

# WEICHU YANG

1210 W Dayton St, Madison, WI 53706

☎ +1-6085903059 ✉ [weichu@cs.wisc.edu](mailto:weichu@cs.wisc.edu) 🔗 [Linkedin](#) 🏠 [Personal Website](#) 🐙 [Github](#)

## Education

### University of Electronic Science and Technology of China (UESTC)

Sep. 2020 – May 2024

*BEng in Computer Science and Technology*

*Chengdu, China*

- Major: Computer Science and Technology, Cumulative GPA: 3.93/4.00
- Awards: *Outstanding Student Scholarship of UESTC* (Thrice), *Game Security Scholarship of Tencent* (5th out of 120)

### University of Wisconsin–Madison (UW-Madison)

Sep. 2024 – present

*MS in Computer Science*

*Madison, WI*

- Course enrolled: CS752 Adv Architecture, CS757 Adv Architecture II, CS540 Intro to AI, ESL343: Oral Communication Skills
- Teaching Assistant for: CS544 Intro to big data system (24 Fall, 25 Spring)

## Internship Experience

### Tencent Technology (Shenzhen) Co.Ltd

Jun. 2022 – Sep. 2022

*Intern Security Enginner*

*Shenzhen, China*

- Performed binary-level logical analysis towards more than 10 malicious software.
- Provided general countermeasure recommendations.
- Developed a malicious sample software based on those analysed .

## Selected Projects

### Integrating uPIMulator into Gem5 and Optimizing Memory Transfers | C++,Python

Nov. 2024 – Present

- Encapsulated uPIMulator, a cycle-level PIM simulator, as a Simobject in Gem5 and extend it to enable CPU↔DPU communication.
- Implemented a memory transfer latency model based on integrated Gem5\_uPIMulator, to better analyze the memory transfer overhead, which is identified as the bottleneck in the overall performance.
- Designed and implemented a potential optimization strategy that partitions a single workload into streams with smaller data blocks and executes them in a staged, pipelined fashion, and yield up to a 1.9× performance improvement in the CPU→DPU data transfer and DPU computation phases.
- Here is the detailed report: [https://xenoppy.github.io/files/Gem5\\_uPIMulator.pdf](https://xenoppy.github.io/files/Gem5_uPIMulator.pdf)

### Host Co-operate SSD Filesystem | C, C++

July. 2023 – Dec. 2023

*Designed and implemented a cooperative filesystem that optimizes I/O performance by coordinating between the host and emerging computing-capable SSD, especially in the case of disaggregated computing and storage resources.*

- Proposed a collaborative solution that offloads certain filesystem operations from the host to the SSD, such as Path-Lookup and File-Mapping.
- Developed a highly adaptive logging system to ensure maximum reliability with minimal overhead for this filesystem.
- Customized and pruned an open-source SSD simulator to provide a simulation platform evaluating the performance of the computing-capable SSD.
- Conducted tests and analysis on the prototype demo, demonstrating reduced write amplification and alleviated bus transmission pressure.

### Heterogeneous Filesystem Based on Non-Volatile Memory | C, C++

Nov. 2021 – Oct. 2022

- Proposed an innovative space layout management scheme that stores metadata on high-performance NVM and file data on SSD separately, reducing random Disk I/O.
- Devised a buffer solution that fully utilized NVM as write buffer, consolidating multiple write disk requests into a single sequential write request for enhanced write throughput.
- Implemented the prototype filesystem using Filesystem in Userspace(FUSE) and leveraged the Storage Performance Development Kit(SPDK) to drive the SSD while bypassing the kernel.

## Skills

**Programming Languages:** C/C++, Python, Lisp, SQL, Java

**Developer Tools:** VS Code, Git, Gdb, Vim, Visual Studio

**Technologies/Frameworks:** Linux, MPI, L<sup>A</sup>T<sub>E</sub>X, Lex&Yacc, CUDA C