






# Matthew Edwin Weingarten

 scholar |  matt@weingarten.org |  +41 79 920 37 68 |  mattweingarten |  matt-weingarten

## EDUCATION

### Computer Science MSc — ETH Zurich

Feb '21 - May '23

*Major in Systems and minor in Programming Languages*

GPA: 5.41/6

**Relevant Courses:** Cloud Computing Architecture(5.25/6), Advanced Operating Systems(5.75/6), Future Internet(5.5/6), Automated Software Testing(6/6), Advanced Systems Lab(5.25/6), Program Verification(5.75/6), Big Data(5.75/6)

### Computer Science BSc — ETH Zurich

Sep '17 - Feb '21

*Focus in Systems and Information/Data processing*

GPA: 4.75/6

**Relevant Courses:** Compiler Design(5.75/6), Computer Systems(5/6), Computer Networking(5.5/6), Parallel Programming(5.25/6), Databases(5.5/6)

## RESEARCH EXPERIENCE

### Scientific Assistant — ETH Zurich

Jul '23 - Sep '23

*Systems Group, supervised by Prof. Dr. Timothy Roscoe*


Independent research on hardware accelerated analysis of CoreSight traces with FPGAs on heterogeneous systems, building on *masters's thesis*  and writing *paper* .


Added tracing capabilities for the custom tracing hardware on the ThunderX chip to enable CoreSight tracing for Enzian.

### Research Intern — Oracle Labs

Sep '21 - Sep '22


*GraalVM Compiler Research, supervised by Dr. Aleksandar Prokopec*


Invented *Interprocedural Partial Escape Analysis*  that combines static escape analysis techniques with profile-guided optimizations to make better inlining decisions and reduce memory allocations, with performance improvements in server-grade workloads of up to 24%.

Designed a *Lockfree Prefix Tree*  internal JavaVM data structure to efficiently sample call stacks in profiling runs used in production.

Intimately studied DaCapo, ScalaDacapo, and Renaissance benchmark suites in search of performance anti-patterns and assisted in the development of optimizations including context-sensitive inlining and incremental inlining and profiling techniques, including sampling-based profile collection.

## ACCEPTED PEER-REVIEWED PUBLICATIONS

**M. E. Weingarten**, N. Hossle, T. Roscoe. *High Throughput Hardware Accelerated CoreSight Trace Decoding*. To appear in IEEE Design & Test (DATE'24). 

**M. E. Weingarten**, T. Theodoridis, A. Prokopec. *Inlining-Benefit Prediction with Interprocedural Partial Escape Analysis*. In Proceedings of the 14th International Workshop on Virtual Machines and Intermediate Languages (VMIL'22). 

## MASTER'S THESIS

### Hardware Accelerated Trace Analysis for Compiler Optimizations

Oct '22 - May '23

*ETH, Systems Group, supervised by N Hossle and Prof. Dr. T Roscoe and*

Built a system that collects zero-overhead runtime traces from the CoreSight hardware components and analyzes the trace data in real-time on an FPGA.

Implemented a trace decoder for the Embedded Trace Macrocell specification on an FPGA with 8× worst-case throughput increase over all prior work.

Worked intimately with CoreSight components on various ARM CPUs, such as the ThunderX, Cortex-A53, and Cortex-A7, including extending the CoreSight Access Library maintained by ARM to support more complex CoreSight topologies.

## TEACHING EXPERIENCE

### Big Data Teaching Assistant | Spark, Map-Reduce, HDFS, HBase, S3, Yarn, MongoDB, Neo4J, Rumble

Fall '22

*Graduate Course, Systems Group, taught by Dr. G Fourny*

Conducting weekly hands-on classes and creating exercise questions for ~20 students to teach large-scale database technologies, including the study of data formats, resource management, data models, and parallel processing.

### Computer Systems Teaching Assistant

Fall '20 & Fall '21

*Undergrad 3rd year course, Systems Group, taught by Prof. Dr. T Roscoe and Prof. Dr. R Wattenhofer*

Taught a group of ~20 students operating and distributed system fundamentals and developed the auto-grader and master solution for the semester project on socket server programming.



## TALKS & PRESENTATIONS

IEEE Design & Test (DATE) (upcoming) — Poster & Author presentation

Mar '24

Virtual Machines and Language Implementations (VMIL) — Author presentation

Dec '22

**Static Single-Assignment Compiler for Strict Functional Languages**   | MLIR, LLVM, C, C++, Lean Fall '20  
*ETH, Advanced Software Technologies Lab, supervised by Prof. Dr. T Grosser & Prof. Dr. Z Su*



Implemented a compiler using the LLVM & MLIR ecosystem to parse high-level lambda calculus IR, convert it to SSA form, and lower to C/LLVM code integrated with the language runtime.

Applied graph-based rewrite rules to insert explicit reference counting instructions for memory management and to allow for destructive updates based on runtime reference counts to avoid copying objects when updating fields reducing allocations up to 50%.

## SELECTED PROJECTS

**Influence of Compiler Optimizations on Fuzzing**   | Clang, LLVM, AFL++, Fuzzbench, Docker, Bash, C Spring '22  
*Automated Software Testing, Prof. Dr. Z Su*

Provided comprehensive statistical analysis on the effects of modified and unmodified compiler flags on fuzzing performance such as the O-level flags, unrolling, inlining, and simplifying CFG.

**DÆM: Denoising Autoencoder Model for Collaborative Filtering**   | Python, TensorFlow Spring '22  
*Computational Intelligence Lab, Prof. Dr. G Rätsch*

Built a Denoising Autoencoder Model for collaborative filtering using a customized loss function, outperforming matrix factorization techniques like ALS and SVD.

**Operating System Development on Barrelfish**   | Barrelfish, C, Assembly Spring '21  
*Advanced Operating Systems, Prof. Dr. T Roscoe and Dr. D Cock*

Partnered with three other graduate students to build a multi-core & exokernel-based OS, including message passing with UMP and LMP, to memory sharing between different kernel instances running on separate cores and more.

Implemented the nameserver and process manager for interprocess communication, bootstrapping communication channels, and handling dead processes.

**Optimizing SQL Queries on Bit-Parallel Database Layouts**   | AVX, C, Assembly Spring '21  
*Advanced Systems Lab, Prof. Dr. M Püschel and Prof. Dr. C Zhang*

Took a deep dive into characterizing critical execution bottlenecks and hand-crafting high-performance software using vectorization, strength reduction, improving cache hit rates, and filling processor instruction pipelines resulting in 40x speedups for SQL queries.

**Cloud Application Scheduler** | Google Cloud Platform, Python, Bash Spring '21  
*Cloud Computing Architecture, Prof. Dr. G Alonso and Prof. Dr. A Klimovic*

Benchmarked, examined, and categorized a mix of server jobs (e.g. memcached) and batch jobs (e.g. dedup) and devised a scheduling and collocation algorithm to maximize efficiency while QOS requirements on a server cluster.

**Neural Network Certification** | Python, PyTorch Fall '20  
*Reliable and Interpretable Artificial Intelligence, Prof. Dr. M Vechev*

Verified neural networks and proved robustness against adversarial attacks on large fully-connected and convolutional neural networks using DeepPoly.

## ACHIEVEMENTS

**Patent (Pending):** A. Prokopec, **M. E. Weingarten**, P. Woegerer, C. Wimmer. Apr '23  
*Improving Program Execution using Interprocedural Escape Analysis with Inlining*

## COMMUNITY INVOLVEMENT

**Board Member & President of Study Support Committee — VIS** Sep '21 - Sep '22  
*Association of Computer Science Students at ETH (VIS)*

Led a team of 10 students responsible for the organization of exam preparation workshops attended by over 300 students with a budget of 30K CHF.

## SKILLS

**Languages:** C, C++, Assembly, VHDL, Java, Scala, Haskell, OCaml, Lean, Javascript, Python, Ruby, SQL, R, Matlab, Bash, LaTeX

## STATUS

**Compulsory Military Service** Jun '23, Oct '22, Jul '18, Mar '17 - Aug '17  
**United States of America Citizenship & Switzerland Citizenship**  
**English & German Native Speaker**