work consisted of the design and implementation of the Smelt runtime library including message-passing abstractions.

Shoal - Smart Memory Allocation for NUMA Machines *github.com/libshoal*

Automatic optimization of memory allocation for parallel programs (Graph processing in Green-Marl) based on access patterns. Project work consisted of the design and implementation of the memory abstractions, Barrelfish runtime support and support for DMA engines of the runtime.

Scientific Presentations

Why write code when you can synthesize address translations? *June 22, 2023*

19th Workshop on Hot Topics in Operating Systems

Fast Local Page Tables for NUMA Servers with Mitosis *June 08, 2021*

ICSA Colloquium – University of Edinburgh

Mitosis: Transparently Self-Replicating Page-Tables for Large-Memory Machines *March 19, 2020*

25th International Conference on Architectural Support for Programming Languages and Operating Systems

Memory Topology Models and Their Application in Operating Systems *February 25, 2020*

Trustworthy Systems, Data 61, CSIRO, Australia.

Faithful Hardware Representation and Least-Privilege Memory Management in Operating Systems *December 06, 2019*

Lab for Advanced Systems Research (LASR), University of Texas at Austin, USA

Realistic Hardware Abstractions and Least-Privilege Memory Management in Operating Systems *November 01, 2019*

Networks, Systems, and Security (NSS) Lab, University of British Columbia, Canada

Mitosis: Transparently Self-Replicating Page-Tables for Large-Memory Machines *November 30, 2018*

Systems Group, ETH Zurich, Switzerland.

Model based system configuration and tasteful hardware *July 06, 2017*

Systems Research Group, University of Cambridge, Cambridge, UK

Provable Correct Memory Management *April 23, 2017*

11th EuroSys Doctoral Workshop

Smelt: Machine-aware Atomic Broadcast Trees for Multicores *November 2, 2016*

12th USENIX Symposium on Operating Systems Design and Implementation

Advised and Mentored Students

University of British Columbia

Phillip Dumitru — Co-Advisor Honours's thesis project. Co-advised with Prof. Alexander Summers. Title: *Formalization of the OSMosis Isolation Model* — *September 2023 - April 2024*

Brice Michael Wilson — Co-Advisor Honours's thesis project. Co-advised with Prof. Margo Seltzer. Title: *Profiling Stack Traces in Non-Linux Virtual Machines* — *September 2022 - April 2023*

Ryan Mehri — Mentor directed studies research project. Topic: *Encoding memory models for address translation in an SMT solver.* — *January 2023 - August 2023*

Emily Chu — Mentor summer research project. Topic: *Generating translation hardware components for the Arm FastModels simulator.* — *June 2023 - December 2023*

David Bromley — Mentor summer research project. Topic: *Formally specifying optimal sparse decision trees (OSDT) in Dafny.* — *June 2022 - August 2022*

Sepehr Noorafshan — Mentor summer research project. Topic: *Revisiting device driver synthesis with behavior trees.* — *June 2022 - August 2022*

Ilias Karimalis — Mentor summer research project. Topic: *Design and implementation of a DSL to specify translation hardware.* — *June 2022 - August 2022*

ETH Zurich

Joel Busch — Mentor Master's thesis project. Title: *Detailed Simulation of Enzian's Cache Coherence Protocol* — *April 2020 - October 2020*

Thore Goebel — Mentor Bachelor's thesis project. Title: *CleanQ for USB* — *February 2020 - August 2020*

Patrick Ziegler — Mentor Bachelor's thesis project. Title: *A Unified Approach to Simulation of Hybrid CPU/FPGA systems* — *October 2019 - April 2020*

Jakob Meier — Mentor Master's thesis project. Title: *Tools for Cache Coherence Protocol Interoperability* — *September 2018 - March 2019*

Nora Hossle — Mentor Master's thesis project. Title: *Multiple Address Spaces in a Distributed Capability System* — *March 2018 - September 2019*

Leo Horne — Mentor Bachelor's thesis project. Title: *Using NetBSD Kernel Components on Barrelfish Through Rump Kernels* — *February 2019 - August 2019*

Giuseppe Arcuti — Mentor Bachelor's thesis project. Title: *Formally modelling hardware standards* — *February 2019 - August 2019*

Sven Knobloch — Mentor Bachelor's thesis project. Title: *System Modeling Co-Design* — *March 2018 - September 2018*

Joel Busch — Mentor Bachelor's thesis project. Title: *Device Queues for USB* — *December 2017 - May 2018*

Daniel Schwyn — Mentor Master's thesis project. Title: *Hardware Configuration With Dynamically-Queried Formal Models* — *April 2017 - October 2017*

Andrei Poenaru — Mentor Master's thesis project. Title: *Explicit OS support for hardware threads* — *September 2016 - March 2017*

David Keller — Mentor Bachelor's thesis project. Title: *Dynamic Linking and Loading in Barrelfish* — *February 2015 - August 2015*

Service

EuroSys '25	Program Committee	*2025*
EuroSys '24	Program Committee	*2024*
ASPLOS '24	Program Committee	*2024*
ACM SRC '23	Program Committee	*2023*
ATC '23	Program Committee	*2023*
EuroSys '23	Program Committee	*2023*
ASPLOS '22 (ERC)	Program Committee	*2022*
ATC '22	Program Committee	*2022*
IEEE Internet Computing	Reviewer	*2022*
ATC '21	Networking Co-chair	*2021*
OSDI '21	Networking Co-chair	*2021*
ATC '21	Program Committee	*2021*
EuroSys '21	Shadow Program Committee	*2021*

References

Available on request.