

# Lei Yan

---

I'm a PhD student at EPFL working with [George Candea](#) and [Sanidhya Kashyap](#). My research interests lie in designing abstractions and tools that simplify the development of concurrent systems software. I focus currently on abstractions for transparent parallelization and worked a bit on tools for debugging concurrency bugs.

I enjoy hacking low-level stuff and hunting down tricky bugs (unless a few days before deadlines...). I also like geography and play Microsoft Flight Simulator 2020.

---

## Education

- 2019-2025 (expected)**      **PhD, Computer Science;** EPFL (Lausanne, Switzerland)  
*Topic: Transparent scaling of single-threaded systems software*  
Advisors: [George Candea](#), [Sanidhya Kashyap](#)
- 2017-2018**      **Exchange master study, Computer Science;** EPFL (Lausanne, Switzerland)
- 2015-2019**      **MSc, Computer Engineering;** RWTH Aachen University (Aachen, Germany)
- 2011-2015**      **BEng, Electronic Science & Technology;** Zhejiang University (HangZhou, China)

## Publications

### [HPCA 2023] [AstriFlash: A Flash-Based System for Online Services](#)

*Siddharth Gupta, Yunho Oh, **Lei Yan**, Mark Sutherland, Abhishek Bhattacharjee, Babak Falsafi, and Peter Hsu*

## Experience

- 2022**      **Research Intern;** Huawei Dresden Research Center (Dresden, Germany)  
*Topic: Mantis: hunting down concurrency bugs with minimal human effort*  
Advisors: Diogo Behrens
- 2018-2019**      **Research Intern;** Oracle Labs (Zurich, Switzerland)  
*Topic: Detecting security anomalies from Linux system logs using ML techniques*  
Advisors: [Matteo Casserini](#), [Milos Vasic](#)
- 2018**      **Research Assistant;** Parallel Systems Architecture Lab (PARSA), EPFL (Lausanne, Switzerland)  
*Topic: DSL for accelerating in-memory data services by exploiting memory-level parallelism*  
Collaborator: [Fengyun Liu](#)  
Advisors: [Dmitrii Ustiugov](#), [Babak Falsafi](#)

Pointer-chasing operations (e.g., binary tree search) on large dataset have low IPCs since they involve sequence of dependent loads from main memory. [AMAC \[Kocberber, VLDB 15\]](#) improves performance of such operations by interleaving multiple operations. The CPU can execute instructions of other operations while prefetching data for one operation instead of blocking on the memory loads. One problem of AMAC is that it requires significant code changes since the developer has to do the prefetching and operation interleaving manually. We solve this problem in this project by designing a DSL based on coroutines. With this DSL, the programmer only needs to identify pointer-chasing operations, the DSL automatically turns each operation into coroutines and interleaves them to hide memory load latency. This DSL allows developers to get the performance benefit of AMAC while only changing a few lines of code. It introduces only limited performance overhead compared to AMAC.

---

## Teaching

- [Principles of Computer Systems \(2020, 2021\)](#)
- [Software Development Project \(2021, 2022, 2023\)](#)

---

## Skills

- Hacking:
  - Fluent: C, perf, DPDK, operating systems, performance analysis.
  - Mid-level: Bash, C++, [VPP](#), networking, computer architecture.
  - Beginner: Python, Kernel development, x86/Arm assembly, graphics, ML.
- Human Languages:
  - Chinese (native speaker), English (fluent), German (Mid-level)

---

## Honors

- EPFL EDIC fellowship (2019-2020)
- Scholarship of RWTH Aachen Education Fund (2016-2017)
- RWTH Dean's List (2015-2016)
- Samsung Scholarship, Zhejiang University (2012-2013)

---

[l.yan@epfl.ch](mailto:l.yan@epfl.ch) • [Github](#) • [Linkedin](#)

(This resume is created based on [pandoc\\_resume](#). Last updated on Mar 23rd, 2023)